Universitat de Girona

UNSTRUCTURED PDP SOCIAL SEARCH QUERY ROUTING ALGORITHMS FOR AGENTIFIED SOCIAL NETWORKS

Albert Trias i Mansilla

Dipòsit legal: Gi. 460-2014
http://hdl.handle.net/10803/131395

http://creativecommons.org/licenses/by-nc/4.0/deed.ca

Aquesta obra està subjecta a una llicència Creative Commons Reconeixement-NoComercial

Esta obra está bajo una licencia Creative Commons Reconocimiento-NoComercial

This work is licensed under a Creative Commons Attribution-NonCommercial licence

Universitat de Girona

DOCTORAL THESIS

Unstructured P2P social search query routing algorithms for agentified social networks

Albert Trias i Mansilla

2013

Universitat de Girona

DOCTORAL THESIS

Unstructured P2P social search query routing algorithms for agentified social networks

Albert Trias i Mansilla

2013

PROGRAMA DE DOCTORAT EN TECNOLOGIA

Dirigida per: Dr. Josep Lluís de la Rosa i Esteva

Memòria presentada per a optar al títol de Doctor/a per la Universitat de Girona

Publications

The following thesis is presented as a compendium of publications. The regulations for the requirements in the PhD technology program are the following:

- Three papers, of which **one must be accepted in a first quarter journal** or two accepted in second quarter journals.

The papers that meet the requirements are the following:

- Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva. **Asknext: An Agent Protocol for Social Search**. Information Sciences 190 (2012): 144-161 http://dx.doi.org/10.1016/j.ins.2011.12.012
  According to JCR 2011, Information Sciences has an Impact Factor of 2.83, is ranked 9/135 in the area of "Computer Science, Information Systems" and belongs to Q1.
- Albert Trias i Mansilla, Eusebi Calle Ortega, Josep Lluís de la Rosa i Esteva, Marc Manzano Castro. **Studies on Social Networks Topologies and their effect in the Dissemination of Information in P2P Social Search**. Submitted to International Journal of Communication Systems.
  According to JCR 2011, International Journal of Communication Systems has an Impact Factor of 0.41, is ranked 63/75 in the area of "Telecommunications" and belongs to Q4.
- Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva. **Question Waves: an algorithm that combines answer relevance with speediness in social search**. Information Sciences (third review).
  According to JCR 2011, Information Sciences has an Impact Factor of 2.83, is ranked 9/135 in the area of "Computer Science, Information Systems" and belongs to Q1.

Although the regulation does not accept literature review papers in the papers that are used to meet the requirements, an addition paper that corresponds to the literature review chapter of this thesis has been submitted:

- Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva. **Survey of social search from the perspectives of the village paradigm and online social networks.**
  According to JCR 2011, Journal of Information Sciences has an Impact Factor of 1.30, is ranked 46/135 in the area of "Computer Science, Information Systems" and belongs to Q2. In the area of "Information Science & Library Science" is ranked 24/83.

Aquesta tesi s'ha realitzat en el marc dels projectes TIN2010-17903 (Comparative approaches to the implementation of intelligent agents in digital preservation from a perspective of the automation of social Networks) i IPT-430000-2010-13 (SAKE). I ha estat realitzada amb una beca pre-doctoral de la Universitat de Girona (BR09/10).

Agraïments

Arribat aquest punt hi ha moltes coses per agrair, més de les que puc recordar i massa per que pugui quedar satisfet en acabar aquestes línies, per molt que m'hi escarrassi.

Intentaré no deixar-me a ningú, però serà impossible, sou molts els que hi heu contribuït, d'una manera més directa o indirecta, ja que el present treball representa el final d'un camí, o millor dit d'una etapa; el camí continua. Malgrat que els darrers anys hagin sigut els de major contribució per aquest treball, no em quedaria tranquil si obviés a tots aquells amics, companys i professors que m'han acompanyat i guiat durant tot el camí de formació previ, tant a l'escola Dr. Masmitjà, com a la Salle de Girona i a la Universitat de Girona; us ho agraeixo de tot cor, encara que no entri en detalls. Seria injust deixar-me als companys del C.E. Gerunda, especialment a aquells que armats de paciència es van implicar en la meva formació i em van despertar l'interès de fer-me preguntes, buscar el perquè de les coses, a planificar i prioritzar objectius, sense deixar de banda els valors de l'esport, i que el més important no és el resultat sinó la satisfacció de la feina ben feta.

Pel que fa més a aquesta darrera etapa, en primer terme, vull agrair a en **Peplluís**, el meu director de tesi, la seva col·laboració en tot el procés. No oblido com va sorgir aquesta tesi doctoral, quan jo ja donava els meus estudis dins l'àmbit tecnològic per finalitzats. Un dia, de bon matí, quan anava a treballar ens vam creuar pel carrer, vam parlar un parell de minuts i al cap d'un parell de setmanes em vas enviar un correu electrònic comentant-me que estaves buscant gent per a fer una tesi doctoral. La veritat és que no vaig trigar gaire en decidir-me, veia que seria un període clarament més creatiu, i no me'n penedeixo de la decisió que vaig prendre. Durant aquest període m'has brindat tot allò que un investigador en formació necessita: confiança; un lloc i entorn de treball adient; supervisió; suggeriments del treball; discussió científica; i sempre has trobat el moment per satisfer els meus dubtes. A més a més has sigut flexible i comprensiu.

També vull donar les gràcies a tots els companys d'**ARLab**, tant als que encara continuen per aquí, com als que han canviat d'aires. Especialment vull donar les gràcies a **l'Albert Figueras**; l'**Esteve de l'Acebo**; en **José Antonio Olvera**; en **Nicolás Hormazábal**; en **Gábor Dobrocsi**; l'**Eduard Muntaner**; la **Silvana Aciar**; i en **Gabriel Lopardo**. Tampoc vull menystenir a la resta de companys del Centre Easy. I així mateix vull agrair als companys del laboratori **BCDS**, especialment a l'**Eusebi Calle** i en **Marc Manzano**, per tota l'ajuda en els aspectes referents a xarxes socials. A l'**Àngels Merino,** cap de la biblioteca del campus de Montilivi, per tota l'ajuda que em va donar per entendre els processos de la biblioteca, que em van servir per poder comprar el paradigma de la biblioteca amb el del poble. A l'**Àngels Carreiro** per la seva paciència en resoldre tots els meus dubtes sobre la normativa de doctorat.

També vull donar les gràcies als meus pares, l'**Eduard** i la **Joana**, per molt que soni a tòpic. Però que jo sigui el 3r de 3 fills que escriu una tesi, ha de ser molta casualitat perquè no hi hageu tingut res a veure! Mama, tu sempre havies dit que t'hauria fet gràcia que algun dels teus fills fos doctor, i mira! Bé, tots sabem que no et referies ben bé a això... Deixant les bromes de banda, de ben petits sempre ens heu mostrat la importància de l'educació; de l'esforç; d'intentar-se superar a un mateix; que sempre hi ha marge de millora. Procurant que no perdéssim de vista els valors, a la vegada que ens brindàveu tot el vostre amor i comprensió, i procuràveu pel nostre bé, per sobre de tot, malgrat les dificultats.

Als meus germans, en **Dani** i la **Ras**. El germà petit sempre és el més torracollons, i aquests darrers anys no us he deixat descans preguntar-vos coses relacionades sobre la tesi. Però no us he d'agrair només això, són minúcies comparat amb el tot. Sempre heu estat al meu costat, i heu donat molt per mi, fins i tot avantposant els meus interessos als vostres.

A l'**Anna**, la meva xicota, que ha patit més directament tot el procés, per tota la teva paciència, per compartir totes les meves alegries i penes, per tot el que m'estimes i el que em dones que em veig incapaç d'expressar en paraules.

Als companys i amics del **C.V. 4-Rius**; i a en **Marc**, l'**Andrea** i en **Jordi** per totes les bones estones.

A tots els que em deixo, a la resta de familiars i amics. No em sentiria còmode si deixés de banda a aquells que ja no hi són, fent èmfasi als que ens han deixat els darrers anys: el tito **Pedro** i el Sr. **Antoni Mediñà**.

Content

Abstract

In 2010 Horowitz and Kamvar explained that there are mainly two paradigms for finding relevant information. The village paradigm is to ask someone who might satisfy an information need, while the library paradigm is based on accessing a document that might accomplish that goal.

Social search has become more important with the advent of online social networks. Social search is based on the capacity to use information that is provided by people to satisfy the information needs of other people instead of using data selected by authorities and indexed in a catalogue. Social search includes the village paradigm, in which people satisfy the information needs instead of the indexed documents of the library paradigm. Out of these two paradigms, only the library paradigm has been automated online; with social search, the first steps to automate the village paradigm have been made.

In fact, the information search process has been automated from the library paradigm perspective only, while the older paradigm, the village, has been almost not automated at all.

Nevertheless, the village paradigm presents some benefits compared with the library paradigm; people can adapt the answer content as a function of who is requesting the information, or furthermore, people can perform clarifications of the answer content; at the same time, the explanations in documents remain static and are the same for all of the readers, and no clarifications can be performed.

The proliferation of online social networks along with the advances in artificial intelligence allow us to consider village paradigm automation. Agent technology is the design metaphor that I have chosen to automate the village paradigm.

In this thesis, I analyse the village paradigm, seeking the aspects that might be automated; among these aspects, I worked out the *selection of candidates* in unstructured peer-to-peer (P2P) social networks because the social networks are inherently P2P. We consider unstructured architectures because social networks are bottom-up, while structured architectures usually use methods to organise nodes and methods, such as distributed hash tables (DHT), to search content. Moreover, in the literature, there are approximations of unstructured P2P social networks that are applied in web searches, as is the case for 6Search.

Although P2P architectures are considered to be more scalable, in the case of unstructured P2P architectures, the query-routing methods based on Breadth First Search (BFS), such as the case of 6Search, generate large quantities of messages that threaten their scalability in the domain of social webs.

In this thesis, I propose a social search protocol that uses stop messages to improve scalability; to be effective, stop messages require the addition of delays in the propagation of the question messages. Next, I study the effect of social network topologies on the proposed protocol; with this aim, we used 21 social network topologies. We used Movielens data in our simulations. Then, I propose a search algorithm that reduces the number of acquaintances (or contacts) that the question message is sent to. Deciding how many acquaintances to whom the question will be sent is not straightforward; selecting only a few has the risk of the question being dropped, ignored or simply forgotten, and no answer might be received, while selecting too many threatens the scalability. Question Waves (QW), the social search algorithm proposed, contributes to relaxing this decision. The idea of QW comprises delaying the question messages that are addressed to less trusted acquaintances compared to those who are more trusted; in this way, acquaintances with lower trust will be asked later, and more trusted acquaintances will be asked without a delay or with a small delay. With this algorithm, the expectation of obtaining a relevant answer from the requested candidates decays with time, so the algorithm proceeds to send another question message to additional acquaintances after a prudent time, which is indicated by the delay explained before. As a result, not only is scalability improved, but the answers also come sorted by the probability of being relevant; in this way, the algorithm contributes to obtaining more relevant answers in the first answers that are received. The answers are automatically sorted (ranked), and as a consequence, the QW algorithm obtains higher precision in a predefined-size window of answers.

I simulated the QW algorithm over two datasets. The first dataset corresponds to knowledge exchanges (KE) that we generated randomly, and the second dataset

corresponds to opinion exchanges that were built with data from Movielens, a dataset for recommender systems.

Although the work that we conducted is promising, yet not far from applicability, it is far from a complete solution of village paradigm automation. We have opened a new line of research within our research group, which is that we will continue to study, with more emphasis on the applicability, improving the agents' behaviour and the QW algorithm and studying the environment properties.

Resum

Al 2010 Horowitz i Kamvar van explicar que principalment hi ha dos paradigmes per trobar informació rellevant. El paradigma del poble consisteix en preguntar a algú que pugui satisfer la necessitat d'informació, mentre que el de la biblioteca es basa en accedir a algun document que la satisfaci.

La cerca social guanya importància amb la irrupció de les xarxes socials online, i es basa en la capacitat d'emprar la informació proveïda per persones (per satisfer les necessitats d'informació d'altres) enlloc d'utilitzar únicament dades seleccionades per autoritats i indexades en un catàleg. La cerca social inclou el que s'ha descrit com el paradigma del poble, on la gent satisfà les necessitats d'informació d'altres persones, a diferència del paradigma de la biblioteca on són els documents indexats els que ho fan. D'aquests dos paradigmes a Internet, fins ara, només s'ha automatitzat el de la biblioteca, i amb la cerca social s'estan fent els primers passos per l'automatització del paradigma del poble.

El fet és que el procés de cerca d'informació s'ha anat automatitzant, sobretot des de la perspectiva del paradigma de la biblioteca. D'altra banda, el paradigma més antic, el del poble, ha sigut menys automatitzat.

El paradigma del poble presenta alguns beneficis enfront al de la biblioteca, com que les persones poden adaptar el contingut de la resposta en funció de qui tenen davant o fins i tot poden fer aclariments sobre el contingut, mentre el contingut dels textos es manté estàtic.

La proliferació de les xarxes socials, conjuntament amb els avenços en intel·ligència artificial, permeten considerar l'automatització del paradigma del poble. La tecnologia agent és el fil conductor, o la metàfora de disseny que he escollit per a l'automatització del paradigma del poble.

En aquesta tesi analitzo el paradigma del poble per veure quins aspectes són automatitzables. D'aquests aspectes hem treballat la selecció de candidats enfocada en xarxes socials d'igual a igual (P2P) desestructurades, atès que les xarxes socials són inherentment P2P. Considerem l'arquitectura desestructurada, atès que les xarxes

socials són bottom-up i l'arquitectura estructurada sol emprar mètodes per organitzar els nodes, com ara taules de hash distribuït (DHT) per cercar el contingut. A més a més, en la literatura trobem aproximacions de xarxes socials P2P desestructurades de cerca social enfocades a la cerca en el web, com és el cas de 6Search.

Tot i que en principi les arquitectures P2P són més escalables, en el cas dels entorns desestructurats, els mètodes que es basen en el mètode Breadth First Search (BFS), com és el cas de 6Search, generen grans quantitats de missatges, cosa que qüestiona la seva escalabilitat a la dimensió de la web social.

En aquesta tesi proposo un protocol de cerca social que utilitza missatges d'aturada per millorar l'escalabilitat; per què els missatges d'aturada siguin efectius cal introduir retards en la propagació dels missatges de pregunta. Tot seguit estudio l'efecte de les característiques de les topologies de les xarxes socials en el protocol proposat; amb aquest propòsit he utilitzat 21 topologies. S'han utilitzat dades de Movielens per a les simulacions. A continuació proposo un algorisme de cerca que d'entrada redueix el nombre de contactes als que s'envia la pregunta. Decidir a quants contactes se'ls envia la pregunta no és trivial; escollir-ne pocs té el risc que aquests ignorin el missatge i no en rebem resposta, mentre que enviar-lo a massa contactes amenaça l'escalabilitat del sistema. Amb aquest sentit proposo l'algorisme Question Waves (QW) que contribueix a millorar la rellevància de les respostes rebudes en el procés de cerca, a l'hora que relaxa aquesta decisió. La idea principal consisteix en retardar els missatges de pregunta destinats als contactes de menor confiança, de manera que quan menys confiança es tingui cap a un contacte més es triga a preguntar-li, i al revés, no hi ha retràs o aquest és mínim pels usuaris de major confiança. A mesura que passa el temps decau l'esperança que els contactes que han rebut la pregunta puguin proporcionar una resposta rellevant, i per tant l'algorisme procedeix a enviar la pregunta a contactes addicionals rere un temps prudencial definit pel retard. A més de millorar l'escalabilitat, com a resultat del procés de cerca s'obté que les primeres respostes que es reben tenen major probabilitat de ser rellevants, contribuint d'aquesta manera en obtenir més respostes rellevants en les primeres posicions de resposta de forma automàticament ordenada, i en conseqüència amb una precisió major donada una finestra, de mida prefixada, de respostes.

He simulat l'algorisme de QW sobre dos conjunts de dades: el primer que correspondria a intercanvis de coneixement (KE) que he generat aleatòriament; i un segon que correspon a intercanvis d'opinions que s'ha construït mitjançant dades provinents de Movielens, un data set per a sistemes recomanadors.

Malgrat que els resultats obtinguts són prometedors estem lluny d'una solució d'automatització del paradigma del poble completa, però no de l'aplicabilitat. Hem obert una nova línia de recerca en el nostre grup de recerca que continuarem treballant, sobretot des del punt de vista d'aplicabilitat, millorant el comportament dels agents, l'algorisme de QW i estudiant les propietats de l'entorn.

Resumen

En el año 2010 Horowitz y Kamvar explicaron que principalmente hay dos paradigmas para encontrar información relevante. El paradigma del pueblo consiste en preguntar a alguien que pueda satisfacer la necesidad de información, mientras que el paradigma de la biblioteca se basa en acceder a algún documento que satisfaga esta necesidad.

La búsqueda social gana importancia con la irrupción de las redes sociales online, y se basa en la capacidad de usar la información proveída por personas para satisfacer las necesidades de información de otros, en vez de usar únicamente datos seleccionados por autoridades e indexados en un catálogo. La búsqueda social incluye lo que ha sido descrito como paradigma del pueblo, donde la gente satisface las necesidades de información de otras personas, a diferencia del paradigma de la biblioteca donde son los documentos indexados los que lo hacen. De estos dos paradigmas, en Internet hasta ahora solo se ha automatizado el de la biblioteca, y con la búsqueda social se están haciendo los primeros pasos para la automatización del paradigma del pueblo.

El hecho es que el proceso de búsqueda de información se ha ido automatizando, sobretodo desde la perspectiva del paradigma de la biblioteca. Por otro lado, el paradigma más antiguo, el del pueblo ha sido menos automatizado.

El paradigma del pueblo presenta algunos beneficios en comparación al de la biblioteca, como que las personas pueden adaptar el contenido de la respuesta en función de quien tienen delante o hasta pueden hacer aclaraciones sobre el contenido, mientras que el contenido de los textos se mantiene estático.

La proliferación de las redes sociales juntamente con los avances en inteligencia artificial, permiten considerar la automatización del paradigma del pueblo. La tecnología agente es el hilo conductor, o metáfora de diseño, que he elegido para la automatización del paradigma del pueblo.

En esta tesis analizo el paradigma del pueblo para ver qué aspectos son automatizables. De estos aspectos considero la selección de candidatos enfocada a redes sociales de igual a igual (P2P) desestructuradas, debido a que las redes sociales son

inherentemente P2P. Consideramos la arquitectura desestructurada, debido a que las redes sociales son bottom-up, la arquitectura estructurada suele usar métodos para organizar a los nodos como tablas de hash distribuidas (DHT) para buscar el contenido. En la literatura encontramos aproximaciones de redes sociales P2P desestructuradas de búsqueda social enfocadas en la búsqueda en la web, como es el caso de 6Search.

Aunque en principio las arquitecturas P2P son más escalables, en el caso de los entornos desestructurados, los métodos de enrutamiento que se basan en el método Breath First Search (BFS), como es el caso de 6Search, generan gran cantidad de mensajes, cuestionando su escalabilidad en la dimensión de la web social.

En esta tesis propongo un protocolo de búsqueda social que utiliza mensajes de parada para mejorar la escalabilidad, para que los mensajes de parada sean efectivos hace falta introducir retrasos en la propagación de los mensajes de pregunta. A continuación estudio el efecto de las características de las topologías de redes sociales con el protocolo propuesto, con este propósito hemos utilizado 21 topologías. Se han utilizado datos de Movielens para las simulaciones. A continuación propongo un protocolo de búsqueda social que de entrada reduce la cantidad de contactos a los que se envía la pregunta. Decidir a cuantos contactos se envía una pregunta no es tarea fácil, enviarlo a demasiado pocos tiene el riesgo que estos ignoren el mensaje y no se reciba respuesta, mientras enviarlo a demasiados contactos amenaza la escalabilidad del sistema. En este sentido propongo el algoritmo Question Waves (QW) que contribuye a mejorar la relevancia de las respuestas recibidas en el proceso de búsqueda, a la vez que relaja esta decisión. La idea principal consiste en atrasar los mensajes de pregunta destinados a los contactos de menor confianza, de manera que como menos confianza se tenga hacia un contacto más se tardará en preguntarle, y al revés, no hay retraso o este es mínimo para los usuarios de mayor confianza. A medida que pasa el tiempo, decae la esperanza que los contactos que han recibido la pregunta puedan proporcionar una respuesta relevante, por este motivo el algoritmo procede a enviar la pregunta a contactos adicionales tras un tiempo prudencial definido por el retardo. Además de mejorar la escalabilidad, como resultado del proceso de búsqueda se obtiene que las primeras respuestas que se reciben tienen una mayor probabilidad de ser relevantes, contribuyendo de esta manera a obtener más respuestas relevantes en las primeras posiciones de respuesta de manera automáticamente ordenada, y en consecuencia con una mayor precisión dada una ventana prefijada de respuestas

He simulado el algoritmo de QW sobre dos conjuntos de datos, el primero que correspondería a intercambios de conocimiento (KE) que generé aleatoriamente, y el segundo que corresponde a intercambios de opinión que se ha construido usando datos de Movielens, que es un data set para sistemas recomendadores.

Aunque los resultados obtenidos son prometedores, y no estamos lejos de la aplicabilidad; estamos lejos de una solución de automatización del paradigma del pueblo completa. Hemos abierto una nueva línea de investigación en nuestro grupo de investigación en la que seguiremos trabajando, sobretodo en su aplicabilidad, mejorando las prestaciones de los agentes, el algoritmo de QW y estudiando las propiedades del entorno.

1. General Introduction

Although finding relevant information is an ancient problem, this issue has attracted substantially more attention since the dawn of the Internet and Web browsing. There are mainly two paradigms for finding relevant information. They are the village and the library paradigms [Horowitz 2010]. The village paradigm is to ask someone who might satisfy a need for information, while the library paradigm is based on accessing a document that might accomplish that goal. The library paradigm has been the object of several automated systems; the results of using this automation approach are the internet search engines, such as Google.

The usefulness of the library paradigm is not questionable. The village paradigm is not questionable, either, especially with the current advent of online social networks. These two paradigms are complementary; for example, the library paradigm uses answers that are indexed from forums and question-answering sites to satisfy the users' needs. However, not all of the answers can be accessed by indexers (implemented as crawlers in the internet). For example, the case in which the answers have not yet been generated is when new answers must be generated ad hoc or adapted for each question. For this reason, social search, which is related to the village paradigm, becomes relevant.

The advent of online social networks with the advances in artificial intelligence enable the automation (yet partially) of the village paradigm. Social search, thus, has been receiving more attention in the scientific community. Social search is a type of search that uses social interactions to obtain answers or results. Chi [Chi 2009] classified social search systems as social feedback systems (SFS), which resort the results, and social answering systems (SAS), which satisfy the information need using opinions or answers from other users. Notably, SFS is not able to address new questions; otherwise, SAS could manage the requirements, but experts might receive the same question several times. Furthermore, in the case of SAS, answers are not obtained immediately after asking because the answers are generated by the users. We consider question-answering sites that belong to SAS and the (now classical) recommender systems [Montaner 2011] based on the collaborative filtering of SFS. We suggest merging SFS and SAS to obtain the benefits from both systems.

With the purpose of automating the village paradigm, which is eminently social, the intelligent agents appear to be an appropriate technology that might emulate the people's questioning and answering behaviour. Agents use a communication language to communicate among them; thus, agents are **social**. Wooldridge defined agents as follows: "An *agent* is a computer system that is *situated* in some *environment*, and that is capable of *autonomous action* in the environment in order to meet its design objectives" [Wooldridge 2002]. More recently, Hendler [Hendler 2008, Hendler 2010] discusses the need for using social machines in web searches; considering that the concept of a *social machine* is not a consolidated term, agents must belong to it and are a consolidated term and technology that is useful for our purpose. These social machines also must be able to reason about the mental states, including knowledge, of other agents and the humans they represent, due that people dynamic human factors are an important component of human intellectual activity [Galitsky 2013].

From Wooldridge's definition of agents, we can spot key aspects that match with my purpose: **autonomy** indicates that agents have the capacity to make decisions independently, which allows decentralisation of the system. Agents' behaviours are oriented toward taking the initiative to fulfil their objectives, which is known as **proactivity**. Another important aspect is the **environment** and, more concretely, its properties: in a dynamic environment, agents should be **reactive** to be adapted in a timely fashion when the environment changes occur. In an environment that is not reliable, agents must pursue their objectives even though there are failed attempts; this strategy is known as **persistence**. In the same vein, agents must be **flexible enough** to have several alternatives to fulfil their objectives.

Agents allow emulating human behaviour in the village paradigm, and in this way, they automate some aspects; specifically, they can answer frequently asked questions and can perform query-routing.

Agents can represent users (work on behalf of them) and satisfy their information needs. The users are related to one personal agent, to whom they behave as masters or owners (of the agents). The agents are in charge of finding answers to their master's questions; and with that aim, they can use their own knowledge or delegate the task to other agents or ask the question to other people. Agents should decide how to satisfy the information need (autonomy); they can take the initiative to fulfil their objectives (proactivity) and, before a failed attempt, they will take the initiative and try other options (persistency).

Some researchers note the need for a distributed solution of social search for online social networks. Some reasons are to avoid the "big brother" effect [Buchegger 2008], or the entities that manage the centralised tools might have a conflict of interest with the results of the search [Walter 2007].

This doctoral work constitutes mainly a collection of three research papers, called a *compendium* of papers; additionally, there is a survey paper on the research area of social search.

The third chapter of this thesis describes the study [Trias 2013a] in which we investigate the village and the library paradigms, dissecting the village paradigm with the aim of finding the aspects that pertain to easier automation. In the village paradigm, there are three aspects that are more important and easier to automate:

- **Candidates:** This aspect comprises the automation of the selection of the candidates; in some environments, such as in P2P architectures, this aspect includes query-routing mechanisms.
- **Answers:** This aspect constitutes answering automatically. Some of the mechanisms re-use previous answers, as is the case of frequently asked question (FAQ) systems.
- **Propensity:** This aspect determines the predisposition to help others to find answers to their questions. Agents and their owners can be more or less motivated as a function of who is asking, the question domain, if any reward is offered, and other concerns.

We found further village paradigm aspects that, in spite of being relevant, we consider to have less priority than other aspects:

- **Identification of the answer need:** This aspect comprises estimating the information need; this task can be specifying the context, the priority, adding a deadline, or other information.

- **Answer evaluation:** There is a need to specify whether the objective has been fulfilled with the answers that were received, which is why evaluating the answers automatically is required.
- **Learning:** When agents receive an answer, they can learn its content to reuse it when a similar question is received.
- **Rewarding:** Candidates that contributed to satisfying the information need might receive a reward, so that a higher propensity to help them in the future might occur.

In this thesis, we contribute to the Automation of *Candidates*, specifically in query-routing mechanisms in P2P social networks.

As parts of the solution, the 4th chapter describes the Asknext social search protocol [Trias 2012], and the thesis framework is introduced. Each user is represented by a multi-agent System, which comprises a single agent in this thesis; this user is the owner (or master) of the agent in the sense that the agent represents her and works for her. The agents are fed from their owners' knowledge, and their goal is to help their owners in the satisfaction of their information needs, which include the questions provided by the owners and the questions received by their acquaintances. When an agent receives a question, it checks whether it knows the answer, in which case it will answer automatically; otherwise, it can show the question to its owner with the aim that she enters a new answer and/or forwards the question to a subset of her acquaintances. Remember that, in our work, acquaintances are those of the owner. In this way, the protocol combines the SFS and SAS approaches, in the sense that it is possible to reuse previous answers and also to allow the agents' owners to provide new answers.

The answers follow the inverse path of the questions. However, it is possible to send them back directly to the user who generated the question. We believe that using the inverse path presents some advantages, which are detailed in the chapter 4.

Asknext uses two mechanisms to improve the system performance. The first mechanism comprises checking (before forwarding a question) whether the selected candidates belong to previous receivers, in which case the question will not be forwarded to them again. We refer to this mechanism as *Check the Question Thread* (CQT). The second mechanism is the usage of Stop Messages (SM); the main idea of an

SM is to stop question propagation when the information need is satisfied. SM needs the question messages to be slower than the other messages (the answer and stop messages); this reduced speed is accomplished by adding delays to the question messages. In this chapter, we mathematically formalise the relation between periods of sending messages (based on the delay that is applied) of the different types of messages (question, answer, and stop). Additionally, we detail the content of different messages, and the protocol is formalised with the framework proposed by Hussain and Toni [Hussain 2007]. In the following paragraphs, we show the simulation results according to which Asknext reduces the number of messages that are generated compared to Breadth First Search (BFS) algorithm using Time-To-Live (TTL).

Given the good results that were obtained, we wondered whether Asknext scalability was independent of the social network properties. This issue is studied in the 5th chapter, which is paper [Trias 2013b]. Social networks are complex networks in which actors are people and links represents interactions; these networks also have a high clustering coefficient and a low average shortest path length. Furthermore, social networks tend to be assortative. Assortativity is the preference of nodes to link similar nodes, often (and in this thesis) similarity is related to node's degree (number of links); in a assortative network high degree nodes tend to be connected to high degree nodes, and low degree nodes tend to be connected to low degree nodes [Newman 2003a]. In this paper, we generated 21 social networks using the small-world model, and we used simulations to study the properties of the networks that have an effect over the mechanisms of TTL, SM, and CQT; we grouped these 3 mechanisms into 6 algorithms. We observed that the behaviour of Asknext is the most scalable among the behaviours that were studied.

In the 6th chapter, which contains paper [Trias 2013c], the Question Waves (QW) algorithm is presented. The idea behind this algorithm is to delay the question in such a way that the delay is inversely proportionally to the trust of the candidates. The effect produced by this delay inverse relationship to trust (maximum trust means no delay, and minimum trust means the highest delay) is that the first answers to arrive have a higher probability of being relevant; the first results on this approach have appeared in [Trias 2011a] and [Trias 2011b], and in [Trias 2013c], we consolidate the results over two datasets. QW presents other mechanisms for effort distribution in the search process or the method of delaying the answers, to improve the correlation of the relevance of the answers with their arrival order. The experiments that were conducted show that QW obtains a higher recall (which is a theoretical measure of relevance that is widely used in recommender systems and information retrieval) and higher precision (which is a more practical measure of relevance), given the same effort and with a lower number of messages than with the other algorithms.

The 7th chapter comprises a general discussion, the 8th chapter contains the conclusions, and the subsequent chapter contains my personal opinions and perceived value of the work accomplished. The 10th chapter documents some minor errors that were found in the papers, and the last chapter explains our future work.

2. Objectives

The objectives of this thesis are the following:

- To study the social search state-of-the-art and propose a classification.
- To propose the usage of stop messages in social search applied to unstructured P2P social networks and to study its impact, stop messages will need to delay question messages.
- To study the effect of social network topologies in the mechanisms proposed.
- To study the effect of other delay mechanisms applied to social search.

3. Survey of social search from the perspectives of the village paradigm and online social networks

Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva.
Journal of Information Science (in review)

Abstract

Two paradigms currently exist for information search. The first is the library paradigm, which has been largely automated and is the prevailing paradigm in today's web search. The second is the village paradigm, and although it is older than the library paradigm, its automation has not been considered, although certain elements of its key aspects have been automated, as in the case of the Q&A communities or novel services such as Quora. The increasing popularity and availability of online social networks and question-answering communities have encouraged a revisiting of the automation of the village paradigm due to new helpful developments, primarily that people are more connected with their acquaintances on the internet and their contact lists are available. In this survey, we study how the village paradigm is currently partially automated: we consider the selection of candidates for answering questions, answering questions automatically and helping candidates to decide what questions to answer. Other aspects are also considered, such as the automation of a reward system. We conclude that a next step toward the automation of the village paradigm involves intelligent agents that can leverage a P2P (peer-to-peer) social network, which will create new and interesting issues deeply entwined with social networks in the form of information processing by agents in parallel with people.

JIS Pre-Referee Version
Candidate Paper for the Journal of Information Science: http://jis.sagepub.co.uk

Article

Survey of social search from the perspectives of the village paradigm and online social networks

Journal of Information Science
pp. 1-21
© The Author(s) 2013
Reprints and Permissions: sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/016555150000000
jis.sagepub.com
SAGE

Albert Trias i Mansilla
Agents Research Laboratory, TECNIO EASY Center, EEEA, Universitat de Girona

Josep Lluís de la Rosa i Esteva
Agents Research Laboratory, TECNIO EASY Center, EEEA, Universitat de Girona

Abstract
Two paradigms currently exist for information search. The first is the library paradigm, which has been largely automated and is the prevailing paradigm in today's web search. The second is the village paradigm, and although it is older than the library paradigm, its automation has not been considered, although certain elements of its key aspects have been automated, as in the case of the Q&A communities or novel services such as Quora. The increasing popularity and availability of online social networks and question-answering communities have encouraged a revisiting of the automation of the village paradigm due to new helpful developments, primarily that people are more connected with their acquaintances on the internet and their contact lists are available. In this survey, we study how the village paradigm is currently partially automated: we consider the selection of candidates for answering questions, answering questions automatically and helping candidates to decide what questions to answer. Other aspects are also considered, such as the automation of a reward system. We conclude that a next step toward the automation of the village paradigm involves intelligent agents that can leverage a P2P (peer-to-peer) social network, which will create new and interesting issues deeply entwined with social networks in the form of information processing by agents in parallel with people.

Keywords
Village paradigm; library paradigm; social search; automation; review.

1. Introduction

Finding relevant information is a longstanding problem that has attracted much attention in the field of computer science. Today, search is a pervasive functionality and a practical example of the importance of locating information in today's world. Before computerization, there were two approaches to location of information: the village paradigm and the library paradigm. Both paradigms can be computerized via knowledge of their essences. With this aim, we begin this work with a brief explanation of the two paradigms from a non-technological and societal point of view by considering that the current technological solutions bear the risk of overlooking certain important aspects. Following that analysis, we dissect the VP paradigm with the purpose of identifying those aspects that can be further automated. Finally, we propose a taxonomy system for the current state-of-the-art automation of these aspects.

The library paradigm (LP) remains at the core of information searches on the web in which this paradigm has been automated, but several researchers have noted several flaws. This paradigm allows the results to be obtained only as published documents, which must be indexed in a catalogue; this means that it is not possible to obtain results if there

Corresponding author:
Albert Trias i Mansilla, Universitat de Girona. Escola Politècnica Superior, C/ Càtedra Lluís Santaló s/n. Edifici PIV, Lab 001. 17071 Girona, SPAIN.
Email: albert.trias@udg.edu

JIS Pre-Referee Version
Candidate Paper for the Journal of Information Science: http://jis.sagepub.co.uk

is a lack of published content. Furthermore, in the current web, anyone can publish content, which is generally a positive property of the internet but contributes to an overload of available information that causes most web searches to return more documents than the requester can handle. Although the library paradigm offers mechanisms to prioritize certain results over others, these mechanisms are usually not personalized and deliver the same sort process for all users whose individual needs may be slightly different.

Taking into account these flaws, several researchers and enterprises have contributed to the use of the village paradigm (VP) and have automated a few aspects of this model. One clear example is the question and answer portals (Q&A) in which the results to queries formulated as questions are returned in the form of answers provided by real people; these answers may not be published in any document, and the number of answers received is largely reduced compared with the number of answers offered by a web search engine.

The first occurrence of the term VP was found in [1] as recently as the year 2010. With Facebook generating nearly twice the daily traffic of Google, the search and communication spheres are no longer the exclusive domains of search engines based on the LP. Thus, an up-and-coming concept is the *Social Search,* which takes advantage of social networks. This method represents a potential implementation of VP, for which there are several definitions:

1. Social Search is a type of Web-search technique that infers the relevance of Web-search results by considering the opinions of end-users as to the value of Web-content [2].
2. Social search is an umbrella term used to describe search acts that make use of social interactions with others. These interactions may be explicit or implicit, co-located or remote, synchronous or asynchronous [3, 4].

Definition 2 from Evans is more general and includes the socialization of the library paradigm (LP).

Although the use of the terms “village paradigm” and “social search” has become more frequent in recent years, a subset of their aspects surfaced much earlier; an example is Usenet, which dates from 1979 [5]. In this survey, we found many works that potentially contributed to the automation of selected aspects of the VP, yet no single work describes nor summarizes which aspects of the VP can be automated. This paper contributes to the collection of techniques that automate the VP and updates the current state-of-the-art in the automation of the VP search. This work contains a comprehensive collection of papers structured according to three main aspects of this search paradigm, i.e., answer generation, candidate selection, and answer propensity, as well as a discussion of the ongoing research in this area. This survey adds the possibility of intelligent agents for search such that several aspects of the social search, including trust management and query routing, are implemented in a distributed manner via the agent paradigm. Intelligent agents could be a suitable approach for the VP because they share the reactive, social, proactive, and autonomous features of the actors or people in the social networks; furthermore, they are a good match for P2P systems [6], which lie at the heart of the query-routing algorithms. Agents decide on the actions to be performed after receiving a question, are capable of asking questions to other agents, and can take the initiative of requesting knowledge maintenance from their users. Agents can also improve the answering time by appropriately routing questions to experts and reusing previous answers.

1.1. The village paradigm

From the dawn of human history, before writing was developed in approximately 3200 years BC, the main mechanism for information search was what Horowitz et al. [1] refer to as the *village paradigm*. In the VP, knowledge was primarily transferred via oral means or in other words, “people use natural language to ask questions, answers are generated in real-time by anyone in the community and trust is based on intimacy” [1]. In a village, knowledge dissemination was carried out via social means, and a retrieval task consisted of finding the right person to provide the answer. The VP still prevails in offline searches, and the advent of social networks gives momentum to its application online. However, this paradigm suffers from a number of drawbacks [7] for online operation; the most important drawbacks are a lack of answers caused by not having the right people accessible or available to provide the answers, taking too long to deliver the answer (if it was a correct and relevant answer), or having no knowledge available at all (this last problem is common to all search techniques and paradigms). People can be *unavailable* for several reasons, such as they cannot be found, they are not willing to answer, they are not accessible due to small bandwidth, or they did not receive the question. Moreover, with the VP, many people are likely to answer the same question several times with similar, if not the same, answers.

Again, from a historical perspective, with the advent of writing, it became possible to **automate** the answer process because books gathered knowledge that allowed questions to be answered in a manner other than asking the right people; in other words, there were alternative paths to satisfying a need for information. We believe that the main achievement of text was the ability to answer many times the information needs without any additional effort from the answerer (the book writer). Text allows communication to a wider public without placing demands on an expert’s time;

furthermore, it allowed (and still allows) communication with people physically distant from each other. As an example, Diderot wanted to incorporate the entire world's knowledge into the *Encyclopaedia or a Systematic Dictionary of the Sciences, Arts and Crafts*, with the hope that its content would be accessible to the public[1]. However, documents also demonstrate limitations because they are unidirectionally intended (i.e., they are not direct communications), as noted by Socrates in Plato's Phaedrus:

> Written text is basically unresponsive. If you ask a person to explain his or her statement, you can get an explanation; if you ask a text, you get back nothing except the same [8].

According to our point of view, the property of inalterable text has both benefits and drawbacks; a text will not forget its content, but it cannot adapt its content based on the reader. A subset of these differences between texts and people as the providers of answers (in other words, the library and the village paradigms) are collected in popular wisdom as proverbs, e.g., the Chinese saying "a library of books does not equal one good teacher" and the German proverb "a teacher is better than two books" contradict the Dutch saying "teachers die, but books live on".

1.2. The library paradigm

We must first understand the process of automating the library paradigm (LP) before we decide how the VP could be automated. The first libraries were created for proper storage of knowledge and to provide another mechanism for finding information, referred to as the *library paradigm* by Horowitz et al. [1]. A library is a collection organized and maintained for the use of a defined group(s) of people [9]. Traditional libraries are centralized and organized around books, which are considered official and indivisible pieces of knowledge that libraries intend to keep forever. In this model, the authors decide what to write but require the agreement of editors to edit and publish, and finally, librarians decide which books the libraries will acquire [10]. The library institution is an authority that selects these objects, and libraries commonly display new acquisitions in a concrete place, e.g., a set of bookshelves, that allow users to easily browse the new acquisitions.

In the library paradigm, the retrieval task consists of finding the right people and finding the right document. With the aim of aiding users in their search tasks, library organization is indispensable; as an example, Zenodotus, the first librarian of the great library of Alexandria, introduced an organizational system in which the documents were assigned to different rooms based on their subject [9], and in each room, the works were sorted alphabetically by authors' names. Furthermore, Zenodotus added a tag to each scroll with the author's name, the title and the subject [9], but this system was not effective when the collection was growing. Callimachus produced the pinakes, one of the first documents that listed and categorized library holdings. This type of catalogue was useful for determining if the library contained a document by a particular author and where it could be found [9]. Many other types of catalogues have existed in terms of author, title, and keyword catalogues [9]. More "recent" cataloguing systems include the Dewey Decimal Classification[2] and the Library of Congress Classification[3].

Computer science has allowed the automation of certain aspects of the library paradigm. The catalogues or *indices* have been computerized such that it is possible to search documents by author name, document name, keywords, etc. using a computer. This computerization and online availability allowed consultation of the catalogues without physical presence in the library as well as concurrent access by several people at the same time. Furthermore, this method enabled the searching of contents from external libraries, and inter-library loans became possible. The computerized search via an index developed into a topic of active research in computer science known as information retrieval.

With the advent of the internet, the web search has been heavily based on the library paradigm. Web search engines act as a catalogue that returns the results that create the best matches with the keywords input by the user. This cataloguing process is automatic, and one important difference from libraries, in which the documents are first filtered by editors and then by librarians who decide what documents are worth storing, is that on the web, anyone can publish content and the documents are indexed in the catalogues faster than in a traditional library, without taking into account the quality of the manuscript.

For a wider perspective of the history of information retrieval, we recommend the recent work of Sanderson and Croft [11] and certain recent and relevant surveys on information retrieval [12,13].

Thus, information needs were initially satisfied only by the village paradigm. However, when writing was invented, the library paradigm was created, and library growth was recently automated. The use of the library paradigm increased until it was (in our opinion) the main approach applied in most cases. However, the village paradigm has not yet been

[1] http://en.wikipedia.org/wiki/Encyclopédie
[2] http://www.oclc.org/dewey/resources/summaries/deweysummaries.pdf
[3] http://www.loc.gov/aba/cataloging/classification/

abandoned because people use it inadvertently. Answering questions is a deeply human process for which automation has not been sufficiently considered. Certain environments on the internet prefer the community question-and-answer method, and internet forums represent the online reformulation of the village paradigm; one clear example of these preference is Naver, which received the 77% of the searches in South Korea, while Google received 1.7% in 2007 [14]. Forums, in their more basic form, offer a new place to ask and answer questions. The generalization of the use of online social networks encourages a reconsideration of the automation of the village paradigm to not only take into account the peoples' expertise but also consider the information seekers' acquaintances. Online social networks offer new and easy methods to gain access to experts, and this access can be exploited to automate the village paradigm.

Table 1 shows a comparison between the VP and the LP in which the main differences are the information sources and how they are addressed to answer questions.

Table 1. Comparison of the library and village paradigms

	Village Paradigm	**Library Paradigm**
Information Sources (IS)	People [1]	Libraries [1]
IS Selection	Profile, trust, reputation, availability	Library profile, location, scheduling
IS Location	Dynamic	Static
IS willingness to answer	Dynamic [15], give-and-take [8]	Constant [8]
Query preparation	Natural language (NL), politeness	Keyword
Performing the query	NL, social interaction	Catalogue, browsing
IS query understanding	NL knowledge, context	
Result set	NL answers	Set of object titles and location
Information Classification	Own semantic map, Folksonomy	Authority, Catalogue, Manual
Answers	Not reusable, variable [8]	Reusable, constant [8]
Trust	Intimacy [1]	Authority [1]
Search Process	Parallel and iterative, active and passive	Not parallel and iterative, active
Referral	Inform on who can know [16,17]	Inform on a "better" library for IN, object citation
Query Routing	Ask in behalf of the questioner, delegation [16,18]	Ask on behalf of the questioner to other library
Query Evaluation	Social interaction, results	Results

In this paper, we study the automation of the VP. Section 2 dissects the VP in detail and considers which aspects of the paradigm can be automated. Section 3 contains a survey on the current state of these aspects of automation, and Section 4 provides conclusions and final remarks.

2. Dissecting the village paradigm

In this section, we consider the VP process and split it into several aspects. We consider that the users (or their agents) involved in the process can play three roles as stated in [19]:

- Questioner: A questioner is a user who initially requires information and asks others with the aim of satisfying his/her own needs.
- Mediator: A mediator is a user who helps another user to satisfy his/her information need, but the mediator does not contribute with his/her own answer. A mediator can contribute by providing an answer through query routing (asking the question to another user and communicating the answer to the questioner or another mediator, etc.) or providing referrals.
- Answerer: An answerer is a user who provides his/her own answer to a questioner or mediator.

The information search process in the VP can be summarized as follows (Figure 1):

- (1) If a questioner has an information need, she usually attempts to answer with his/her own knowledge; although this process may not satisfy the need, it can also provide relevant information in terms of the domain of the information need, its importance, deadlines, etc.
- (2) After this process, the questioner decides that certain people are suitable **candidates** for satisfying her information need. As information sources, the selected candidates can be chosen for different reasons; the main reason is because they were identified as potential information sources, which would strongly depend on the views of the requester with respect to the information sources (acquaintances profile). This situation is one of the reasons that questioners often select someone with whom s/he interacts as a possible information source when the information need is unsolved; the questioner may have the information need in mind, or the situation retrieves it. Other options include selecting people found in certain spaces, a case in which the selected source might be a complete stranger; for example, when the questioner asks a shop assistant about the features of a product or when the questioner requests directions in a city. As a function of the individual, many reasons may exist for preferring one information source to others; the most rational reason is formulation of heuristics with respect to the probability that the information source can satisfy the information need [20] and how easy it will be to obtain the answer.
  - (2.1) The questioner may be able to communicate with the source to answer the question, which is why the questioner may try to locate the source or will wait for the chance to meet. Locating someone is not always an easy task (more difficult in the past than today); on certain occasions, additional information may be needed, e.g., the location of the possible information source or at what time the questioner can find this source. Moreover, being present at the same location as the potential answerer does not imply that asking is possible because the source might be not available or the occasion may be not appropriate for asking; thus, a long period of time could elapse from selecting a candidate to asking the question.
  - (2.2) Once the information source is located, the questioner performs a few tasks before asking. The questioner will adapt the message in base of the person and the personal situation to increase the chance that the selected source understands the information need and to encourage a willingness to answer, e.g., perhaps an introduction or a conversation will precede the formulation of the question in natural language.
- (3) In this paradigm, the individual variables of the information source (i.e., mood, personality, demographic, culture, and state) may affect his/her willingness to help the questioner with the request; the willingness to answer (**propensivity**) can be influenced by their relationship, reputation (if known), and the assessment of the questioner as to the question itself (i.e., if it is adequate or if it is something that the answerer likes or dislikes). When the information source receives the request, it is possible that s/he could request a clarification [8, 21]; the answer deadline and the importance of the need are possibly communicated as well.
- (4) If the candidate agrees to help the questioner or mediator, she decides on her familiarity with the question. To be able to provide an **answer**, the candidate must understand the question. She would need to understand the language used, and the context may also be relevant. In certain cases, the candidate might be able to answer immediately but in other cases, the candidate might not be able to offer an answer right away because she does not know the answer, does not have time to answer or requires time and effort to prepare an answer. In certain cases, the candidate might contribute by acting as a mediator:
  - (4.1) The mediator may ask her acquaintances to obtain the answer and subsequently relay it to the questioner. Thus, the requested peers that do not offer content can also become valuable contributors if they provide important connections or routes (**query routing**) [16] through which the question can be adapted (reformulated) or if they also ask a complementary question in attempting to complete their answer. The information source that receives the request may continue to route it, and the process becomes iterative. The interest of the mediators may decrease, and presumably, the greater the distance from the initial questioner, the less effort put forth into the search. It is possible that the questioner could obtain better results from query routing than asking the answerer directly, which could be a case in which there was no a proper or accessible relationship or the occurrence of a communication problem, e.g., because they use different languages. When the query is routed to another candidate, the role of the questioner becomes passive for a portion of the process; while she waits for an answer provided by the mediator, she may ask the question to another candidate. In this context, query routing can be viewed as a type of delegation.

  - (4.2) The mediator may provide a referral because she is aware of a source that can satisfy the information need, or in certain cases, the questioner can indicate who referred her to the new candidate with the aim of increasing the chances of obtaining an answer.
- (5) When the questioner receives an answer, she will evaluate whether the information need is satisfied; if so, the search process is finished, but in other cases, the answers may only partially satisfy the information need, or a new aspect may be discovered that must be solved.
- (6) When the information need is solved, the answerer might learn from the process and may be able and willing to help others with the same or similar information needs.
- (7) Furthermore, the questioner may decide to reward those that helped her to satisfy the information need, and the reward can be indirect as an increase of the strength of the tie between the parties.

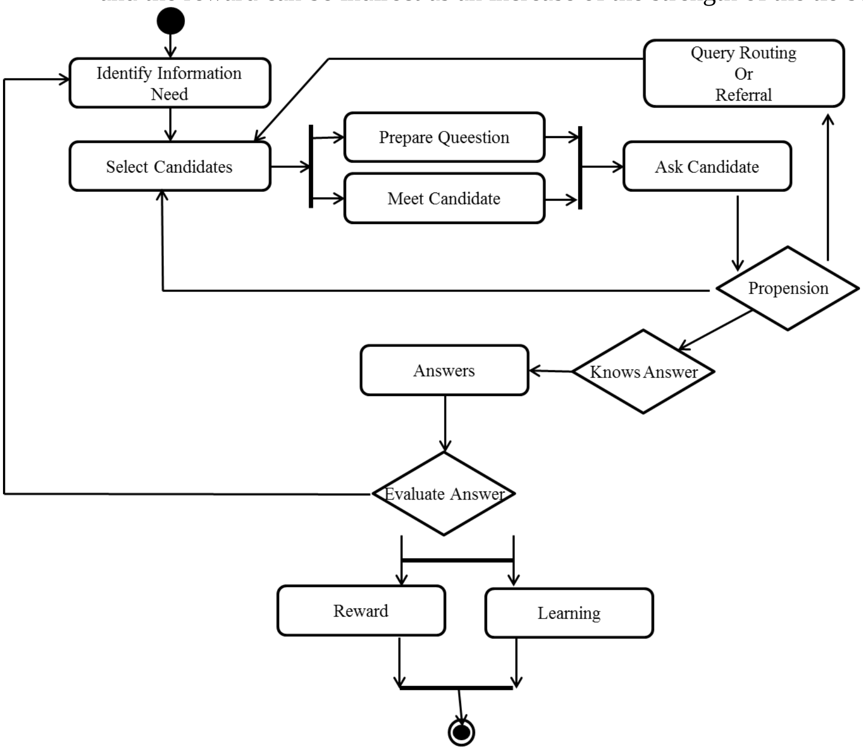


Figure 1. Village paradigm: information search process

People's relationships may be affected within these processes, e.g., the requested people might feel annoyed. In the VP, people expect balanced relationships, e.g., the explanation [22] that the *social exchange* theory assumes that people attempt to maintain balanced relationships (reciprocity) because people prefer relationships in which they give and receive a similar amount of support. However, this process can improve the relationships if the parties share certain interests, e.g., in the work of Yang [15], the researchers showed that a motivation to ask and answer acquaintances is keeping in touch and to be connected.

From our point of view, although this process does belong to social search, the socialization of the LP approaches does not belong to the village paradigm. The works in which the library paradigm has been socialized display two primary trends. The first is the cooperative search, which consists of tools that aid a group of people who search together. Examples of this approach are the S3 [23], coSearch [24], and Heystacks [25] methods. The second trend consists of recommending results based on other users opinions; one clear example is the Collaborative Filtering recommender systems [26].

From the above explanation, we consider the information search process in the village paradigm to contain three main aspects (although others exist that we believe to have lesser importance) from the point of view of automation. These main aspects are:

- **Candidates**: This aspect consists of selecting and accessing the information sources and includes finding a formulation for the question. Because the candidates can contribute to routing the query or answer using referrals, these aspects are also included.

- **Propensity**: This aspect consists of the process through which a candidate decides to help (or not help) the questioner/mediator.
- **Answer**: The aspect represents the process in which the selected candidate provides her own answer to the questioner/mediator.

Other interesting aspects include:

- **Identify the information need:** This aspect consists of defining the information need and may include factors such as deadlines, the importance of the information need, etc.
- **Answer Evaluation:** The answerer/mediator considers if the answer received is of sufficient quality. It is possible that the answer partially satisfies the information need, and as a result, the information need may be reformulated.
- **Learning:** The candidates and the questioners who receive an answer may learn from the question subject.
- **Rewarding:** A candidate who helps a questioner or a mediator may obtain a reward.

2.1. What can be automated?

As previously stated, the library paradigm (LP) is the best option for a search that outperforms the village paradigm (VP). This method allows **reuse** of content, **reduction** of the time for answering, and the use of catalogues (keyword-based) to reduce the problem of finding the right people, and these processes have been largely automated. Moreover, the VP is still in use, and advances in technology have enabled the improvement of this automation.

Technology (concrete communication tools) has reduced a subset of the problems of the VP because it is easier to locate and access possible sources of whom to ask questions. Mobile phones, emails, instant messaging or online social networks enable contact between people who are not located in the same place at the same time, and thus, there is no need to know the location and schedule of the recipient. Furthermore, several aspects that can be automated have only been partially automated in recent social search environments. From consideration of the three main aspects:

- **Candidates:** It is possible to recommend answerer candidates to a questioner by allowing that the questioner only queries a (search) system, and the system may suggest someone out of the questioner's social network, or instead, the system may ask those contacts directly.
- **Answers:** Second, when the system receives a question, it can reuse previous answers, and thus it can operate using a centralized perspective in which all of the previous answers are available in the system or using a P2P system in which each answerer owns her own knowledge base, which is only used when the question goes through him/her. A system may be able to generate new answers; in this aspect, technology and the current state-of-the-art research is not developed in most contexts, but as examples, Wolfram Alpha and Google are able to solve mathematical expressions or convert the value of one currency to another, and modern data-mining and opinion-mining techniques are able to generate answers, though in a limited scope.
- **Propensity:** It is also possible to predict the answer's willingness and prompt the user to answer those questions with a higher estimated value.

Other aspects that may be considered once the three main aspects are solved include:

- **Identification of the information need:** This aspect is not at all automatable at present; furthermore, certain examples in the literature automatically take context into account in the users' queries.
- **Answer evaluation:** Several heuristics have been used to evaluate answers. Furthermore, Agicthein and Liu [27] studied the questioners' answer satisfaction and found that humans have difficulty in predicting the questioner's satisfaction and that there are classification algorithms that are able to outperform human classifiers.
- **Learning:** Automation can be achieved by storing the received answers with the intent of reuse, but automatically deciding which answers can be reused is not a straightforward process. Additionally, it is possible to learn about the social network or the users' answering behavior.
- **Rewarding.** Many methods for rewards exist in the village paradigm, although a subset is indirect. Giving out points or payments for suitable answers are techniques that have been extensively used in the Q&A communities, whereas social network approaches have also increased trust values that (in certain environments) increase willingness to help the answerer in the future.

We consider that the main problem of the village paradigm is the bottleneck created by the users' time; erratic answering time as the most important obstacle, which may tend to be quite long, needs to be reliably short and must be reduced to the minimum to meet the requirement of many of the answerers' time horizon. Furthermore, the selection of candidates and the propensity may save significant time for the user. In the following section, we review the literature

JIS Pre-Referee Version
Candidate Paper for the Journal of Information Science: http://jis.sagepub.co.uk

on automation of these aspects; we focus mainly on candidates, answers and propensity, although other aspects are also briefly considered.

3. Survey on social search from the point of view of the village paradigm

In this section, we describe surveys on the automation of the VP aspects. We begin by reviewing the current classifications in Social Search, which currently acts as an umbrella term to describe search actions that make use of social interactions with others.

Chi [28] proposes the following classification of social search systems:

- 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 use people's expertise or opinions to answer questions in a particular domain, and the answerers can take the form of friends, colleagues, or strangers.

We assume that knowledge exchange portals (Q&A) fall into the Social Answering Systems category, according to Chi's classification. Analogously, we consider Recommender Systems that use social data (commonly based on Collaborative Filtering and Demographic filtering) to fall into the Social Feedback Systems category. We can imagine a recommender system for all uses, which recommends web pages as a function of the query and the user because it is equivalent to a search engine with personalized results. Typical drawbacks of this category are that problems in storing the user-items matrix may occur as a result of the recommendation technique, and the matrix may also be sparse.

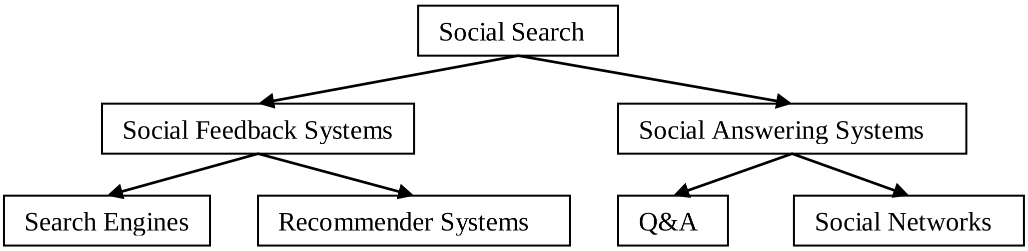


Figure 2. Social search taxonomy, from updated [19]

According to Chi's classification, Social Feedback Systems lack the ability to address new questions if the information is not available online; although Social Answering Systems might solve this problem, the asked experts can be requested to answer the same question multiple times and the questioners do not receive the content immediately, as Morris noted, because the response times on Q&A sites tend to be long [29].

As stated, we consider the Recommender Systems and the Social Search Engines to be a type of Social Feedback System, whereas we consider the Q&A portals and Online Social Networks as Social Answering Systems.

In another classification proposed by Liu, three options are available to provide questioners with good results by minimizing the waiting time in Q&A [30]: a first option is the reuse of existing content in the Q&A [31, 32], a second option consists of routing questions to experts on the topic of the question [33, 34, 35], and the third option applies an attempt to understand the answering behavior to encourage additional answers [30,36]. We note a degree of analogy to the main aspects described. Liu's first option directly corresponds to the automation of answers, the second option corresponds to candidate automation and the third corresponds to the study of *propensity*. Although there is a high degree of similarity in the concepts, certain slight differences exist compared with the work of Liu, which is focused on Q&A portals, and propensity automation is included inside the candidates category because it is possible to perform these two tasks in a centralized approach. In our point of view, the candidates only take into account why the questioner (or mediator) would consider sending a question to a possible answerer, whereas in the propensity, a type of filtering and resorting of the questions received occurs before the candidate acts as answerer or mediator. Table 2 presents a selection of works that automate selected portions of the three main aspects of the LP.

Table 2. Selection of works that automate an aspect of the VP. Works in bold were classified by Liu in [30], and those in plain text denote our update

	Answers	Candidates	Propensity
Jeon [32]	✓		
Cao [37]	✓		
Cao [38]	✓		
Bian [31]	✓		
Suryanto [39]	✓		
Shtok [40]	✓		
Guo [41]		✓	
Qu [42]		✓	✓
M. Liu [43]		✓	✓
Horowitz [1]		✓	
Zhou [35]		✓	
Nam [44]			✓
Q. Liu [30]			✓
Kabutoya [46]			✓
Walter [47]	✓	✓	
SoPPSoN [16]		✓	
Sixearch [18]	✓	✓	
Question Waves [48]		✓	✓
Asknext [19]		✓	
ASKS[4]	✓	✓	
PROTAGE [49]	✓	✓	
MARS [17]		✓	
Opinion Filtering [50]	✓	✓	
Collaborative Filtering [51]	✓	✓	
De Souza [52]		✓	
Wenyin [53]	✓	✓	✓
Krulwich [54]		✓	
Krulwich [55]		✓	
Simmons [56]	✓		
Sneiders [57]	✓		
Quextit[5]		✓	
Google answers[6]		✓	
Askville[7]		✓	
InfoRocket[8]		✓	
Quora[9]		✓	
Yahoo! Answers[10]		✓	
Hsieh [58]			✓

[4] http://www.asks.cat
[5] http://www.quextit.com
[6] http://en.wikipedia.org/wiki/Google_Answers
[7] http://askville.amazon.com/
[8] http://www.startupzone.com/list2/startupcategory/WC1110StartupDetail.asp?seq=630
[9] http://www.quora.com
[10] http://answers.yahoo.com/

 © The Author(s), 2013

The following subsection contains a study of the main trends in the automation of each of these *aspects*. We begin with *answers* automation, followed by *candidates* automation, and as a continuation, we consider the *propensity* automation. Finally, we address the automation of the other aspects in the form of *identifying the information need*, *answer evaluation*, *learning*, and *rewarding*.

3.1. Answers – answer automation: reusing existing content

Techniques for reusing existing content are related to the state-of-the-art in Information Retrieval, and this area has been largely automated by the library paradigm, although it plays an important role for this component of the village paradigm. Thus, we summarize the recent approaches to this aspect that also contribute to the goal of the village paradigm.

Jeon et al. [32] proposed that any search should assess whether a question was previously asked; in that case, the answers associated with the question in the past can be quickly returned. Identifying question similarity is not straightforward because questions with similar meanings can appear completely different lexically. These researchers conclude that using the question title to identify the question similarity provides better results than using the question body.

Jeon et al. assumed that there are three main approaches to compute question similarity: the first is the use of a knowledge database, the second is the use of manual rules or templates, and the last approach is the use of statistical techniques.

Cao et al. [38] proposed recommending other questions when a question is queried. They considered a question to be a combination of a topic and a focus and proposed to represent questions in terms of a graph of topic terms and a ranking of the question candidates. With the aim of representing the question as a graph, they used a tree cut model and minimum description length to obtain a graph of topic terms and subsequently determined which of these terms could be substituted.

Shtok et al. [40] proposed the filtering of questions with similar titles that have an answer marked as a "best answer"/rated with three or more stars, following this measure of the similarity with the title and the body of the question. The last step of their approach consists of using a classifier to validate that the answer from the selected similar question is able answer the new question.

Wenyin et al. [53] provided automatic answers using previous answers to similar or repeated questions. Certain authors proposed the use of P2P systems in which each user has a knowledge base that it queries with the aim of reusing its content [19, 49].

As shown in Section 3, certain recommender systems belong to Social Feedback Systems. In recommender systems, the answer is traditionally an item (or a sorted list of items) that has been included in the system, and the information is given for the answer. Recommender systems are able to answer with a suitable item for a user within a given category [47]; the techniques used to select one item or another are based on the ratings of those items by similar users, which can be filtered by similar ratings [51] or similar demographic profiles [54]. Recommender systems usually display a limited domain in which this answer automation via selecting a category is more reliable, and the type of questions addressed usually belong to the type of "which element in that category do you recommend to me?"

Furthermore, many efforts have been made to ensure that automatic systems are able to answer the users' questions in natural language. One of these examples is the Text Retrieval Conference (TREC). An automatic system that aims to answer questions must analyze the question [59], and the answer that is obtained through a document repository can be extracted or generated from the documents. As stated by [59], automatic answers are not new; a survey was published in 1965 with at least 15 systems in which computers answered questions [56].

Sneiders [57] presented an email answering system that is able to answer frequent questions. From all possible answers, this system evaluates two patterns within the question: the required pattern determines whether the question contains a text portion that enables a possible answer, and if a forbidden pattern is matched, the answer is discarded.

Nishimura et al. [60] reports a QA system that automates answering based on a mailing list that is used to generate a knowledge base.

3.2. Candidates - candidate automation: routing questions to suitable candidates

Selecting a subset of users as answerer candidates is a crucial aspect of village paradigm systems, as expressed by Liu [43]:

> A user who knows well the answer to a particular question may not answer the question because the user may not visit the system frequently or the user may be faced with many questions…. It is a natural idea to push new

questions to the right persons in a Q&A system to obtain quick, high-quality answers. The reduced waiting time and improvements in the quality of answers are expected to improve users satisfaction.

We consider that candidate automation contains two important aspects. The first aspect indicates how the suitable candidates are selected and can take into account candidate expertise and similarity with the information seeker, among other factors. The second aspect indicates how the candidates can be queried; certain systems assume that all users can be requested, whereas others assume that a user may only request information directly from her acquaintances. In Subsection 3.2.1, we present the different approaches for selecting the candidates, and in Subsection 3.2.2, we cover different query routing models.

3.2.1 Selecting suitable candidates

We have organized the following criteria to classify the different works based on the method used to select the suitable candidates:

- Expertise: The system attempts to evaluate the degree of user expertise for each domain, which focuses on their ability to produce correct answers in a given domain [17, 54].
- Similar tastes: User similarity can be computed in many ways, such as Pearson's coefficient [61, 62] and cosine [63, 64], the traditional methods used in collaborative filtering (a type of social feedback system), although more recent measures have been introduced, such as the PIP similarity [65].
- Demographic profile: The candidates are selected based on features such as age, gender, nationality, lifestyle, purchasing history, etc. [54].
- Trust: Trust has been studied in several disciplines, including psychology, sociology and economics [66]. Rousseau et al. define trust as: "A psychological state comprising the intention to accept vulnerability based on positive expectations of the intentions or behaviors of another" [67]. Selected interesting surveys on trust are those of [66, 68].
- Sociability [17]: Sociability is the ability to produce accurate referrals [17] or to accurately forward the query.
- Activity: The answering system takes into account how often and when the user utilizes the service to answer questions [43].
- Authority: Authoritative users are most likely to provide authoritative answers to new questions [43]. Fang et al. note that authoritative users represent a small proportion of active users and also claim that the "authority" consists of the combination of answer quality and quantity [69].
- Answerer interests: If an answerer provides the best answer to a question, this may equally indicate that the answerer has a particular interest in the question or that she might be an expert in the domain [43].

Table 3. Candidate selection criteria

	Expertise	Similarity	Demographic profile	Trust	Sociability	Activity	Answerer interest	Authority
Horowitz [1]	✓					✓		
MARS [17]	✓				✓			
6Search [18]								
PROTAGE [49]	✓		✓	✓	✓			
ASKS	✓			✓				
Google Answers	✓							
Quextit			✓				✓	
Walter [47]				✓				
Collaborative Filtering [51]		✓						
Opinion Filtering [50]		✓		✓				
SoPPSoN [16]				✓	✓	✓		

JIS Pre-Referee Version
Candidate Paper for the Journal of Information Science: http://jis.sagepub.co.uk

Guo [41]						
Qu [71]					✓	
M. Liu [43]				✓	✓	✓
Zhou [35]	✓					
Wenyin [52]			✓		✓	✓
Krulwich [55]	✓	✓				
Krulwich [54]	✓	✓				

Liu et al. [43] proposed a probabilistic method to select the top-k best answerer candidates by considering the answerers' interests, the users' authority and the users' activity. This group modeled the answerers' interest by considering the questions that the users had best answered and used the Latent Dirichlet Allocation [70] and Language Model to predict the answerers' interest in a question. The users' authority was modeled using the number of best answers provided, and the users' activity was modeled using the current question posting time and the time of the most recent answer provided by the answerer.

Qu et al. [71] recommended questions to the top-k users that may have interest in the question, and the goal is to identify the questions topically; for this purpose, they used the PLSA model [72].

Liu et al. [45] studied the effect of what topics the answerers were web browsing when the answer to their question was received. The concept is that candidates browsing on the given question topic may have greater willingness to answer. The users' web browsing data can be accessed via toolbars.

In MARS [17], the research group decided to split the information on the contacts into two models: expertise (ability to produce correct domain answers) and sociability (ability to produce accurate referrals). Furthermore, they distinguished between the interest and expertise of each user. A user will query others with respect to their interest but will answer the queries of others based on expertise. This work used the vector space model (VSM) to allocate people. The VSM estimates the importance of each term in a query and the term's power of discrimination among the users to whom that query may be sent based on the expertise exhibited. By comparing the query vector with the expertise vectors of other users, they found the users whose expertise was the most similar to the query. The sociability of an agent reflects its ability to produce good referrals. Each agent evaluates others based on a linear combination of their expertise and sociability.

The work of Krulwich and Burkey [55] extracted user expertise from documents and answered questions with a referral.

Fang et al. [69] recommended selection of the top-k candidates by considering the relevance between the user and the query as well as the user willingness to answer.

Zhou et al. [35] proposed three approaches to compute user expertise with the aim of finding the top-k candidates in a forum based on expertise. The first approach is the profile-based model, which consists of building a user profile using the answered questions and the related answers with the aim of computing the probability of the user as an expert for a question. The second approach is the thread-based model, in which the authors build a user profile for each thread; the authors compute the probability of an expert based on threads and relevant users. The last approach is the cluster-based model, which generates clusters of threads in which each cluster represents a topic and users are ranked based on all clusters.

Aardvark [1] selected candidates based on expertise and availability and also took into account the questioner's social network. Aardvark allowed users to specify their domains of expertise and when they can be requested; this system is also able to contact users using IM and email.

Yan et al. [73] used Latent Dirichlet Allocation to identify the latent topics of questions and subsequently modeled the relationships among the questioner, the question and the answerer, which indicated the answers provided by an answerer to a questioner under a determined topic with the aim of predicting the top-k candidates.

In 6S [18], queries can only be routed to known peers, and nodes store the profile of each known peer. When a node must route a query, the known peers are ranked by comparing the query with the profiles. The profiles are based on a vector space representation.

In PROTAGE, the candidates are selected by trust, demographic data, or by a combination of both.

3.2.2 Query routing methods

The sources that can be asked to answer a question depend on the model upon which the system is based. Certain methods assume that all users can be requested (centralized approach), whereas others (P2P approach) assume that only the users' contacts can be accessed. A subset of these methods includes the use of referrals that allow expansion of the users' contacts. Finally, other methods allow candidates to forward the question to a subset of their acquaintances.

Table 4. Query routing methods

	All users can be requested	Contacts	Referrals	Forward query
Askville	✓			
ASKS	✓			
InfoRocket	✓			
MARS [17]		✓	✓	
Krulwich [55]		✓	✓	
PROTAGE [49]		✓		✓
Quextit	✓			
Quora	✓			
SoPPSoN [16]		✓		✓
Yahoo! Answers	✓			
6Search [18]		✓		✓
Walter [47]		✓		✓
Collaborative Filtering [51]	✓			
Opinion Filtering [50]		✓		
Guo [41]	✓			
Qu [42]	✓			
M. Liu [43]	✓			
Horowitz [1]	✓			
Zhou [35]	✓			
Wenyin [53]	✓			

Currently, the Q&A portals (such as the trendy Quora, Yahoo! Answers, or the extinct Aardvark and Google Answers from Google) are centralized and allow any user in the community to answer a question, although they might take into account the social network, as Aardvark did, or might allow suggestion of another answerer, as Aardvark also did [1].

However, those authors emphasize the scalability issues of the information search system, and many of them back P2P systems as the state-of-the-art for social search, as is the case for MARS [17], Walter et al. [47], 6Search [18], PROTAGE [49], Asknext [19], Question Waves [48], and SoPPSoN [16]. Among these, systems such as MARS allow a request for contacts only, and candidates can answer with a referral, which may be requested. Others such as Walter et al., 6Search, PROTAGE, Asknext, Question Waves and SoPPSoN allow the candidates to forward the question. The P2P systems that allow forwarding of questions to many contacts are multicast models [74] that fit well a priori with the needs of social search. Multicast systems should use selected techniques to sort out the threat of their scalability. A subset of these techniques include the use of TTL [18, 75], stop messages [19], check-the-question-thread [19], the use of several attempts to obtain answers [48], random walks [76], query routing with ants [77] and Branching Factor [17].

3.3. Propensity - propensity automation

The propensity is the process of recommending a query to be answered in such a way that a candidate will be willing to provide an answer. In several cases, this category overlaps with the CANDIDATES process, e.g., in the Q&A portals in which it is not clear when a user receives a recommendation to answer a question or from what perspective this process takes place.

The good news is that this overlap is minimized when P2P systems play a role in the social answering systems; in the candidate process, the questioners or mediators decide who requests the query because in the propensity, the "candidate" is the one who decides whether s/he wants to answer.

In this subsection, we classify works that consider the propensity using the following criteria:

- Expertise: Expertise determines whether the requested user has sufficient knowledge to answer the question. Self-confidence can also be taken into account.
- Interest: Users are only interested in a few question categories [78]. Their interest can be viewed as a desire to learn from that category [44].
- Who requests (requester): The propensity to answer will change depending on who is asking. People are often more motivated to help friends than complete strangers [79]. At times, people are motivated to improve a relationship for future benefit or as reciprocity behavior.
- Effort: The effort that the candidate must put forth to answer the question may be a demotivating factor.
- Reward: A promised reward may motivate the candidate. The rewards might consist of points, virtual currencies or an increased probability that others will help in the future.
- Answerer context: Liu et al. [43] consider that the candidate's actions at the moment when the question is received impact his/her willingness to answer.

The results of the classification are shown in Table 5. Nam et al. [44] also found that altruism encourages answerers. However, we do not believe that this aspect can be used to filter the queries to a specific user from a propensity perspective if we consider the altruism to be equal for all queries and questioners/mediators.

Table 5. Propensity aspects

	Expertise	Interest	Requester	Effort	Reward	Answerer Context
Qu [42]	✓	✓				
M. Liu [43]		✓				
Nam 2009 [44]		✓				
Liu 2010 [45]						✓
Question Waves [48]	✓	✓	✓			
Kabutoya [46]	✓	✓				
Hsieh [58]				✓	✓	
Wenyin [53]					✓	

In experiments with Question Waves [48], Trias and de la Rosa considered that the answer quality could be affected by expertise, implication, and external factors. Implication is explained as the willingness to contribute due to the subject and the requester. These aspects can also have an effect on the propensity.

According to Fiske [80], there are four essential styles of human relationships which determine the way people behave with others: communal sharing (CS), authority ranking (AR), equality matching (EM) and market pricing (MP) [81].

In the CS group, the relationships consist of people who contribute because they belong to a group. Aspects such as altruism belong to this category. In the AR, a hierarchy exists and people in positions under others carry out the requests made by those in higher positions. In the EM, the relationships are based on reciprocity. In an EM relationship, the contributions are due an expectation of reciprocity, reward, or association. In MP, the contribution is evaluated financially. Contributors expect a tangible reward, and providing knowledge has a cost.

3.4. Other aspects

In this section, we explain certain other aspects that maybe automated, although we assume that these aspects require the previous automation of the main aspects.

3.4.1 Identifying the information need

Context in the user search process has been studied with the library paradigm. One clear example is query disambiguation in which the same query could refer to several different concepts [82], stated as "in human language and communication, context serves to disambiguate meaning" [83]. In this case, knowledge of the domain might reduce the ambiguity problem. The context in web searches can contain aspects such as previous queries or results selected [82].

[84] attempted to extract contextual information from the task the users perform, such as terms on a web page she is reading or in an opened document.

3.4.2 Answer evaluation

Answer evaluation is not only an interesting aspect for answer automation but is also necessary to determine if added effort is required to search for additional answers or if the received answers are satisfactory. Several heuristics can be used to evaluate the answers, including trust and reputation. How to evaluate answers is context dependent.

3.4.3 Learning

Learning in the village paradigm is related to other aspects such as Answers, Candidates, and Propensity. The Answers are related to learning because the new answers may be reused [32, 53]. In the case of Candidates, this aspect improves the information on the candidates with the aim of routing questions with a higher efficiency, which could be carried out using trust, e.g., [16, 17, 48]. In the case of Propensity, it is possible to improve the selection of questions recommended to a user for answer, e.g., by learning the users' interests [42]. Furthermore, learning is a tool that may be used to adapt the solution to changes in the environment.

3.4.4 Reward

Knowledge exchanges have been analyzed using the exchange theory, an economics theory that assumes that individuals pursue the options that provide the highest benefit at the lowest cost [85]. Moreover, sharing of knowledge has a cost, and contributors expect satisfactory compensation [86], even when their contributions are automated on their behalf.

Market-based institutions offer promising mechanisms to evaluate knowledge and to design incentives to motivate and support valuable knowledge transfer [87]. These rewards can be classified as hard, a category that includes more tangible items such as economic gain, career opportunities, information and knowledge, or soft, a category that includes aspects such as increasing reputation and personal satisfaction [85]. Furthermore, [81] explained that in certain cases, contributors expect reciprocity or a reward.

The Q&A literature contains several examples in which reward has been considered. In general, we identify four approaches:

- Use of a legal currency: Users who answer receive legal currency.
- Use of a non-legal currency: Users who answer receive a non-legal currency that they can exchange for certain services, such as asking a question in the portal.
- Reciprocity: Users who answer gain a higher probability of receiving answers to their questions.
- Reputation: Certain approaches give out reputation points that illustrate how much the users contribute.

Table 6 lists several works and the related reward approaches.

Table 6. Reward approaches

	Legal currency	Non-legal currency	Reciprocity	Reputation
InfoRocket	✓			
Yahoo! Answers		✓		✓
Google Answers	✓			
Askville		✓		✓
Carrillo [88]		✓		
Moreno [89]		✓		
Quickerland[11]	✓			

[11] http://www.quickerland.com

SoPPSON [16]		✓	
Question Waves [48]		✓	
Kamvar [90]		✓	
Quora[12]	✓		✓

In InfoRocket, the questioner posts a reward (that consists of a legal currency amount) with the question, and the different answer providers may obtain these rewards if chosen by the questioner [21,89]

In Yahoo! Answers, users obtain points by answering questions, and posting a question has a cost. Certain users seem motivated by the reputation gained by owning many points. [89]. Quora uses the "Top Writers program on Quora", which includes people with high contributions in Quora. These people appear in a list that provides them with reputation, and furthermore, they obtain other rewards, e.g., a t-shirt. Furthermore, Quora uses Quora credits[13]; each user starts with a budget of 500 credits that are refreshed regularly (to allow users can keep up putting questions forward), posting a question has a cost (that varies from 50 credits to 0 credits), and users obtain new credits if people follow their questions or vote positively on their answers. Moreover, users can promote their contributions using the credits.

In Google Answers, each question has an associated price per answer, and the published answers only appear after paying a fee [89]. Quickerland also pays experts for answers, and the amount that a user pays for answers depends on the expertise of the expert and the deadline required for an answer.

In Askville, users obtain experience points in each topic based on their answer quality, and this experience can be viewed as a case of reputation. Furthermore, the users can obtain "quest gold coins" with participation that can be used to obtain Amazon products [89].

Carrillo et. al.[88] and Moreno et al. [89] assumed that users might obtain points from their previous content that are accessed.

Kamvar [90] proposed that the bandwidth in a P2P system could be distributed based on the user contributions.

4. Final remarks

In this survey, we have presented the evolution of the area of information search. Information search was initially based on the VP and with the creation of libraries, was followed by the library paradigm, which consists of using a catalogue to find the right document to satisfy an information need formulated with keywords instead of a question.
Several aspects of the LP have been automated by computer science, and an area known as Information Retrieval has been devoted to this topic. The ubiquitous web search has been largely based on this paradigm. However, although the library paradigm is widely used, the village paradigm has not been abandoned and has remained nearly completely manually implemented (with low automation). The popularity of online social networks has pushed the VP towards new levels of automation.

In Section 2, we presented several aspects of the village paradigm. We consider the following main aspects: the first is *candidates*, which consists of selecting a set of possible people as information sources and also includes asking the question, finding the right moment for asking, and preparing the query. The candidates might answer with their own knowledge, but they could also route the query and provide an answer from another information source. The second aspect is the *answers* process, which allows the candidate to provide an answer to the questioner or a mediator. The third aspect affects the candidate's willingness to answer because although she may be able to answer, she might not want to contribute. We refer to this willingness as *propensity*, which can be affected by the relationship with the requester and individual factors of the candidate, such as state and mood.

In the automation of answers, Jeon et al. [32] searched for similar questions with the aim of using the most satisfactory answers. In a similar vein, Cao et al. [38] recommended other similar questions to the questioner, and Shtok et al. [40] not only searched similar questions, they also validated whether the answer was adequate for the new question. Other approaches to promote automation are based on existing content, rely on the information retrieval state-of-the-art, and have been used to automate the library paradigm.

[12] http://www.quora.com/blog/Announcing-Top-Writers-on-Quora

[13] http://www.quora.com/Quora-Credits/What-are-Quora-Credits

JIS Pre-Referee Version
Candidate Paper for the Journal of Information Science: http://jis.sagepub.co.uk

Two important sub-aspects exist in the automation of candidates. The first involves how to select the candidates, and the literature contains works that have attempted to evaluate the candidate's expertise, interests, authority, sociability or activity whereas other works have considered candidates with similar tastes, with a specific demographic profile, or those that are trusted by the questioner. The second sub-aspect focuses on how the candidate is requested; certain centralized systems exist in most Q&A communities in which candidates are selected from all users whereas other environments consider a P2P architecture in which the answerers must exist on a path in the questioner's social network. In this case, the questioners use the candidates as their contacts, but they can forward the query to their acquaintances or provide a referral.

The last main aspect that we consider in automating the village paradigm is the automation of propensity. Certain works have considered that the propensity is affected by user altruism, participation, knowledge, effort, reward and who is asking.

In all, we assumed that the candidate, the propensity, and the answer aspects of the VP can be automated, and these topics are included in the scope of this paper. As stated in Section 3.4, other aspects remain to be automated (Rewarding, Learning, Identifying the information need, and Answer Evaluation) and will be the topics for an extended survey in the future.

In the automation of the village paradigm, we consider an online social network in which each user is represented by a multi-agent system consisting of at least one agent. For the sake of simplicity, we assume that each of these multi-agents systems is represented by agents, and each user will interact directly with her agent. The agents act as servants to their owners in the information search environment, and each of those agents contains the following:

- Ties with the acquaintances of their owners;
- A knowledge base that will be fed by her owner's knowledge and the knowledge obtained from interactions with other agents;
- A trust model to evaluate the performance of other agents;
- A set of "solitary being" objectives;
- A set of "social being" objectives;
- A set of resources;

We believe that the most natural approach to deploying agents for social search in the VP is one in which each user owns an agent (or a multi-agent system) that works on her behalf to answer questions. We also believe that a bottom-up P2P architecture is suitable for question answering in the sense that communities are generated from individuals who all work as clients and servers. This approach seems the most natural, as expressed by Buchegger and Datta [91]. Because P2P allows the use of different pieces of software that can communicate with others who share common protocols, this structure will aid in setting up personalization because the agents of each user can be different.

James Hendler recently expressed the situation as follows [92]:

> Social-networking sites are replacing search engines and news sites as people's favoured homepages and a new generation of applications is centring on large groups of people collaborating over an increasingly distributed network. I won't try to predict the future of these waves [...] but one thing is for sure: many of the entrenched models of AI will have to be rethought as computers change from application devices to social machines.

Agents represent one of these types of social machines due to their social properties, which stand above other hard and weak agency properties such as proactivity, self-awareness and rationality, described by the last Agentlink roadmap [6].

Intelligent agents have enabled support representation, 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 improved relationships between suppliers and consumers of knowledge to provide them better control of their interactions [93]. Therefore, agents are a social and suitable approach for the automation of social networks and a good match for P2P systems. The roadmap of the Automation of the Village Paradigm could be accomplished via unstructured P2P Social Networks and Agents.

Acknowledgements

We thank Àngels Merino, the head of the Montilivi campus library of the Universitat de Girona, for her assistance with the library processes and for helping us to clarify certain points.

Funding

Albert Trias acknowledges the PhD support grant Universitat de Girona BR/10. This work was partially funded by the MINECO project IPT-430000-2010-13 (SAKE) and the AGAUR grant for the CSI-ref.2009SGR-1202.

References

[1] Horowitz D and Kamvar SD, The anatomy of large-scale social search engine. In: Proc. 19th Int. Conf. World Wide Web. New York: ACM, 2010, pp. 431-440.

[2] Longo L, Dondio P and Barret S. Enhancing Social Search: a Computational Collective Intelligence Model of Behavioural Traits, Trust and Time. In: Nguyen NT and Kowalczyk R (eds.) Transactions on Computational Collective Intelligence. LNCS 6450. 2010, pp. 46-69

[3] Evans BM and Chi EH. 2008. Towards a model of understanding social search. In: Proc. 2008 ACM conference on Computer supported cooperative work (CSCW '08). New York: ACM, 2008, pp. 485-494. Doi:10.1145/1460563.1460641

[4] Evans BM and Chi EH. An elaborated model of social search. *Inf. Process. Manage.* 2010 46(6):656-678. doi:10.1016/j.ipm.2009.10.012 http://dx.doi.org/10.1016/j.ipm.2009.10.012

[5] Lueg C, Fisher D. From Usenet to CoWebs: Interacting with Social Information Spaces. Chapter 1, Studying Social Information Spaces. 2003

[6] Luck M, McBurney P, Shehory O. Agent Technology: Computing as Interation. A Roadmap for Agent-Based Computing. AgentLink, 2005. ISBN 085432 845 9

[7] Liljenback ME. ContextQA: Experiments in interactive restricted domain question answering, Master's thesis, San Diego University, 2007.

[8] Ong WJ. Orality and Literacy: The Technologizing of the Word. 1982. pp. 78

[9] Phillips H. The Great Library of Alexandria?. *Library Philosophy and Practice* 2010 (August) ISSN 1522-0222

[10] Borbinha JL, Jorge J, Ferreira J and Delgado J. A Digital Library for a Virtual Organization. In Proc. 31th Annual Hawaii International Conference on System Sciences) Vol. 2. IEEE Computer Society, Washington, DC, USA. 1998 doi:10.1109/HICSS.1998.651691

[11] Sanderson M and Croft WB. The History of Information Rerieval Research. Accessed online on: 8/10/2012 http://maroo.cs.umass.edu/pub/web/getpdf.php?id=1066

[12] Steichen B, Ashman H and Wade V. A comparative survey of Personalised Information Retrieval and Adaptive Hypermedia techniques. *Information Processing & Management*.(2012) 48:698-724

[13] Thomee B and Lew MS, Interactive search in image retrieval: a survey. *Int J Mulimed Info Retr*(2012) 1:1-86 doi:10.1007/s13735-012-0014-4

[14] Harper FM, Raban D, Rafaeli S and Konstan JA. Predictors of answer quality in online Q&A sites. In: Proc. 26th annual SIGCHI Conf. on Human Factors in Computing Systems, CHI '08, New York: 2008, pp. 865–874.

[15] Yang J, Morris MR, Teevan J, Adamic LA and Ackerman MS. Culture Matters: A survey Study of Social Q&A Behavior. In: Proc. 5th Int. Conf on Weblogs and Social Media 2011.

[16] Wang F and Sun Y. Self-Organizing Peer-to-Peer Social Networks. *Computational Intelligence*, 2008 24(3): 213–233. doi:10.1111/j.1467-8640.2008.00328.x

[17] Yu B and Singh MP, An agent-based approach to knowledge management, In: CIKM '02: Proc. 11th international conference on Information and knowledge management. New York: ACM, 2002, pp. 642–644

[18] Wu LS, Akavipat R, Maguitman A and Menczer F, Adaptive peer to peer social networks for distributed content based web search, in Social Information Retrieval Systems: Emergent Technologies and Applications for Searching the Web Effectively. IGI Global, 2007. [Online Available] http://sixearch.org/paper/bookchapter.pdf

[19] Trias i Mansilla A and de la Rosa i Esteva, JL Asknext: An agent protocol for social search, *Information Sciences*, 2012, 190:144-161. doi:10.1016/j.ins.2011.12.012.

[20] Aciar S, Royo M, de la Rosa JL and Serarols C. Increasing effectiveness in e-commerce: recommendations applying intelligent agents. *International Journal of Business and Systems Research* (IJBSR) 1/2007(1) pp. 26-40 ISSN: 1751-200X

[21] Gosain S. Mobilizing software expertise in personal knowledge exchanges, *The Journal of Strategic Information Systems*, 2007 16(3):254-277

[22] Ikkink KK and van Tilburg T. Broken ties: reciprocity and other factors affecting the termination of older adults' relationships. *Social Networks* 1999 21:131-146

[23] Morris MR and Horvitz E. S3: Storable, Shareable Search. In: Baranauskas C et al. (eds.). INTERACT 2007, LNCS 4662 Part I, 2007, pp.120-123

 © The Author(s), 2013

JIS Pre-Referee Version
Candidate Paper for the Journal of Information Science: http://jis.sagepub.co.uk

[24] Amershi S and Morris MR. A System for Co-located Collaborative Web Search. In: Proc. of ACM CHI 2008.
[25] Smyth B, Briggs P, Coyle M and O'Mahony M, Google Shared. A Case-Study in Social Search, Houben GJ et al. (eds.): UMAP 2009, LNCS 5535, Springer, 2009, pp. 283–294.
[26] Montaner M, López B, de la Rosa JL. A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review* 2003 19(4):285-330. doi:10.1023/A:1022850703159
[27] Agichtein E and Liu T. Modeling Information-Seeker Satisfaction in Community Question Answering. ACM Transactions on Knowledge Discovery from Data 3(2), Article 10. 2009.
[28] Chi EH, Information seeking can be social, *Computer* 2009 42 (3):42–46.
[29] Morris MR, Teevan J, Panovich K. What Do People Ask Their Social Networks, and Why? A Survey Study of Status Messages Q&A Behavior. CHI 2010. Atlanta: 2010.
[30] Liu Q and Agichtein E, Modeling Answering Behavior in Collaborative Question Answering Systems, In: Clough P et al. (eds.), Proc. 33rd European Conf. Advances in Information Retrieval, ECIR 2011, LNCS 6611, Springer, 2011, pp. 67-79.
[31] Bian J, Liu Y, Agichtein E and Zha H. Finding the right facts in the crowd: factoid question answering over social media. In: WWW, 2008, pp. 467-476.
[32] Jeon J, Croft WB and Lee JH. Finding similar questions in large question and answer archives. In: CIKM, 2005, pp. 84-90.
[33] Jurczyk P and Agichtein E. Discovering authorities in question answer communities by using link analysis. In: CIKM, 2007, pp. 919-922.
[34] Liu X, Croft WB and Koll M, Finding experts in community-based question-answering services.. In: CIKM, 2005, pp. 315-316.
[35] Zhou Y, Cong G, Cui B, Jensen CS and Yao J. Routing questions to the right users in online communities. In: ICDE, 2009, pp. 700-711.
[36] Guo L, Tan E, Chen S, Zhang X and Zhao YE. Analyzing patterns of user content generation in online social networks. In: KDD, 2009 pp. 369-378.
[37] Cao X, Cong G, Cui B, Jensen CS and Zhang C. The use of categorization information in language models for question retrieval. In: CIKM, 2009, pp 265-274.
[38] Cao Y, Duan H, Lin CY, Yu Y and Hon HW. Recommending questions using the mdl-based tree cut model. In: WWW, 2008, pp. 81-90.
[39] Suryanto MA, Lim EP, Sun A and Chiang RHL. Quality-aware collaborative question answering: methods and evaluation. In: WSDM 2007, pp. 221-230.
[40] Shtok A, Dror G, Maarek Y and Szpektor I. Learning from the Past: Answering new questions with past answers. World Wide Web Conference. WWW 2012- Session: Leveraging User-Generated Content. ACM 978-1-4503-1229.
[41] Guo J, Xu S, Bao S and Yu Y. Tapping on the potential of q&a community by recommending answer providers. In: CIKIM, 2008, pp. 921-930.
[42] Qu M, Qiu C, He X, Zhang C, Wu H, Bu J and Chen C. Probabilistic question recommendation for question answering communities. In WWW (2009).
[43] Liu M, Liu Y and Yang Q. Predicting best answerers for new questions in community question answering. In: L. Chen, C. Tang, J. Yang, Y. Gao. (eds.) WAIM 2010. LNCS, vol 6184. Springer: Heidelberg, 2010, pp.. 127-138.
[44] Nam KK, Ackerman MS and Adamic LA. Questions in, knowledge in?: a study of naver's question answering community. In: CHI 2009, pp. 779-788.
[45] Liu Q, Liu Y and Agichtein E. Exploring web browsing context for collaborative question answering. In: IIiX, 2010, pp. 305-310
[46] Kabutoya Y, Iwata T, Shiohara H and Fujimura K. Effective Question Recommendation Based on Multiple Features for Question Answering Communities. In: Proc 4th International AAAI Conference on Weblogs and Social Media. 2010. http://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/viewFile/1471/1871
[47] Walter FE, Battiston S and Schweitzer F. A model of a trust-based recommendation system on a social network, *Autonomous Agents and Multi-Agent Systems* 2008 16(1): 57–74.
[48] Trias A and de la Rosa JL. Propagation of Question Waves by Means of Trust in a Social Network, Christiansen H et al. (eds.): FQAS'11, LNAI 7022, Belgium: Springer, 2011, pp. 186–197.
[49] de la Rosa JL, Trias A, Ruusalepp R, et al. Agents for Social Search in Long-Term Digital Preservation. In: Sixth International Conference on Semantics Knowledge and Grid (SKG), 2010, pp. 363-366 ISBN:978-1-4244-8125-5 doi:10.1109/SKG.2010.60
[50] Montaner M, López B and de la Rosa JL. Opinion-Based Filtering Through Trust. CIA'02 Proceedings of the 6th International Workshop on Cooperative Information Agents VI, 2002, pp. 164-178.

 © The Author(s), 2013

[51] Su X and Khoshgoftaar TM. A survey of Collaborative Filtering Techniques. *Advances in Artificial Intelligence* 2009, article Id 421425 doi:10.1155/2009/421425

[52] de Souza CC, de Magalhaes JJ, de Costa EB and Fechine JM. Predicting Potential Responders in Twitter: A Query Routing Algorithm. In: Murgante B, Gervasi O, Misra S, Nedjah N, Rocha A, et al. (eds.) Computational Science and Its Applications. LNCS 7335, Springer, 2012 pp. 714-729. doi:10.1007/978-3-642-31137-6_54

[53] Wenyin L, Hao T, Chen W and Feng M. A Web-Based Platform for User-Interactive Question-Answering. *World Wide Web* 2010 12(2): 107-124. Doi: 10.1007/s11280-008-0051-3

[54] Krulwich B. LifeStyle Finder: Intelligent User Profiling Using Large-Scale Demographic Data. *AI Magazine* 1997 18(2): 37–45.

[55] Krulwich B and Burkey C. ContactFinder: Extracting Indications of Expertise and Answering Questions with Referrals. Working Notes of the 1995 Fall Symposium on Intelligent Knowledge Navigation and Retrieval. Technical Report FS-95-03, The AAAI Press. 1995, pp. 85–91.

[56] Simmons RF. Answering English questions by computer: A survey. *Communications Association for Computing Machinery* (ACM) 1965 8(1): 53-77.

[57] Sneiders E. Automated email answering by text pattern matching. In: Loftsson H, Rögnvaldsson E, and Helgadóttir S (eds.), Proc.Int Conf. on Advances in natural language processing (IceTAL'10), Berlin: Springer-Verlag, 2010, pp.381-392.

[58] Hsieh G, Kraut RE and Hudson SE. Why Pay?: Exploring How Financial Inventives are Used for Question & Answering. CHI 2010. https://www.msu.edu/~garyh/docs/hsieh-CHI2010-mahalo.pdf

[59] Hirschman L and Gaizauskas R. Natural language question answering: the view from here. *Natural Language Engineering* 2001 7(4):275-300. doi:10.1017/S1351324901002807

[60] Nishimura R, Watanabe Y and Okada Y. A Question Answer System Based on Confirmed Knowledge Developed by Using Mails Posted to a Mailing List. In: Proc. of the IJCNLP 2005, 2005.

[61] Resnick P, Iakovou N, Sushak M, Bergstrom P, Riedl J. GroupLens: An open architecture for collaborative filtering of netnews, In: Proc. Conf. Computer Supported Cooperative Work, CSCW'94, 1994, pp.175-186.

[62] Shardanand U and Maes P. Social information filtering: Algorithms for automating 'word of mouth', In: Proc. Conf. Human Factors in Computing Systems. CHI'95, 1995, pp. 210-127.

[63] Breese JS, Heckerman D and Kadie C. Empirical analysis of predictive algorithms for collaborative filtering, In: Proc. 14th Conf. Uncertainty in Artificial Intelligence, Madison:1998, pp. 43-52.

[64] Sarwar B, Karypis G, Konstan J and Riedl J. Item-based Collaborative Filtering Recommendation Algorithms, In: Proc. 10th Int. Conf. World Wide Web, WWW'01, 2001, pp. 286-295.

[65] Ahn HJ. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Information Sciences*, 2008 178(1):37-51.

[66] Sabater-Mir J and Sierra C. Review on Computationa Trust and Reputation Models. *Artificial Intelligence Review.* 2005. 24(1) 33-60. Doi: 10.1007/s10462-004-0041-5

[67] Rousseau D, Sitkin S, Burt R and Camerer R. Not so different after all: a cross discipline view of trust. *Academy of Management Review* 1988 23(3): 393-404.

[68] Pinyol I and Sabater-Mir J, Computational trust and reputation models for open multi-agent systems: a review. *Artificial Intelligence Review* 2012. doi:10.1007/s10462-011-9277-z

[69] Fang L, Huang M and Zhu X. Question routing in community based QA: incorporating answer quality and answer content. In proceedings of the ACM SIGKDD Workshop on Mining Data Semantics, ACM, 2012. Doi: 10.1145/2350190.2350195

[70] Blei DM, Andrew YN and Jordan MI. Latent dirichlet allocation. *The journal of machine learning research*, 2003 3:993-1022.

[71] Qu, M, Qiu G, He S, Zhang C, Wu H, Bu J and Chen C. Probabilistic question recommendation for question answering communities. In: WWW (2009)

[72] Hoffman T. Probabilistic latent semantic indexing. In: SIGIR 99.

[73] Yan Z and Zhou J. A new approach to answerer recommendation in community question answering services. In: Proc. 34th European conf. on advances in information retrieval ECIR'12. 2012. pp 121-132. doi:10.1007/978-3-642-28997-2_11

[74] Banerjee A, Basu S. A Social Query Model for Descentralized Search, In: Proc. 2nd ACM Workshop on Social Network Mining and Analysis, SNAKDD-08, Las Vegas: ACM, 2008.

[75] Cholvi V, Felber P and Biersack E. Efficient Search in Unstructured Peer-to-Peer Networks. *European Transactions on Telecommunications* 2004 15(6).

[76] Adamic L, Rajan L, Puniyani A and Huberman B. Search in Power-Law Networks. *Physical Review E*, 2001 64(4):46135-46143. Doi:10.1103/PhysRevE.64.046135

[77] Michlmayr E, Pany A and Kappel G, Using Taxonomies for Content-based Routing with Ants, Journal of Computer Networks 2007 51:4514-4528.
[78] Gyöngyi Z, Koutrika G, Pedersen J and Garcia-Molina H. Questioning yahoo! Answers. In: Proc. Of the 1st Worshop on Question Answering on the Web. 2008
[79] Kassin SM, Fein S, Markus HR. Social Psychology. Cengage Learning. Chapter 10. pp 432. ISBN: 9781133957744. 2010.
[80] Fiske AP. The four elementary forms of sociality : framework for a unified theory of social relations. *Psychological Review* 1992 99: 689-723.
[81] Lin TC, Wu S, Lu CT. Exploring the affect factors of knowledge sharing behavior: the relations model theory perspective. *Expert Systems with Applications* 2012 39:751-764.
[82] Xiang B, Jiang D, Pei J, S Xiahoui, Chen E and Li H. Context-Aware Ranking in Web Search. In: Proc. 33rd Int. ACM SIGIR conf. Research and development in information retrieval. ACM, 2010, pp. 451-458. doi:10.1145/1835449.1835525
[83] Freund L, Toms EG. Contextual Search: from Information Behaviour to Information Retrieval. In: Proc. of Canadian Association for Information Science. 2005. http://www.cais-acsi.ca/proceedings/2005/freund_2005.pdf
[84] Kraft R, Maghoul F, Chang CC. Y!Q: Contextual Search at the Point of Inspiration. In: Proc. 14th ACM Int. Conf. on information and knowledge management. 2005, pp. 816-823. doi:10.1145/1099554.1099746
[85] Hall H. Input-friendliness: motivating knowledge sharing across intranets. *Journal of Information Science* 2001: 27(3): 139-146
[86] Brett DH and Bontis N, MICRO-PAYMENT FOR YOUR THOUGHTS: Corporate knowledge exchanges and why incentives drive their use. Knexa white-paper. 2008. http://www.knexa.com/cgi-bin/Clients/signup.pl
[87] Malhotra Y. Enabling knowledge exchanges for e-business communities, *Information strategy: the executives journal*, 2002 18(3): 26-31.
[88] Carrillo C, de la Rosa JL and Canals A. Towards a Knowledge Economy, *International Journal of Community Currency Research*, 2007 11: 84-97
[89] Moreno A, de la Rosa JL, Szymanski BK and Bárcenas JM. Reward System for Completing FAQ, Frontiers in Artificial Intelligence and Applications – AI Research & Development ISSN 0922-6389, Vol: 202, pp: 361-370, Nov 2009, IOS Press
[90] Kamvar SD, Schlosser MT, Garcia-Molina H. Incentives for Combatting Freeriding on P2P Networks. In: Euro-Par, 2003 http://kamvar.org/assets/papers/freerider.pdf
[91] Buchegger S and Data A. A case for P2P infrastructure of social networks- opportunities & challenges. In: Proc. 6th Int. Conf. Wireless On-Demand Network Systems and Services. Utah: 2009, pp. 161–168.
[92] Hendler J, Avoiding another AI winter, *IEEE Intelligent Systems* 2008 23(2): 2–4.
[93] Dignum V, An overview of agents in knowledge management. In: Umeda M, Wolf A, Bartenstein O, Geske U, Seipel D, Takata O (eds.), Proc. 16th Int. Conf. on Applications of Declarative Programming and Knowledge Management, LNCS 4369. Fukuoka: 2005, pp. 175–189.

4. Asknext: an agent protocol for social search

Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva
Information Sciences 190 (2012) 144-161
doi:10.1016/j.ins.2011.12.012

Abstract

This paper describes the Asknext protocol, which automates knowledge exchanges by connecting agents using social networks. The protocol combines social answering systems, in which people answer queries or questions, with social feedback systems, in which feedback is used to sort search results. In Asknext, each agent represents a user, her contacts and her knowledge. When an agent receives a question, it will try to answer it; if it has no answer available, the agent forwards a message with that question to its contacts or shows it to its user. Asknext's main contributions are that it introduces stop messages and different propagation speeds for different messages, which allow a reduction in the spread of messages after an information need has been satisfied. A difference in the propagation speed of messages results in the search stopping when a relevant answer has been found. This paper describes a formalisation of the protocol, its mathematical model and experiments showing that the number of messages generated is lower than those generated by social search protocols such as Sixearch. Concretely, Asknext sends 20 times fewer messages than Sixearch; therefore, Asknext significantly improves the scalability of social search protocols, with as little as 0.05% loss of answer suitability, with the result that in many cases, there is no loss of suitability.

Published version cannot be used

Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva. “Asknext: an agent protocol for social search”. *Information Sciences*. Vol. 190, (May 2012) : 144-161

http://dx.doi.org/10.1016/j.ins.2011.12.012

http://www.sciencedirect.com/science/article/pii/S0020025511006517

Received 7 February 2011

Received in revised form 4 November 2011

Accepted 2 December 2011

Available online 14 December 2011

Abstract

This paper describes the Asknext protocol, which automates knowledge exchanges by connecting agents using social networks. The protocol combines social answering systems, in which people answer queries or questions, with social feedback systems, in which feedback is used to sort search results. In Asknext, each agent represents a user, her contacts and her knowledge. When an agent receives a question, it will try to answer it; if it has no answer available, the agent forwards a message with that question to its contacts or shows it to its user. Asknext’s main contributions are that it introduces stop messages and different propagation speeds for different messages, which allow a reduction in the spread of messages after an information need has been satisfied. A difference in the propagation speed of messages results in the search stopping when a relevant answer has been found. This paper describes a formalisation of the protocol, its mathematical model and experiments showing that the number of messages generated is lower than those generated by social search protocols such as Sixearch. Concretely, Asknext sends 20 times fewer messages than Sixearch; therefore, Asknext significantly improves the scalability of social search protocols, with as little as 0.05% loss of answer suitability, with the result that in many cases, there is no loss of suitability.

Keywords

Social search; Multi-agent systems; Scalability; Social networks; p2p

5. Studies on social network topologies and their effect in the dissemination of information in P2P social search

Albert Trias i Mansilla, Eusebi Calle Ortega,
Josep Lluís de la Rosa i Esteva, Marc Manzano Castro.
International Journal of Communication Systems (in review)

Abstract

In social search, both the information that the actor knows and the connections provided by these nodes are important. These connections allow actors to obtain the required information from unknown actors. Network topology can thus have an important effect on different query routing policies. Understanding which social network features affect query routing algorithms may help optimize some aspects, e.g., scalability, in social search systems. Unfortunately, popular data sets do not contain information about a social network; furthermore, it is not possible to have several real social networks with the same real data. The first main objective of this paper is to generate a set of artificial social networks for a recommender system data set, which requires studying which properties characterize social networks from other small network topologies. The second objective is to simulate several query routing algorithms, compare them and study which social network features affect them. To achieve these objectives, we generated 21 social networks and set up a simulator to evaluate the different query routing algorithms.

Embargo: until publication

Albert Trias i Mansilla, Eusebi Calle Ortega, Josep Lluís de la Rosa i Esteva, Marc Manzano Castro. "Studies on Social Networks Topologies and their effect in the Dissemination of Information in P2P Social Search". *International Journal of Communication Systems*. Submitted

http://onlinelibrary.wiley.com/journal/10.1002/(ISSN)1099-1131

Abstract

In social search, both the information that the actor knows and the connections provided by these nodes are important. These connections allow actors to obtain the required information from unknown actors. Network topology can thus have an important effect on different query routing policies. Understanding which social network features affect query routing algorithms may help optimize some aspects, e.g., scalability, in social search systems. Unfortunately, popular data sets do not contain information about a social network; furthermore, it is not possible to have several real social networks with the same real data. The first main objective of this paper is to generate a set of artificial social networks for a recommender system data set, which requires studying which properties characterize social networks from other small network topologies. The second objective is to simulate several query routing algorithms, compare them and study which social network features affect them. To achieve these objectives, we generated 21 social networks and set up a simulator to evaluate the different query routing algorithms

Keywords

Social Networks; Social Search; Multi-Agent System; Network Topology; P2P

6. Question Waves: an algorithm that combines answer relevance with speediness in social search

Albert Trias i Mansilla, Josep Lluís de la Rosa i Esteva
Information Sciences (in third review)

Abstract

This paper describes Question Waves, an algorithm that can be applied to social search protocols, such as Asknext or Sixearch. In this model, the queries are propagated through the social network, with propagation being faster through more trustable acquaintances. Question Waves uses local information to make decisions and obtain an answer ranking. With Question Waves, the answers that arrive first are likely to be the most relevant, and we computed the correlation of answer relevance with the order of arrival to demonstrate this result. We obtained correlations equivalent to heuristics that use global knowledge, such as profile similarity among users or the expertise value of an agent. Because Question Waves is compatible with the social search protocol Asknext, it is possible to stop a search when enough relevant answers are found; additionally, it is possible to stop the search with a small risk of not obtaining the best answers available. Furthermore, Question Waves does not require a re-ranking algorithm because the results come sorted.

Question Waves: an algorithm that combines answer relevance with speediness in social search

Albert Trias i Mansilla[1a], Josep Lluís de la Rosa i Esteva[a]

[a] *Agents Research Lab, TECNIO Centre EASY, University of Girona, Campus de Montilivi, E17071 Girona, Catalonia, Spain (EU)*

ABSTRACT

This paper describes Question Waves, an algorithm that can be applied to social search protocols, such as Asknext or Sixearch. In this model, the queries are propagated through the social network, with propagation being faster through more trustable acquaintances. Question Waves uses local information to make decisions and obtain an answer ranking. With Question Waves, the answers that arrive first are likely to be the most relevant, and we computed the correlation of answer relevance with the order of arrival to demonstrate this result. We obtained correlations equivalent to heuristics that use global knowledge, such as profile similarity among users or the expertise value of an agent. Because Question Waves is compatible with the social search protocol Asknext, it is possible to stop a search when enough relevant answers are found; additionally, it is possible to stop the search with a small risk of not obtaining the best answers available. Furthermore, Question Waves does not require a re-ranking algorithm because the results come sorted.

Keywords: multi-agent systems, p2p, query-routing, social networks, social search.

1. Introduction

Query-routing is a key element of social search; where the search process can be reduced in finding the right person to ask [42]. We propose a query-routing algorithm for social search that provides first the answers that are probably the most relevant. The heuristic behind the algorithm consists on asking first the best candidates to satisfy the information need. Although than the usefulness of search engines cannot be questioned, there are information that is not indexable (deep Web [5]) which contains data from online social networks with restricted access. Furthermore, one pending issue in the evolution of search engines is about answering open questions as can be recommendations, which are still ill-served, and social search [10] is a valid answer to them. In this paper, we refer to open questions as those abstract, subjective, and recommendation type questions [39]. For example, questions like: “What would be a good app to find bargains in Barcelona?” cannot be expressed in a way that today’s search engines can understand, let alone answer it. In social search, which emerged as a new paradigm of research interest, a good way to get a useful answer for that question would be by posing it to all your closer or distant friends or colleagues. Furthermore, Smyth et al. [33] note that 70% of the time users are searching for information previously found by their friends or colleagues. For these reasons, there are recently developed tools, such as HeyStacks [33] or the +1 button of Google, that contribute to the sharing of interesting results.

Due to these multiple concerns regarding how searches are conducted and the need to turn search into a social issue, several researchers suggest a change of paradigm. Advocates of the Social Search like Yu and Singh [42] expressed that often the most relevant information can be only accessed by asking the right people, and Nardi [28] wrote that “It is not what you know, but who you know”, which can be the motto of today’s search. Hendler [16] expressed this concept even more clearly by introducing the term of social machines: “Social-networking sites are replacing search engines and news sites as people’s favoured homepages and a new generation of applications is centring on large groups of people collaborating over an increasingly distributed network” and “one thing is for sure:

[1] Corresponding author, email address: albert.trias@udg.edu
Email addresses {albert.trias, joseplluis.delarosa}@udg.edu

many of the entrenched models of AI will have to be rethought as computers change from application devices to social machines".

Social search [10] is then a new paradigm, referred to as the village paradigm [17], a type of search that uses social interactions to provide results. It can be classified as Social Feedback Systems (SFS) and Social Answering Systems (SAS) [10]. SFS use social data to sort the answers, while SAS use people's expertise or opinions to question answering. SFS cannot address questions when the information is not available; SAS solve this problem, yet experts might receive a same question many times. What is interesting is that recommender systems [26] can be considered a SFS, ranking within a category (RWC) recommender systems [40]. In an RWC system, users obtain a list of items with a personalized ranking, as in SFS. That is why we consider recommender systems data are valid for testing algorithms of social search, what will be novel and extremely useful for the design and test of social search algorithms. Knowledge exchange portals (Q&A) are SAS that are usually used when users do not obtain satisfactory results from search engines [15], as is the case when they are searching for opinions or customized advice. The most important drawback of these portals is the frequent lack of answers, perhaps due to the fact that the right people are not willing to answer or are not accessible, or the question did not get through to them. Usually, experts receive a burden of questions that overloads their limited bandwidth, and in consequence, the questions might be answered slowly. There are three main options to minimize answering time and provide good results [1]: reusing content [6,19], routing questions to experts on the topic [13,23,45] and encouraging more answers by understanding answering behaviours [13,22]. Furthermore, questioner satisfaction is an interesting problem in SAS; Agichtein and Liu [1] showed that previous interactions are useful predictors of questioner satisfaction and are able to outperform human classifiers.

The drawbacks of social search is that querying all the people you know is very expensive in terms of effort and is thus not a viable solution. As we will see in the paper, there are several query-routing algorithms to help users satisfying their answer need through finding the right person to ask such a question. Instead, there should be an algorithm for getting the most relevant answers, relevant to the questioner, from the right person to answer such a question, with reduced questioning effort. That algorithm that put the question forward a sequence of subsets of acquaintances and later aggregates or selects the right answer is a new type of query-routing algorithm that is referred to as Question Waves (QW). QW uses intelligent agents that works on behalf of people with the aim of automating query-routing and combining the SFS with SAS approaches of social search that takes advantage of the benefits of both of them: when an agent receives a question; if it has enough knowledge then it will answer automatically, otherwise, if it considers that its owner will be interested to answer, then it poses to her the question; the agent also can forward the question to its acquaintances; the agent will send first the question to the acquaintances with a higher probability of providing a relevant answer. After certain time, if the information need is not satisfied, the agent will send the question to the following subset of acquaintances that are best suited to answer; this last process is repeated until their information need is satisfied or there are not acquaintances available.

To the best of our knowledge, there are no such algorithms with a proven track record of getting the most relevant answers. Experiments use measures of relevance (as recall) for comparing the several aspects of the algorithm that contributes to its performance.

QW algorithm is executed at every node of the social network, using local knowledge and contact lists to decide query-routing, social behaviour, and question-answering. Intelligent agents are chosen for QW as a suitable approach for social search because they share the reactive, social, proactive, and autonomous features of the actors, people, in the social networks; furthermore, they are a good match for P2P systems [24] that are at the cradle of the query-routing algorithms. Agents decide the actions to perform after receiving a question, are capable of asking questions to other agents, and can take the initiative of requesting knowledge maintenance from their users. Agents can improve the answering time by appropriately routing questions to experts and reusing previous answers.

We advocate the usage of P2P systems for this scenario due the following reasons: the model of a general search engine for everything might not be scalable with the size, variability and heterogeneity of the Web and its users [41]; the entities that create these centralized systems can have conflict of interest with the results [40]; or can act as a "big brother" [8]; social networks are inherently peer-to-peer (P2P) [8].

P2P systems require query-routing protocols to find the right information source. Query-routing for decentralized search is an important research problem with applications in social search and file-sharing environments [3].

In all, QW can be described as a query-routing algorithm for social search using agents that operate on the same (P2P) social network of people. We will see that QW provides first the answers of the highest probability of being relevant before providing other, less relevant answers. This study demonstrates the mechanisms behind such results with every detail. Section 2 contains a brief literature review of query routing in unstructured P2P networks. In Section 3, the proposal of QW is presented in detail. In Section 4, simulations of the algorithm are described, and their results are discussed. Finally, in Section 5, the conclusions and future work are presented.

2. Literature review

Considering social search, query-routing has been applied to several domains, such as Internet browsers [41], question answering [3,17,38] and recommender systems [40]. The classification of the query-routing algorithms used in those domains is based on if the algorithms are unicast or multicast. In unicast query-routing algorithms, a query is only sent to one acquaintance or user [3,25,34], while multicast algorithms are ones in which a query is sent to many of them [38,40,41].

To better understand query-routing, it is important to know Milgram's experiments. In the 1960s, Milgram carried out a set of experiments to study the small-world problem. [34] explains the experiments: in brief, they selected 2 sets of individuals. The individuals in the first set received a document explaining the experiment, defining who should be the target person (belonging to the second set) of the document and containing some information about her. If the people of the first set knew the target person (TP) personally, then she would have sent the document to the TP; otherwise, she would have sent it to an acquaintance more likely to know the TP, and these acquaintances would be requested to repeat the process. The authors sent the document to 296 initial users, of which 73% followed the instructions, and the other 27% ignored it. Each time the folder was forwarded, there was probability of it being forgotten (dropped). From these experiments, we can see that unicast query-routing is feasible, but its drop rate considerably reduces the success rate; in the work of Travers and Milgram, only 64 folders reached their destination, while the others were dropped at some point.

More recently, but in a similar vein, Banerjee and Basu proposed the Social Query Model (SQM) [3], which is a probabilistic model that measures the probability of obtaining an answer P_i (Eq. 1). In the SQM, the nodes that receive a query can ignore, answer or forward it. The model considers the following components:

- The expertise $e_i \in [0,1]$ indicates the probability that the node a_i answers a query, with a probability 1- e_i that the node a_i forwards the query.
- The correctness $w_i \in [0,1]$ indicates the probability that the answer provided by the node a_i is correct.
- The response rate $r_{i,j} \in [0,1]$ indicates the probability that a node a_j accepts a query from node a_i. The probability that a node a_j drops the query is then 1- $r_{i,j}$.
- The policy π_j of a node a_i is a probability distribution that indicates the probability of selecting neighbours of a_i (N_i) when a_i decides to forward a query; concretely, each π_j^i indicates the probability that a_i forwards the query to a_j.

$$P_i = w_i e_i + (1 - e_i) \sum_{j \in N_i}^{n} \pi_j^i \, r_{i,j} P_j \tag{1}$$

The SQM is based on unicast query-routing in which the authors propose an algorithm to compute the optimal policy, although they also suggest finding the next-best policies in the case of a multicast application. In their experiments [3], they set the expertise by random or based on degree, making nodes with higher in-degree to have more expertise. The response rate can also be random or degree-based; the degree-based rate is directly proportional to the expertise of the asking node and inversely to the node requested because the authors assumed that "people tend not to ignore requests from experts" and "experts tend to be often too busy to answer".

[25] presents SEMANT, an algorithm for distributed searching in P2P networks. In this algorithm, a forward ant that has the responsibility of answering the query is generated for each one; when it reaches a node with some content that might satisfy the query as an answer, the forward ant finishes its travel, and a new backward ant is generated and follows the inverse path while dropping pheromones. The algorithm uses two strategies for query-routing: in the exploiting strategy, the ant follows a link to the best, yet unvisited, neighbour; in the exploring strategy, the forward ant tries to find new paths. For every unvisited peer, it is randomly decided whether the peer should be visited on the basis of its goodness value. If more than one peer is selected, the forward ant is duplicated; each peer stores the information of the ants that visited it, allowing termination of the copies of the forward ant if they reach the peer more than once. In SEMANT, the exploiting strategy is clearly unicast, while the exploring strategy is multicast.

Multicast query-routing in unstructured P2P networks is usually based on the breadth first search (BFS) algorithm, with the following main idea:

- one node, that starts a search, asks a subset of its neighbours by sending a query/question message;
- when a node receives a question, the node may answer or forward it;
- when a node has an answer, an answer message is sent following the inverse path.

These algorithms usually use the *time to live* (TTL) to reduce the number of question messages. The TTL determines how many hops can go between the questioner and the answerers. The TTL is initialized, and every time it is forwarded, it is reduced by one; when the TTL reaches 0, the message cannot be forwarded anymore.

Sixearch [41] is a P2P search engine based on BFS that uses TTL. Peers store their neighbours' profiles with the goal of increasing the probability of choosing an appropriate neighbour to receive a question. Peers store information in these profiles about the terms in which their acquaintances have expertise. Initially, a profile is requested from an acquaintance, and the profile is subsequently updated with the interactions of the node with its acquaintances.

Walter et al. [40] proposed a recommender system for P2P social networks in which query-routing is also based on BFS. In their model, each answer has a trust value, which is the trust associated to the path followed to find the answer. All possible answers are collected, and finally, an answer is selected randomly out of the set, with answers that come from more trusted paths having a higher probability of being chosen. The authors concluded that this model shows self-organisation and sub-optimal performance.

More recently, the Asknext protocol [35,38] was introduced, resulting in 20-fold fewer messages and higher scalability than Sixearch or related social search BFS query-routing based protocols. Asknext is also based on BFS with the novelty of stop messages, which stop questions from being sent when a relevant answer is found. The use of *stop messages* requires that *answer* and *stop* messages travel faster than *question* messages. It is possible to use a dynamic speed for *question* messages, making their propagation slower, and doing so allows the question propagation to be stopped at the same hops distance that a relevant answer is found (Eq. 2). T_F and T_R are the periods during which questions are forwarded and answers are sent, respectively. The forward speed can be set using Eq. 2. For example, with a $T_R = 1$ for nodes at distance (r) 1, T_F should be ≥ 3; for nodes at $r = 2$, T_F should be ≥ 5, and for nodes at $r = 3$, T_F should be ≥ 7. In practice, using different periods for T_F and T_R does not mean that the question messages travel slowly but rather that a node retains a question message for longer before forwarding it.

$$r < \left\lfloor \frac{T_F}{2T_R} \right\rfloor = \left\lfloor \frac{v_R}{2v_F} \right\rfloor \tag{2}$$

In the case of multi-agent referral system (MARS) [42,43,44], the authors follow a different approach. Yu and Singh proposed a P2P application in which agents assist users in obtaining and following referrals to find experts who might answer their questions. When a user formulates a question, it is processed by her agent, who determines which contacts should be the question recipients. When an agent receives a question, it decides whether to show the question to its owner or answer by providing referrals. When an agent receives an answer, the user's feedback is used to evaluate the expertise of the answerer.

While in the unicast query-routing there is no threat to the system scalability and costs, there is a lower probability of finding an answer. In the end, using multicast query-routing threats the system scalability, yet there is higher probability of satisfying the information need.

In the following Section, we will explain our contribution: each node consists on an intelligent agent that represents a user; these agents conduct several attempts to obtain answers in multicast query-routing and relax the problem of threatening system scalability to increase the probability of finding an answer. Initially, a question is sent to the best candidates. After a given time, if the information need is not satisfied (because it is not answered or the answer is not satisfactory) then the question is sent to the next best candidates. The process is repeated at each agent until the question is answered or there are no more candidates.

3. Question Waves

As has been stated in Section 2, unicast query-routing can be used to satisfy information needs while bothering only a few people in the network, but the probability of satisfying the information need is low; in contrast, using multicast query-routing, so that a question is propagated to all of the acquaintances, threats the system scalability. Finding the best number of acquaintances to which to forward a query is not straightforward. We consider that the best behaviour is requesting as few users as possible and obtaining the greatest or enough number of correct, relevant answers available. However, going rather far in reducing the number of recipients might avoid obtaining some relevant answers; otherwise, requesting all available peers can overload the system.

We consider that one option to deal with this problem is to first ask the candidates who are estimated to give the best answers and only then, if after some time the initial candidates are not able to provide a suitable answer, requesting the next in the contact list. Heuristically, a unicast query-routing can be used when the requester is near to an expert willing to answer and highly available. In the worst scenario, the system will work as a BFS multicast query-routing in which the majority of the nodes are requested only when no answer is yet available. The number of candidates that are requested will be adapted on the basis of the availability of answers. Subsection 4.3 provides details about the number of messages generated and the number of relevant answers provided by the different algorithms considered.

The work described in this paper is based on an agent P2P social network; in which each user is represented by an intelligent agent (we will refer them as agents), we also will refer users as owners of an agent. Each user has a knowledge base that is managed and can be accessed by her agent. Those agents can send questions to a subset (Sa) of their acquaintances, and when they receive a question they can forward the question to Sa as well or can answer. The answers follow the reverse path [25,38,40,41].

For each question, the agents can play three roles:

- **Questioner**: The questioner is the agent who started the question.
- **Answerer**: An answerer is an agent who answers a question with the agent's own knowledge or its user's knowledge.
- **Mediator**: A mediator is an agent who receives a question and forwards it to others. They also may forward an answer that they receive from another mediator or an answerer.

In the following Subsection, we will explain briefly the probability of obtaining an answer in a multicast query-routing, starting from the unicast social query model [3]. Then, with a better understanding of multicast query-routing, we will explain our main contribution, Question Waves, including the main points and the analogy from physical waves. In Subsection 3.3, we will explain some aspects that can be taken into account to set up Question Waves.

3.1 Introduction to multicast query-routing

In Section 2, we explained briefly the SQM proposed by Banerjee and Basu [3]. Their model is based on unicast query-routing and in that case, the probability of obtaining an answer asking others (expressed by the sum operator) contains a mutually exclusive probability. In the case of multicast query-routing, the probability of obtaining an answer that satisfies the question is not mutually exclusive, as several candidates can answer it correctly. We consider that it is the only aspect needed to be considered with the aim to apply their model in multicast query-routing.

In this Subsection we explain the probability of satisfying a query in multicast query-routing, in function of the distance. Given that:

- At a distance d, $\mathrm{P}_{i,j}(d)$ is the probability that the question q_j of questioner agent a_i is satisfied with the answers obtained from answerers at a maximum distance d.
- At a distance d, $\partial\mathrm{P}_{i,j}(d)$ is the probability that the question q_j of questioner agent a_i is satisfied with the answers obtained from answerers at a distance d exactly.
- $k_{i,j}$ is the probability that an agent a_i satisfies the question need with an own answer the question q_j.
- At a distance d = 0, the probability of satisfying the question need depends only on the agent a_i.

Then:

- $P_{i,j}(0) = k_{i,j}$
- $P_{i,j}(d+1) = P_{i,j}(d)\ or\ \partial P_{i,j}(d+1) = P_{i,j}(d) + \left(1 - P_{i,j}(d)\right) * \ \partial P_{i,j}(d+1)$

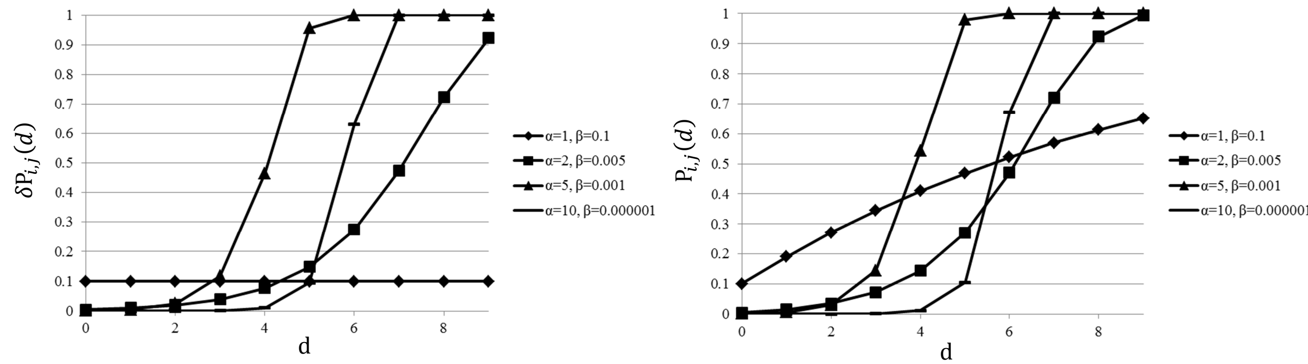
As a simplification, we assume the following major constraints:

- $\forall_i\ k_{i,j}{=}\beta.\ \beta\epsilon[0,1]$.
- Agents never drop a question.
- The social network topology is an infinite tree of degree α (α∊ℕ), so when the nodes forward a question, it is forwarded to α nodes that did not previously receive it.

Now:

- $\delta\mathrm{P}_{i,j}(\mathrm{d}) = 1 - (1 - \beta)^{\alpha\mathrm{d}}$
- Given a $\beta < 1$ and a $\alpha > 1$, then $\lim_{d\to\infty} 1 - (1 - \beta)^{\alpha\mathrm{d}} = 1$

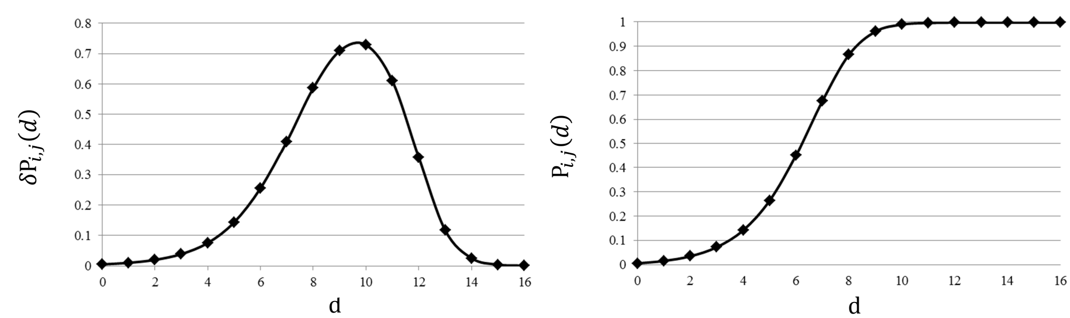
Example for several α and β are shown on Fig. 1.



(a) (b)

Fig. 1. (a) $\delta P_{i,j}$ in function of distance (b) and $P_{i,j}$ in function of distance. For several β and α.

Because social networks do not follow a tree topology, it is not always possible that each node forwards the question to α nodes that did not previously receive the question. Furthermore, social network topologies are finite. If the network topology is random, the probability that a neighbour already received the question will depend on the number of nodes that have been forwarded to before. In this case, the probability depends on the current distance. However, all nodes do not have the same amount of contacts. For example, for a topology of 1,000 agents with an average nodal degree of 3, we can obtain a situation similar to that in Fig. 2.



(a) (b)

Fig. 2. (a) $\delta P_{i,j}$ in function of distance, and (b) and $P_{i,j}$ in function of distance. For $\beta = .005$ in a finite network of 1000 agents with an average nodal degree of 3.

We can see that $\delta P_{i,j}(d)$ starts growing until a certain distance, after which it starts to decrease. The maximum point will depend on the shortest path from the questioner to the other nodes the point at which $\delta P_{i,j}(d)$ starts to decrease is the average shortest path length. When using these simplifications, $P_{i,j}$ does not depend on the network topology as long as all nodes are requested. With different k_i but accessing the nodes in the same way, we obtain the same kind of results as in Fig. 2.

If we consider a case in which all of the agents do not put equal effort toward trying to obtain an answer—for example, not all of the agents forward the question to the same number of contacts and the probability that agents can satisfy the information need (p) depends on the contacts' expertise (k), correctness and response rate—and if we assume that all of the agents are able to identify the nodes that are more likely to answer and first ask the nodes with greater p then ask other nodes with the next highest p in different attempts (or *waves*), we expect a higher probability of satisfying the information need with answers from earlier waves.

Now instead of evaluating $\delta \mathrm{P}_{i,j}(d)$, we are evaluating $\delta \mathrm{P}_{i,j}(t)$ where t represents the time in simulation steps. We simulated an environment in which the questioner agent sends a request to 5 nodes, sorted from greater expertise to less, at timestamps $\{0, 1, 2, 3, 4\}$. There is a higher $\delta \mathrm{P}_{i,j}(t)$ from the initial waves (Fig. 3 (a) and (b)).

Furthermore, if the questioner agent delays the waves, for example, initiating them at timestamps $\{0, 10, 20, 30, 40\}$, we obtain the probabilities in Fig. 3 (c) and (d). In the first attempt (first wave), $\delta \mathrm{P}_{i,j}(t)$ is higher than in the other waves; after an initial increase, $\delta \mathrm{P}_{i,j}(t)$ decays over time. In this Section, we will try to obtain a system that generates the kind of probability distribution from Fig. 3 (c) when all of the agents have the same behaviour.

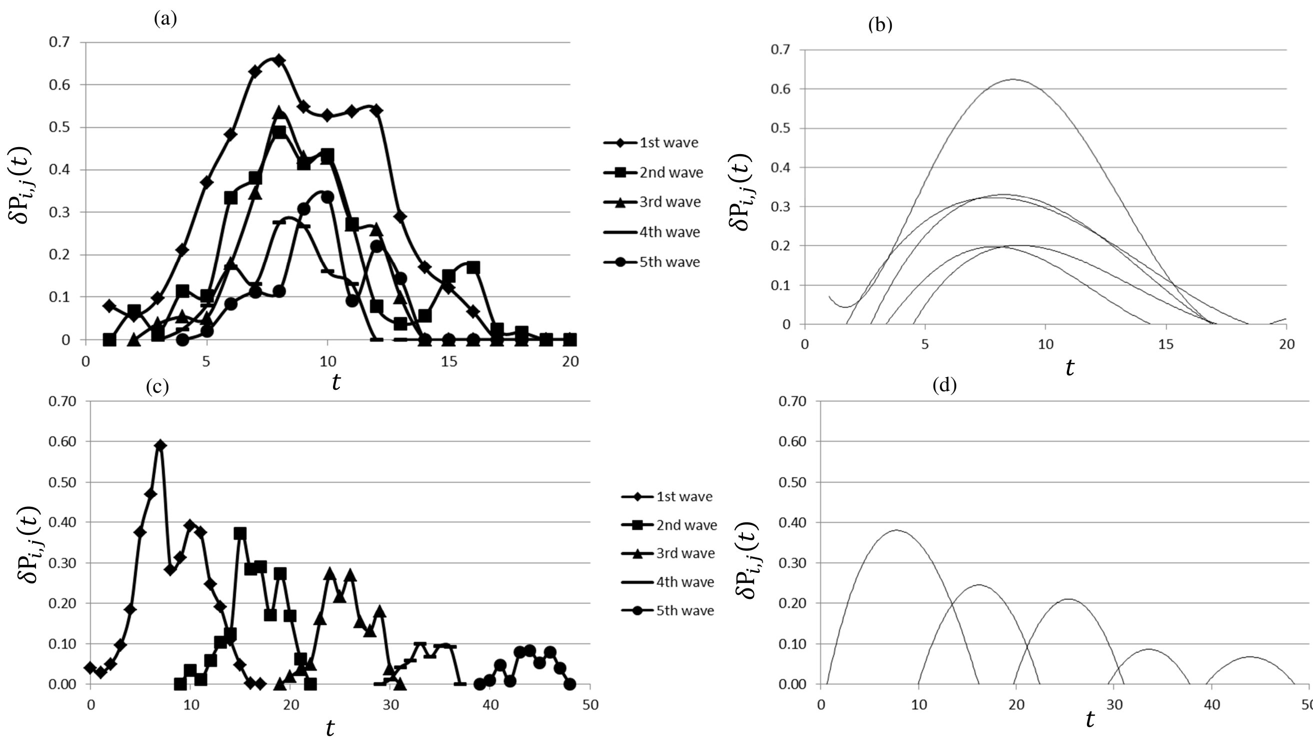


Fig 3. $\delta \mathrm{P}_{i,j}(t)$ for each attempt or wave. (a) and (c) show the values obtained through simulation, while (b) and (d) show its tendency lines. (a) and (b) are the results when all of the agents execute the same behaviour, and (c) and (d) are the results when a questioner agent has applied higher delay to the waves. y axis represents the probability in that time step, while x axis represents the time in simulation steps.

In the area of delay tolerant networks, Bulut et al. proposed and algorithm, that also uses some attempts to distribute copies of a message [9]. The problem they deal with consists on delivering a message before a deadline using the minimum number of message copies. Their algorithm first send a small number of copies (this is an attempt), and if after some time periods it is not delivered to the destiny, then more messages are sent to increase the delivery probability (further attempts). The main idea is that if the message is delivered in the initial attempts then the number of message copies needed to deliver the message has been reduced. They also use messages to stop the propagation of the copies once the message has been delivered (they call it acknowledgment of delivery). Although its domain of application is different, the usage of several attempts follows a similar idea of QW. One of the most important differences dealing with different attempts in social search and delay tolerant networks, is that in the first one it is important to provide relevant answers, while in the second one it is only important to deliver the message. That is why in this paper we compute several measures in our simulation to evaluate that the most relevant answers are received early.

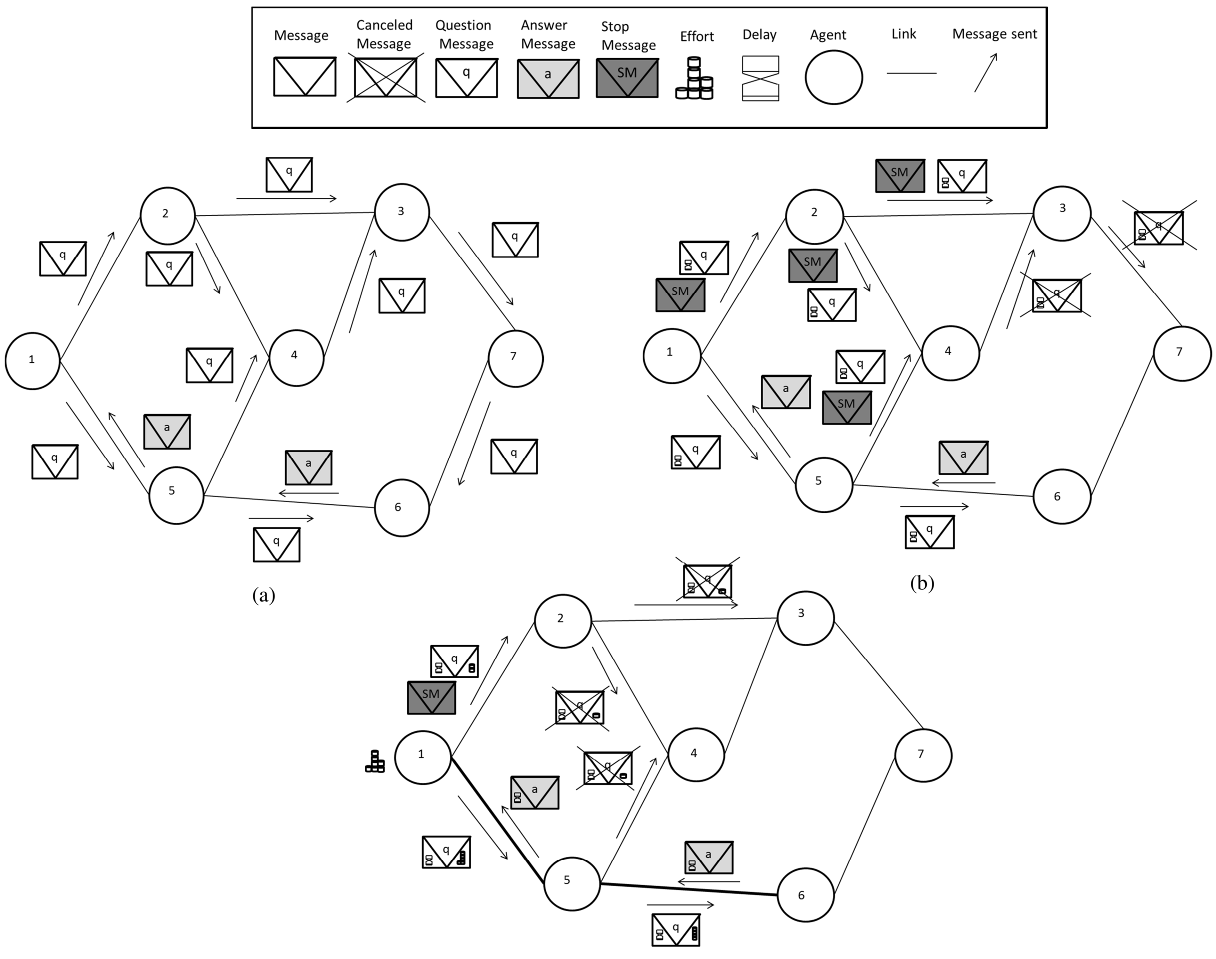


Fig. 4. Example of BFS (a), Asknext (b), and QW (c) with the same social network, where a_1 is the questioner and a_6 the answerer.

Fig. 4. shows some of the main differences among BFS, Asknext and QW search algorithms. In this example, agent a_1 is the questioner and a_6 is the only possible answerer, because one answer would be enough for the questioner and mediators. We will explain how it may work with the different algorithms:

- BFS is the most basic algorithm, where only there are question messages (q) and answer messages (a). These example corresponds to a TTL of 4 or greater:
  - Step 0: a_1 is the questioner and sends the question to $\{a_2,a_5\}$.
  - Step 1: a_2 and a_5 forward the question to $\{a_3,a_4\}$ and $\{a_4,a_6\}$ respectively.
  - Step 2: a_6 answers to a_5, at the same time a_4 forward the question to a_3, and a_3, who received the message previously from a_2, forwards it to a_7.
  - Step 3: a_5 forwards the answer to a_1, and a_7 forwards the question to a_6.

- Asknext introduces the usage of stop messages (SM), and it needs a delay on the question messages that depends on the distance, according Eq. 2.
  - Step 0: a_1 is the questioner and sends the question to $\{a_2,a_5\}$.
  - Step 1: a_2 and a_5 receive the question, they do not have an answer, so they will forward the question in step 3.
  - Step 2: a_2 and a_5 wait until step 3.
  - Step 3: a_2 and a_5 forward the question to $\{a_3,a_4\}$ and $\{a_4,a_6\}$ respectively.
  - Step 4: a_6 answers to a_5. a_3 and a_4 will forward the question at step 8.
  - Step 5: a_5 answers to a_1. a_5 and sends a SM to a_4.
  - Step 6: a_1 sends a SM to a_2. a_4 cancels forwarding the question.
  - Step 7: a_2 sends a SM to a_3.
  - Step 8: a_3 cancels forwarding the question.
- QW adds the concepts of effort, the question delay is based on trust on the contacts, not in the distance from the questioner as Asknext, and the answers also can contain a delay. In Fig. 4. (c) thicker links represents trust, and the question message propagate earlier for those links.
  - Step 0: a_1 has 7 units of effort to try to obtain and answer. a_1 decide to send first the question to a_5 with an effort of 5 units, and it will send to a_2 the question with an effort of 2 units at step 3 if no answer is found yet.
  - Step 1: a_5 sends the question immediately to a_6 with an effort of 4 units, and a_5 will forward the question to a_4 at step 4 if no answer is found yet.
  - Step 2: a_6 has confidence in his answer and sends it immediately to a_5.
  - Step 3: a_5 forward the answer to a1, and cancels forwarding the question to a_4. a_1 has no received an answer yet and forwards the question to a_2 (it is the second wave).
  - Step 4: a_1 receives the answer, and sends a SM to a_2. a_2 decides to forward the question to a_3 and a_4 at step 6 as it has not enough trust on that agents.
  - Step 5: a_2 receives the SM and cancels forwarding the question to a_2 and a_4.

3.2 Modelling multicast query routing as waves of questions or question waves

As we have seen in Subsection 3.1, it is possible to first ask the acquaintances with the highest probability of providing a relevant answer. We were distantly inspired by physical waves for modelling this process, which is why we call it *Question Waves* (QW), however the aim of QW is not obtaining physical waves properties and the resulting algorithm does not guaranties any of them, physical waves have been only an inspiration source.

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 probability of finding appropriate answers decays after every attempt. The question propagates through the network of agents, which amplify or attenuate the effort invested in answering it, as a wave is attenuated over distance in an aquatic environment. QW tries to solve the following problems:

a) In P2P there are scalability issues when you ask all peers.
b) If you ask only one peer (or a small subset), it is very likely you will not find the right answer.
c) If you ask several sequentially it is very likely you get stuck if any in the sequence not answers you.

In any case (b, c) choosing the right peers to ask is not straightforward. The solution that we propose to address the above problem is to use several attempts (waves), elapsed in time. In each attempt, the question sender (questioner or mediator) selects the most reliable acquaintances who have not been previously selected for that question. Adding new recipients implies that the agent is not sure that it will receive an answer from the current recipients and is trying to obtain an answer from less-trusted recipients. 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 least relevant. The advantage of QW is that multiple agents are committed to finding the answers with diversified options to find them, resulting in more relevant[2], faster[3], and more robust[4] answers.

In physics, "a wave may be defined as a periodic disturbance in a medium that carries energy from one point to another" [14]. In our model, when an agent has a question, the agent is a wave source and generates a disturbance on the system with an amount of energy (E_0). This energy indicates the amount of effort that the agent devotes to satisfy its question need. The probability of finding a relevant answer depends on the amount of effort invested in finding, and the agents can transmit part of this effort to others, delegating a part of the task (finding relevant answers). The propagation media are the links of the social network.

For physical waves, the Huygens principle states that each region reached by a wave generated by a disturbance generates another disturbance in other regions. In our model, when an agent receives a question, it may act as the next mediator that transmits the disturbance to other agents forwarding the question.

A physical wave speed depends on the medium properties. In our model, the medium is the link, and the speed of the question message will depend on the trust as the key property of the link. When a physical wave is propagated to several directions, the energy is spread over a large area, reducing the intensity at each point, although the total amount of energy in the system remains equal. In our model, the effort is also distributed among a subset of acquaintances; on average, each agent will receive less effort as more acquaintances are requested. In our model of question waves, effort does not need to be propagated uniformly to all of the acquaintances; agents can decide how they distribute the effort among their acquaintances.

In physics, waves may lose energy due to attenuation [14]. The loss of energy is due to the viscosity and friction of non-perfectly elastic media and is described by the absorption law. As in question waves we modelled energy as effort it can be lost due to communication costs. Some effort will be dissipated due to friction in the sense that not all of the effort transmitted to an agent is converted to effort to answer a question because these agents may be less motivated to help a concrete acquaintance or the acquaintance does not want to contribute to this question. The principle of the superposition of physical waves explains that when several waves are coincident in the same space, their amplitudes are added. This interference can be constructive, i.e., resulting in a bigger amplitude, or destructive, resulting in a lower amplitude. In our model, it is possible that one agent receives a same question more than once from different sources or as a result of several question waves. In this case, it is possible to use more effort than that contained in each request; for example, the agent may add the efforts requested from each request. Finally, an answer can act as a destructive interference, e.g., when a relevant answer is found, further attempts are not needed. This destructive interference can be propagated in the form of answers and stop messages.

To set up an environment to experiment with Question Waves (QW), it is necessary to consider the effort introduced with disturbances, attenuation, friction, speed of links to acquaintances, effort distribution, and constructive and destructive interferences. In the following Subsection, we will explain some ideas to build a QW environment.

3.3 Setting up a question wave behaviour

Previously, we have seen that to build a QW environment, we must consider the effort introduced with disturbances, attenuation or effort costs, medium properties (such as speed and the amount of effort transmitted to the recipients), and interferences (constructive and destructive). But in QW, the properties of the media are not physically different; the agents decide which acquaintances to send the question to first and which amount of effort the recipients would receive. A trust model is needed for this aim to evaluate the behaviour of the acquaintances, requiring some measures to evaluate the answers that they provide. In this Subsection, we will start with the answer evaluation, we will explain what the trust model must consider, and then, we will explain how to set the effort costs, medium speeds, effort distribution, and interferences.

[2] Relevant in the sense that the answers come ranked by trust.
[3] Faster in the sense of reducing the burden of questions, and the agents are less overwhelmed.
[4] Robust in the sense of finding answers persistently.

3.3.1 Answer relevance and trust model

The relevance of one answer might depend on several factors, such as the expertise (*k*) of the person who generated it, the grade of implication (I) put on answering (the amount of effort), and the rewards used to motivate the agent, although motivation might also stem from the task itself or who asked for the task. Answer relevance also depends on other factors (R), such as the personal situation, the amount of time available to answer, the answerer mood, etc. [36,37]. The first aspect that we must consider before setting up a QW environment is if the answers will be generated to satisfy the requested information (i.e. requesting an answer from the user), if the system will reuse the answers available in the community, or both. It is also important to consider if the answer relevance is determined subjectively or objectively. In Section 4, we simulated a case in which the answers were objective and generated another one in which the answers were reused and subjective.

The trust model is a key aspect of the QW behaviour because trust is used to decide the order in which to ask agents. Trust can be used as a predictor of answer relevance as well as it can be used to distribute the energy among acquaintances. In environments like this network, trust models tell us what is expected from an acquaintance.

Agents' behaviour may be based on *reciprocity*. [18] explains that social exchange theory assumes that people try to have balanced relationships (reciprocity); people prefer relationships in which they give and receive a similar amount of support. When there is a discrepancy between giving and receiving, the continuation of the relationship is threatened. There are some exceptions, such the cases of family or close friends. In close relationships, people feel responsible for each other's well-being and do not consider the balance. For these reasons, we think that a Q&A agent will have contacts classified into two groups: one group for whom the agent feels responsible and does not consider the balance in the relationship and another group for whom the agent expects the relationship to be balanced. We call the members of the first group **close acquaintances** and those of the second group **convenient acquaintances** [36].

In QW, when we consider convenient acquaintances, a trust model can be used to estimate the viscosity as a kind of *reciprocity*. If an agent does not receive any help from an acquaintance, the acquaintance's trust value should decrease. Trust models also should detect the presence of malicious agents, who, for example, answer fast with spam.

3.3.2 Question delay

The QW behaviour is based on initially requesting sources with a higher likelihood of returning a suitable answer. For this proposal, it is necessary to implement a function that indicates the time needed to wait before submitting the question. This function can be implemented as a continuous or a piecewise function. When an agent sends a question to another recipient, the recipient's ability as an answerer but also as a mediator should be considered.

3.3.3 Delaying an answer

When an agent receives a question and has an answer (or a set of them) available, it is able to answer immediately. In contrast, if the agent thinks that it is possible to obtain a better answer from an acquaintance, then it can delay the answer to allow a better one to be sent. In the case that the agent does not know of an available answer but can generate one, then it can delay the answer generation if the agent is aware of the quality of the answer it can generate. The heuristics used are sending any available answer when the agent considers that the probability of obtaining a better answer is low and delaying answering otherwise.

3.3.4 Effort

When an agent has to initiate a question, the agent puts an amount of effort E_0 to answer it. This effort is put directly into answering or obtaining an answer from acquaintances or into the effort that the agent will delegate to acquaintances.

Some of the effort can be lost in the search of answers because there are some communication costs that include processing the messages, sending them, and planning how to satisfy the requests. Most of these costs do not depend on the amount of effort requested, and we will refer them to as *constant costs*[5]. In contrast, the delegated agents may give less effort than requested due to the implication relation they have to the task, and the imbalance can be seen as some kind of commission or the benefit that the delegated agents get in the case that the system uses social currencies; we will refer to these as *friction costs*.

Effort is also used to obtain answers. In the functioning of the environment, answer generation can require different amounts of effort (one clear example is numerical analysis, in which more effort can be understood as more iterations) or can have a constant cost, such as querying a database.

The effort that an agent uses in solving a question can be used to set a threshold τ after which the agent will stop trying to obtain new answers. This threshold can indicate the number of answers sent, the probability of providing a relevant answer, etc.

Although the model considers that the effort will be attenuated with the question distribution, agents are autonomous, and it is possible that one agent might use more effort to obtain an answer than requested. This excess can happen in the case that one agent has its own interest in obtaining an answer for the question or it would like to improve the relationship with the requester (which can be seen as a kind of investment).

3.3.5 Effort distribution

When an agent delegates a question to its acquaintances, the agent can decide how the effort can be distributed. The most basic configuration would be an isotropic distribution, in which all of the acquaintances would receive the same amount of effort. However, it is also possible to use other configurations, which distribute effort as a function of trust, for example. The distribution may depend on the question, on aspects as the domain or question context, or on aspects of the self-confidence of providing a relevant answer. An agent can also take into account not continuously annoying the same agent, with the intention of not threatening their relationship.

3.3.6 Interferences

Mainly, the model considers two kinds of interference, constructive and destructive. Constructive interference is produced when an agent receives the same petition from different sources. As a function of how the model is designed, the amount of effort taken into account to try to answer a question can be the minimum effort transmitted in one of the requests or the sum of the requests.

Destructive interferences are transmitted by answers and stop messages. For answers, when an agent receives enough answers of a certain quality that cross the threshold σ, the agent will stop trying to find more answers, and it can also ask to stop the search process by sending a stop message to the agents it posed the question. The objective threshold σ will depend on the effort put toward answering the question.

4. Simulations

The objective of this Section is to show how the QW model can be implemented and the benefits that it can provide. Optimizing the results will be a next step out of the scope of this paper because it would depend on every context. In our simulations, a set of agents $A=\{a_0, a_1, \ldots, a_i\}$ perform the algorithm in Fig. 5 at every simulation step.

[5] By *constant*, we refer to the fact that the costs do not depend on the amount of effort, but the costs can be different for each agent or can change over time.

This Section contains two Subsections: in the first Subsection, we will use randomly generated data models to show that the answer relevance is correlated with the answer speed and that the distribution of ∂P in practice is similar to that expected from Fig. 3 (c). The second Subsection shows how QW is used as a collaborative filtering system.

```
Method Step
For each Received Answer
  If Own Question, Update result and Trust
  Else forward answer
      If first answer from sender, Update Trust
      If question objective achieved
             Deprogram messages
             If use stop messages, program stop messages
If I have a stop message
   Deprogram messages
   Forward stop message to my recipients
If I have a new Own question
     Select contacts in contact waves; Program question messages
For each received question
   If I received it before, ignore it
   Else
     If I can answer, Program answer message
     Select contacts in contact waves; Program question messages
If I have a programmed answer
  Send answer
  If question objective achieved
     Deprogram messages
     If use stop messages, program stop messages
Send programmed messages
```

Fig. 5. Agents step QW algorithm.

4.1 Knowledge exchanges dataset

In the knowledge exchanges (KE) dataset simulations, the data has been randomly generated. We configured the simulator with the following requirements:

- The expertise vector k of each agent has dimension 5. The value of $k_j \in [0,1]$ of an expertise vector $k = \{k_1, k_2, k_3, k_4, k_5\}$ means the expertise level in a domain j. In the initial conditions, k values are set randomly.
- At each step, each agent has a question probability of having its own question (Ψ). In our simulations, $\Psi = .05$.
- The interest vector $It=\{i_1, i_2, i_3, i_4, i_5\}$ has the same dimension as vector k. It denotes the probability that a question will belong to a domain j. $\sum_j i_j = 1$.
- Each execution consists of 2,000 simulation steps with 100 agents.
- The social network is represented by an undirected graph, although each connection has two different trust values. We used the social networks represented in Fig. 6. SN-a has 495 links, a clustering coefficient of .521, an assortativity of .0763, a diameter of 5, and an average shortest path length of 2.84. SN-b has 395 links, a clustering coefficient of .386, an assortativity of .0568, a diameter of 5, and an average shortest path length of 2.86.
- The initial trust from each agent to its acquaintances is .75 for each domain.
- We executed each configuration 20 times.
- The questioning agents rate the answers received at their ϑ value.
- Each query belongs only to one domain.

- In all three of the cases, we consider that the probability that an answer satisfies a question is p(ϑ)= $\vartheta/100$.

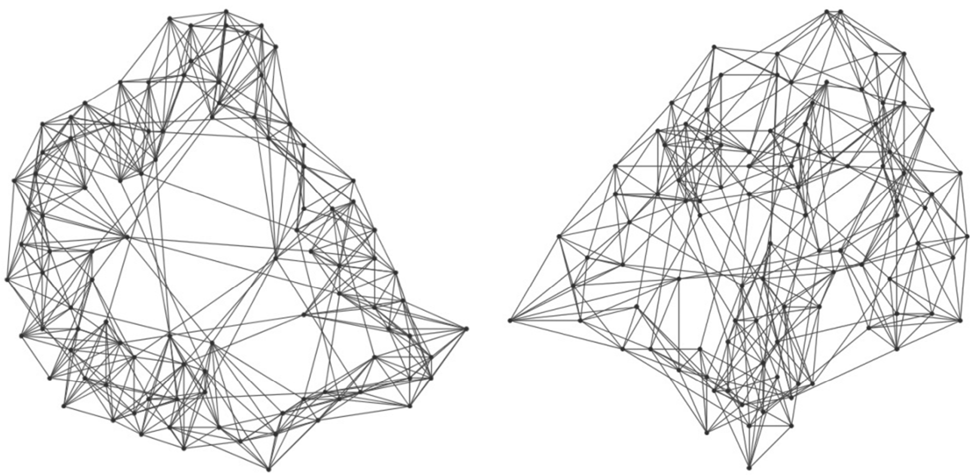


Fig. 6. Social networks used. The network on the left is SN-a, and the network on the right is SN-b.

4.1.1 *Answer relevance and trust model*

In these simulations, the answer quality (ϑ) is computed in three different ways:

- KM0: Use the knowledge model used by [36,37], expressed in Eq. 4, where I represents the implication, k is the expertise, and R is the external factors represented randomly, with the settings $w_i = .2$, $w_k = .7$, and w_R= .1
- KM1: Use the above knowledge model with the settings $w_i = .4$, $w_k = .5$, and $w_R = .1$

$$\vartheta = w_i I + w_k\, k + w_R R \tag{4}$$

- KM2: Use a knowledge model in which an agent with an expertise of e will produce a ϑ of e when the energy used to generate an answer is ∞ , with $E=10$, $\vartheta = \frac{k}{2}$, with $E=50$, $\vartheta = .9k$ and with $E=0$, $\vartheta = 0$ (Eq. 5). The model is inspired by the ART-Testbed [12], in which the authors focus on the standard deviation produced instead of ϑ.

$$\vartheta = k\,\frac{E^2}{E^2+\frac{40}{9}E+\frac{500}{9}} \tag{5}$$

Agents evaluate their acquaintances for each domain considering the mean ϑ of the first answer provided. When agents have to determine the amount of effort that they will allocate to obtaining answers or generating an answer, the agents use the highest trust value.

4.1.2 *Question delay and delaying own answer*

Although there are many ways to delay questions and answers based on estimations of their relevance, we considered that there are two important points. The first one is that the answers that are more likely to be relevant should arrive quickly, and the time that users have to wait should be short. The second one is that answers that are more likely to be not relevant have to be sent only when there are no other answers available, so their delay should increase considerably. With this purpose, in our simulations, we used Eq. 6, and the delay distribution based on trust can be seen in Fig. 7. With these function, with a $T_{i,j}$ of {.9,.8,.6} $t_{i,j}$ are ≍{2,5,15} respectively.

$$t_{i,j} = \frac{1}{3}\left(-150\, {T_{i,j}}^{3} + 550\, {T_{i,j}}^{2} - 700\, T_{i,j} + 300\right) \tag{6}$$

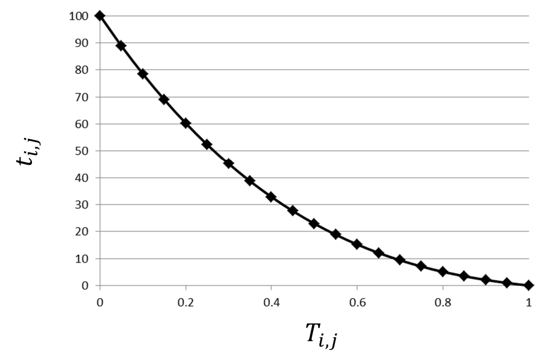


Fig.7. Question and answering delay ($t_{i,j}$) in simulation steps in function of trust ($T_{i,j}$).

4.1.3 Effort settings

In these simulations, we used 3 effort settings:

- ES0: The more trusted acquaintances receive more effort. An acquaintance a_j receives an amount of effort $E_{i,j}$ from the agent a_i based on the trust that a_i has in a_j $T_{i,j}$, the effort that a_i can transmit E_i and in comparison to the trust that a_i has of the other acquaintances. $E_{i,j} = E_i \, {T_{i,j}^2}\big/{\sum_k T_{i,k}^2}$.
- ES1: The effort is distributed equally among the acquaintances.
- ES2: The effort used to try to answer a question is infinite.

The initial effort (E_0) in ES0 and ES1 is 10^5, and the effort costs are 2 for planning how to answer a question, including checking if would be possible to answer the question without forwarding it, selecting how to distribute the effort and sending messages. The costs do not include the cost of generating an answer, for which the maximum value will be 90 in ES0 and ES1.The effort loss due to *friction* is expressed in Eq. 7, were E_j is the effort that agent a_j will use to try to solve the question while $E_{i,j}$ is the effort transmitted by a_i to a_j (so, before applying friction). We used $\mu = .1$ because we assumed that the maximum loss due to friction for these simulations may be 10%, which will happen when the trust value is 0.

$$E_j = E_{i,j}\left(1 - \mu\,(1 - T_{i,j})\right) \tag{7}$$

4.1.4 Interferences

In the following simulations, we only considered destructive interferences. As a function of the effort used to obtain answers, each agent will continue seeking answers until the answers reach an estimated probability of answering the question correctly (σ). When an agent reaches σ, it will not put more effort into trying to obtain answers to the question, and it may send a stop message to the agents to which it delegated the question. When an agent receives a stop message, the agent will not put more effort into trying to answer the question. However, in both cases, agents may continue forwarding answers received. We consider that σ must tend to 1 when E tends to ∞ and that σ must be 0 when $E = 0$. Furthermore, considering that $E_0 = 10^5$, we designed σ to be .9 when $E = E_0$, and σ when $E = E_0/2$. An approximated function is Eq. 8. In this equation, we used E' instead of E, where $E' = \varphi E$. The idea is to use φ to modify agents sensibility. $\varphi \in [0,\infty]$, values of φ lower than 1 suggest that the agents have a lower σ, and they decide to stop their contribution faster. In contrast, values of φ greater than 1 suggest that agents are more conservative, and they will need more answers to end their contribution.

$$\sigma = \frac{E'^2}{E'^2 - 2.78\ \ 10^4 E' + 3.89\ \ 10^9} \qquad (8)$$

4.1.5 Evaluation and results

With the aim of evaluating the correlation of answer speediness with ϑ (- Delay), we added other correlations that would not be as easy to use in a real system and would be needed to re-rank the results:

- the distance to the answering agent (D)
- the reputation of the acquaintance that sends the answer (R)
- the transitive reputation of the path (TR). We computed this factor as the product of the reputation values of the path
- the reputation of the answerer (RA)
- the expertise of the answerer (Ex), directly based on the expertise value with which the answers are generated.

To compute these correlations, we used simulations without waves. The results are shown in Table 1. As expected, Ex has the best correlation with ϑ in the simulations with KM0, as Ex has a weight of .7 to ϑ. RA has results close to Ex, and the correlation with answer speed is high, but D and R provide no correlation in this group. There are no significant differences in the ES and networks for this KM.

Considering KM1, the results are similar to KM0, but there are weaker correlations for TR, Ex, RA and -Delay. Also, the results seem to indicate that there is a slightly higher correlation with Time and TR for the second network than the first.

Considering KM2, with finite effort (ES0 and ES1), D has some correlation, TR has the best correlation for network SN-a, while -Delay has the best correlation for network SN-b. In the case of infinite effort, the simulation has a similar behaviour as KM0 and KM1.

Additionally, Fig. 8. shows $\partial P(t)$ for each ES and KM; we can observe that it follows a similar pattern than the presented in Subsection 3.1 (Fig. 3. (c)), where after a setup time, the probability of finding a relevant answer at time instant t is inversely proportional to t.

Table 1.
QW results.

KM	E. S.	Network	D	R	TR	Ex	RA	-Delay
0	0	SN-a	-.028	.090	.446	.985	.984	.935
0	0	SN-b	-.010	.065	.447	.985	.984	.937
0	1	SN-a	.003	.098	.441	.985	.985	.935
0	1	SN-b	.001	.076	.445	.985	.985	.936
0	2	SN-a	.002	.098	.435	.985	.985	.936
0	2	SN-a	.001	.076	.442	.985	.985	.937
1	0	SN-a	-.010	.092	.341	.954	.955	.909
1	0	SN-b	.0	.078	.352	.953	.955	.912
1	1	SN-a	.003	.101	.346	.954	.956	.909
1	1	SN-b	.002	.080	.353	.953	.955	.912
1	2	SN-a	.003	.100	.341	.954	.956	.908
1	2	SN-b	.002	.080	.350	.953	.955	.912
2	0	SN-a	.540	.119	.776	.573	.599	.759
2	0	SN-b	.448	.133	.749	.670	.684	.811
2	1	SN-a	.627	.055	.668	.586	.617	.612
2	1	SN-b	.514	.056	.632	.699	.711	.736
2	2	SN-a	.002	.095	.488	1	.999	.953
2	2	SN-b	.001	.073	.488	1	.999	.953

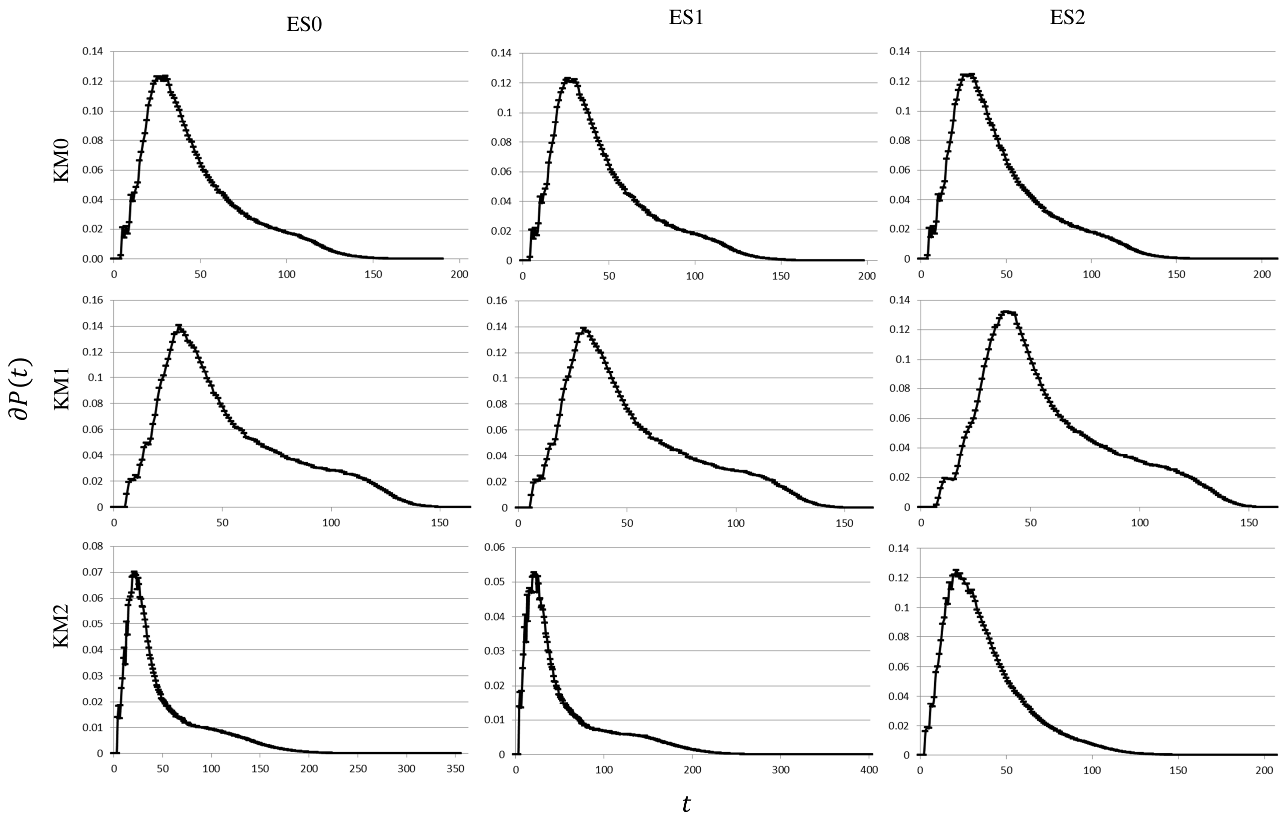


Fig. 8. Probability of finding a relevant answer at time t, $\partial P(t)$, as a function of time. The x axis shows the time (in simulation steps), and the y axis shows the probability.

Table 2.
Results using interferences.

	φ	-Delay	$\bar{\vartheta}$	m (10^6)	*a/q*
Without Interferences	No	.935	.569	4.77	97.35
With Interferences	4	.838	.671	3.304	42.958
	2	.804	.687	2.677	35.34
	1	.757	.714	2.586	30.731
	.5	.544	.792	1.726	17.19
	.25	.488	.804	1.43	14.34

To show the behaviour of interferences, we used KM0, SN-a and stop messages that will be sent when the probability of having a good answer is higher than τ (Eq. 8). The results can be seen in Table 2, where we can observe that when we increase φ, there are more answers per question (*a/q*), but the mean quality of the answers ($\bar{\vartheta}$) is decreased. In contrast, decreasing φ makes less (*a/q*), but $\bar{\vartheta}$ is increased. The correlation with -Delay is reduced when *a/q* is reduced. We see several explanations for this pattern. The first one, as can be seen in Fig. 7, is that the difference of the time delays for higher trust values is really small; a second one is that the R factor takes more importance when the expertise rankings are similar. The third explanation is that the distance to reach agents with higher expertise also affects the arrival time and has more importance when we only receive a few answers. This loss of correlation is not really important because the system provides the best answers, as can be seen for the increase of $\bar{\vartheta}$.

4.2 Experiments with real data extracted from a recommender systems dataset

It is not straightforward to use real data with questions answered and rated by users because on Q&A sites the same question is often repeated, and it is difficult to determine which questions are equivalent.

Some of the protocols referred to in this paper [40,41,44] used generated data especially suited for their experiments. For example, [40,44] set up a few categories and generated random profiles that were used for one [44] or more agents [40], while the queries were restricted to one category only. The answer suitability was based on user expertise evaluated at random [40] or on the similarity among user profiles [40]. In Sixearch [41], the authors based the content on real data, categories and answers (that consisted of real webpages), although they modelled the users randomly. In all of the cases, they generated random test users.

Using a real social network with real data for this kind of simulation is not straightforward because there is no available real data pool that contains all of the following points:

- A real social network in which the links among users indicate relationships (friends, colleagues, or family).
- A set of questions sent by the users of that social network and a set of similar or equivalent questions sent to one another, based on the idea that more than one user asks the same question.
- A set of answers for different questions provided by one user, which should be rated by a considerable number of users, not only a few.

Some exceptions might be Twitter and Quora. In the case of Twitter, and compared to other social networks, it might be easier to extract information, being it done through hoses (accessible through API), and several datasets have been published in the very recent years (e.g. [21]). In Twitter, there are less question answering interactions compared to other social networks yet there is vital exchange of comments and opinions among their users that are following each other. As said, Twitter is not a platform for question answering, though its open contact lists are very advantageous for our research, as we can create a replica of the social network behind and do experimentation. In the case of Quora, users also follow others and its usage is well focused on Q&A, and as an additional strong feature, Quora does suggest similar questions to guide the question preparation, yet it does not require the existence of any path in the social network from the questioner to the answerer (that is, potential answerers are accessible directly), and finally we did not find any dataset with our required data (contact lists, questions and answers). In all, Quora did not fit our experimentation needs for our approach of social search research, yet we are open to extend our results to that platform soon in the future. Due to the lack of a true social question-answering environment useful for experimentation, we reused existing datasets used for information retrieval or recommender systems. Two of the most popular datasets are Text REtrieval Conference (TREC[6]) and movielens[7]. The first dataset contains a set of textual questions with a set of documents that may contain the answer. However, TREC datasets have no user ratings and no users who would encourage the reconstruction of a possible social network on it. In the case of movielens, the identified users rate a set of items. In this case, we can consider an item as a type of question and a rating as a type of answer, so there are many answers for each question. It is also possible to use recommendation techniques to evaluate answer suitability for a user, which equivalent to having the answers rated. The main item missing is a true social network, which could be generated randomly or using similarity functions [26]. For these experiments, we consider it better to generate the social network randomly, although considering that the network has social network properties. The main motivation is that in QW, we consider that people should have a network in which most of their contacts are people that they know as friends, family, colleagues, etc. An additional reason is that to identify similar profiles from unknown users, a centralized approach or some method to construct this similarity network that is out of the scope of this paper would be required.

Social search engines and recommendation systems (collaborative filtering or opinion-based filtering) are converging, as we consider that recommendation systems must be classified as SFS according to the classification of social search systems by [10], more concretely when we consider RWC recommender systems [40], which provide a personalized list of items. In SFS and RWC, users receive a list of items sorted with the aim to fit the user's taste. Therefore, using a dataset originally designed as a recommender-system dataset for social search simulations is appropriate for experimenting with social search because recommender systems can be seen as a subtype of social search (like an SFS, according to Chi [10]). Ratings gathered in these databases can be considered subjective answers to open questions like "What do you think of Gone With the Wind?", to which one can answer "I love it", "I hate it" or "It is all right, but not that much".

We decided to use the movielens dataset, consisting of 100,000 ratings for 1,682 movies produced by 943 users. The ratings are split into two sets: a training or base set with 80% of the ratings and a test set with the remaining 20%. With the movielens dataset, the only aspect missing is the social network, which is generated at random.

In the following Subsection, we will explain how we prepared the data for the simulations and, briefly, the social networks used. In Subsection 4.3.2, we will briefly explain collaborative filtering recommender systems, then we will explain how we used Question Waves with these data, and finally, we will show our results.

4.2.1 Simulation data

We used the data that we prepared to use with Asknext [38], it was as follows:

- We have removed the users who had fewer than 10 questions in the test set because many users had no questions.
- We have a queue of questions generated by distributing them between 1 and 1,000 steps of simulation.
- All of the simulations end at step 2,000.

As a result, the number of questions is 19,463, and there are 387 users. We generated 2 social networks, SN-1 and SN-2 (Fig. 9). SN-1 has 3,089 links, a clustering coefficient of .53, an assortativity of .0401, an average path length of 3.02 and a diameter of 5. SN-2 has 1,933 links, a clustering coefficient of .515, an assortativity of .0277 and average path length of 3.905 and a diameter of 7.

[6] http://trec.nist.gov/data/qa.html
[7] http://www.grouplens.org/node/73

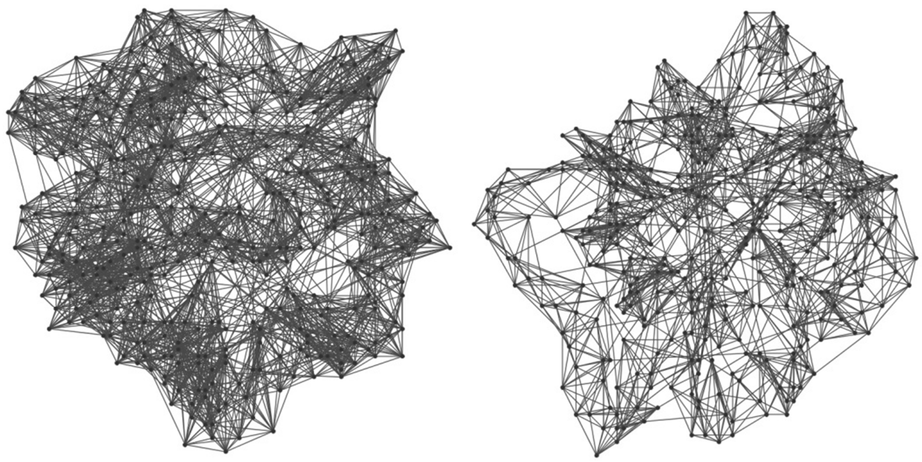


Fig. 9. Networks used in the simulations with movielens data. The network on the left is SN-1, and the network on the right is SN-2.

4.2.2 Collaborative filtering recommender systems

Collaborative filtering (CF) is used to recommend items that similar users have liked. CF is based on the social practice of exchange opinions. In the 1990s, GroupLens worked with this family of algorithms [20,30], leading a research line on recommender systems that expanded globally because of growing interest in the Internet. Usually, these systems rate items between 1 and 5. The most popular measures to estimate the similarity of pairs of users are the cosine [7,31] (Eq. 9) and Pearson's correlation coefficient (Eq. 10) [30,32].

$$u(t,v) = \frac{\sum_{i \in Im}(r_{t,i})(r_{v,i})}{\sqrt{\sum_{i \in Im}(r_{t,i})^2 \sum_{i \in I}(r_{v,i})^2}} \tag{9}$$

$$u(t,v) = \frac{\sum_{i \in Im}(r_{t,i} - \overline{r_t})(r_{v,i} - \overline{r_v})}{\sqrt{\sum_{i \in Im}(r_{t,i} - \overline{r_t})^2 \sum_{i \in I}(r_{v,i} - \overline{r_v})^2}} \tag{10}$$

Where:

- $u(t,v)$ is the similarity of owners of the agents a_t and a_v.
- $r_{t,i}$ is the rating given by owner of the agent a_t to item im_i.
- *Im* is the set of items.
- $\overline{r_t}$ is the mean of the ratings of owner of the agent a_t.

There are other measures, for example, the PIP proposed by Ahn [2] that is composed of three factors: proximity, impact, and popularity. A PIP score is computed for each co-rated item, and the similarity between two users is the sum of these PIP scores. This measure takes into account the distance between the co-rated items (proximity), gives more significance to ratings in the extremes of the rating range (impact) and also increases the importance of similar ratings that are far from the mean rating of the items (popularity).

The owner of the agent a_i rating item im_j ($r_{i,j}$) can be predicted with Eq. 11. In this equation, the constant *K* is used to match the prediction within the range of values.

$$r_{i,j} = \overline{r_i} + K \sum_{r_{k,j} \mid k \neq i} u_{i,k}\left(r_{k,j} - \overline{r_k}\right) \tag{11}$$

The main benefits of this method compared to another dominant recommendation technique, content-based filtering (CBF), are that items can be recommended without knowing their features, and users with hard-to-describe taste can still be classified with accurate recommendations. The drawbacks are that this method needs a lot of ratings and suffers from problems of cold starts and scattering data.

As alternative evaluation measures to CBF, the mean absolute error (MAE), mean squared error (MSE) or rooted MSE (RMSE) are well suited for simulated experiments and are good alternatives for measuring the suitability of answers or recommendations. The MAE, MSE, and RMSE show the mean error of the predictions; to achieve relevance on this kind of SFS, it is important to obtain low errors.

Once we have introduced the different metrics and concepts related to CF, in the following Subsection, we will explain how we simulated the network and show the results that were obtained.

4.2.3 Simulating a P2P Recommender System with Question Waves

In the literature, there are some examples of P2P recommender systems. The first ones, proposed by Walter et al. [40] and Montaner [26], were mentioned previously. Other ones based on BFS are [4,11,29]. The objective of this Subsection is to show the benefits of using QW in a more real scenario where it can be applied. To study the answer suitability offered by Asknext, we use an algorithm similar to traditional CF. How this proposal works is beyond the scope of this paper, except that because these CF systems are BFS based, Question Waves can be used with them.

To use Question Waves as a P2P recommender system, we determined the following:

- The question message includes the questioner profile.
- In the answer field of the answer message, we added two fields:
  - Accumulated vector: $\sum_{r_{k,j} \mid k \neq i} u_{i,k}\left(r_{k,j} - \overline{r_k}\right)$
  - Similarity vector: $\sum_{r_{k,j} \mid k \neq i} u_{i,k}$
- The prediction of $r_{i,j}$ is performed with Eq. 11 using $K = 1/(\, a^{1/2} \sum_{r_{k,j} \mid k \neq i} u_{i,k})$, where a is the number of answers.

When we applied CF, we obtained an MAE of .765 and an MSE of .958. It was possible to predict 19,210 items. We use these values as a reference target for our implementation of QW.

We modelled the QW as follows:

- In the simulations, we used Pearson's similarity measure between questioners and answerers to measure answer relevance.
- We added a domain for each possible questioner, and agents determine how good a mediator is at providing answers for a profile similar to a concrete questioner.
- The question time and the delay of the own answer are determined with a piecewise function (Eq. 12)
- The initial effort used by the questioners is $E_0 = 10^5$. The effort is distributed equally to all of the agents, and there is no cost of answering, but we maintained the same friction cost of Eq. 7, as in Subsection 4.1.3.
- When a mediator sends back an answer, the mediator will check if it has achieved its objective for that task; if so, it will send stop messages and will stop forwarding the questions related to that task. The objective for the task (σ) depends on the effort applied to obtain an answer ($E' = \varphi E_0$), computed with Eq. 13. φ has the same meaning as in Subsection 4.1. We used $\varphi = \{.25, .5625, 1, 1.5625, 2.25, 3.0625, 4\}$.

$$t_{i,j} = \begin{cases} 1\ if\ T_{i,j} > .9 \\ 2\ if\ .9 \geq T_{i,j} > .8 \\ 3\ if\ .8 \geq T_{i,j} > .7 \\ 4\ if\ .7 \geq T_{i,j}\ > .6 \\ 5\ if\ .6\ \geq T_{i,j} > .5 \\ 10\ if\ .5 \geq T_{i,j} > .3 \\ 15\ if\ .3 \geq T_{i,j} > .1 \\ 20\ if\ .1 \geq T_{i,j} > -.3 \\ 25\ if\ -.3 \geq T_{i,j} > -.5 \\ 40\ if -.5\ \geq T_{i,j} \end{cases} \tag{12}$$

$$\sigma = \frac{\sqrt{E'}}{\sqrt{E_0}} \tag{13}$$

4.2.4 Results

In Fig. 10, one can see the results of applying QW as a collaborative filtering system. Fig. 10 (a) and (b) show the MAE and MSE, respectively; the system provides slightly fewer errors using QW, and the results are getting closer of the reference values of CF (MAE of .765 and MSE of .958) along with the effort applied. Fig. 10 (c) shows the number of messages sent, where one can see that the configurations without waves (A1 and A2) have a higher slope. A2 starts using fewer messages than QW2, but due to the higher slope, the simulations have a similar number of messages at $\varphi = 1$.

Fig. 10 (d) shows the number of questions being answered. The variation between A and QW is not significant; in some cases, A does more predictions and vice versa.

Table 3.
Correlations of error prediction with time and similarity.

	$-\text{Delay}_{\text{SN-1}}$	$-\text{Delay}_{\text{SN-2}}$	Similarity
Mean correlation	-.5759	-.62	-.5934
St. dev.	.075	.08	0

We computed the correlation of the error produced by each answer message and the answer delay, and we obtained a correlation of 57.59% for SN-1 with a standard deviation of .075 and 62% for SN-2 with a standard deviation of .08, as can be checked in Table 3. We also computed the correlation of the error produced and users' similarity (similarity has been computed with Eq. 10) and we obtained -59.34%. We can claim that the correlation obtained between the answer relevance and answer speed (that is the inverse of the answer delay) is equivalent to the correlation of users' similarity and answer relevance. The similarity is the standard reference in a centralised CF recommender system. Our results of QW using partial information are comparable to a CF system that has information full access.

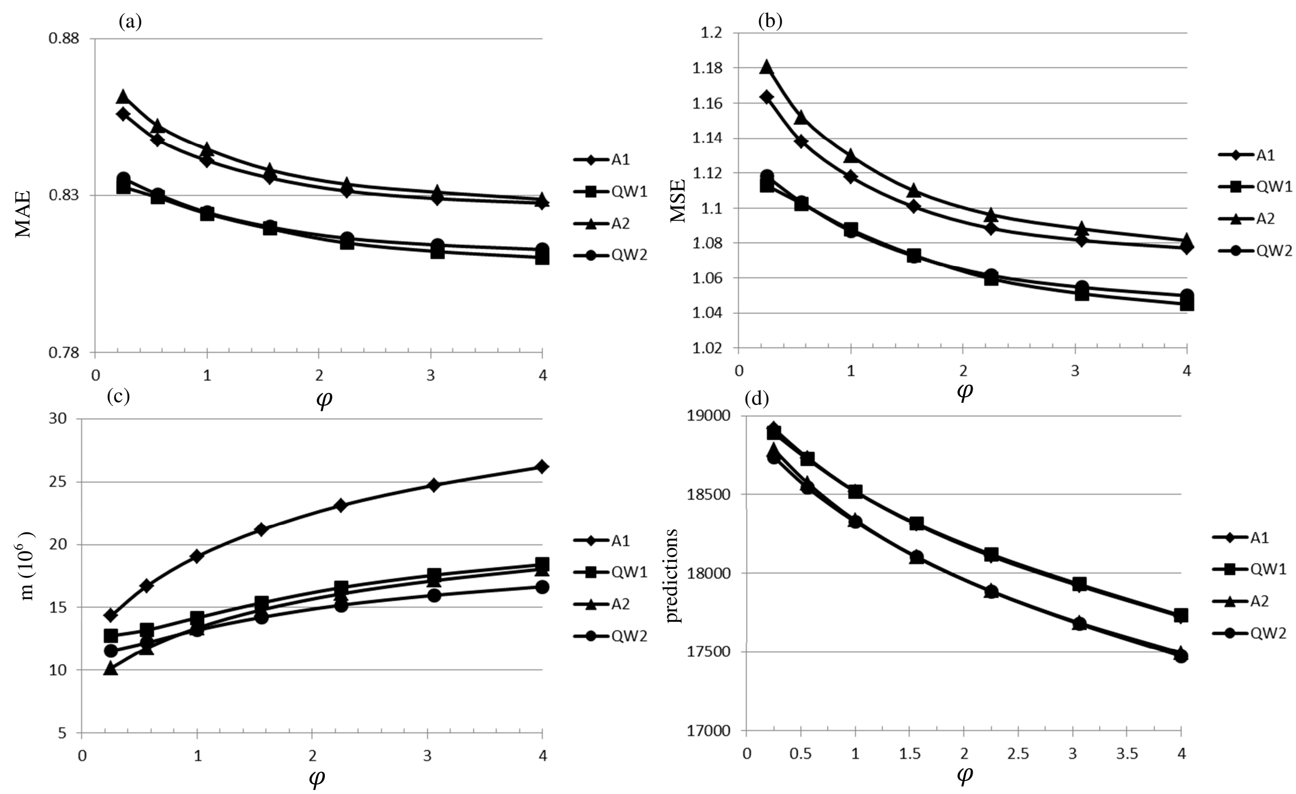


Fig. 10. Results of applying QW as a CF, comparing the results of using waves (QW1 and QW2) and not using waves (A1 and A2). (a) shows the MAE, (b) the MSE, (c) the number of messages and (d) the number of predictions done, all as a function of φ.

4.3 Practical Considerations

At the beginning of Section 3, we explained that the best behaviour of query-routing would be requesting to the minimum number of agents and obtaining the maximum or enough number of relevant answers. We perform now a set of experiments to measure the cost of obtaining relevant answers regarding the effort used and the number of generated messages, to test what configuration of QW is behaving best, and whether they are in all behaving better than the other algorithms of social search.

We compare 4 configurations:

- The first configuration (BFSe), is BFS with the improvement of effort to limit the question propagation.
- A second configuration (SMe) uses stop messages, as Asknext does; and it uses effort to limit the question propagation.
- A third configuration (QW) introduces the effects of answer and question messages delay but does not use stop messages.
- The fourth configuration (QW-SM) uses answer and question delay messages and uses stop messages.

The only input for every simulation is the initial effort per question, while the outputs are the following:

- Number of messages generated (m). In the introduction, we talked about how many agents are requested. Now, more precisely, the number of messages represents how many times agents are requested.
- Proportion of relevant answers obtained from relevant answers available (recall). Answers would be considered as relevant when their ϑ is above a threshold χ.
- Proportion of relevant answers obtained in order of arrival within a window of size n (precision). We considered size n=3.

The thresholds χ that we used are {.95, .85, .5} for the KE dataset, and {.95, .75, .45} for the movielens dataset creating the scenarios of {very high, high, and medium χ} for every configuration and dataset. The networks used are SN-a and SN-1. Furthermore, we ensured that all compared models had a same number of relevant answers available: in the case of the KE dataset, we computed the answer relevance only with the expertise, which is equivalent to Eq. 4 with $w_I = 0$, $w_k = 1$, and $w_R = 0$.

Fig. 11 (a) and (b) shows the number of messages in function of the effort for KE dataset and movielens dataset respectively. In the KE dataset, we can see that QW and QW-SM start with higher number of messages than BFSe and SMe, but their convergence is slower; they converge to the highest number of messages because answers might follow longest paths than in BFSe and SMe. In the KE dataset, stop messages do not reduce the number of messages. In the case of the movielens dataset, the usage of stop messages shows some benefits, although in the case of QW-SM there are some effort values where the number of messages goes over the steady value of the number of messages. In this scenario, BFSe shows faster convergence, while SMe shows the slowest convergence.

Fig. 12 (a), (b) and (c), shows the number of relevant answers for the KE dataset with χ = {.95, .85, .5} respectively. We can observe than BFSe and SMe follow a same behaviour. In the cases of QW and QW-SM, in the initial values, QW-SM obtains few more answers, but with χ = {.90, .85} QW shows faster convergence.

Fig. 12 (d), (e) and (f), shows the number of relevant answers for the movielens dataset, with {.95, .75, .45} respectively. In this scenario, we can see that the behaviour of QW is equivalent to QW-SM, while BFSe is equivalent to SMe. Also, it is possible to observe that the QW and QW-SM converge faster, yet the convergence gap of QW (and QW-SM) and BFSe and SMe get narrower when χ is being decreased (from very high to medium values) so that in Fig. 12 (f) (for medium χ) the convergence behaviours are nearly equivalent for all configurations.

In Fig. 11 (b) and Fig. 12 (d, e, f) corresponding to the movielens dataset, we can clearly appreciate that the recall and the number of messages belong to a saturated growth model.

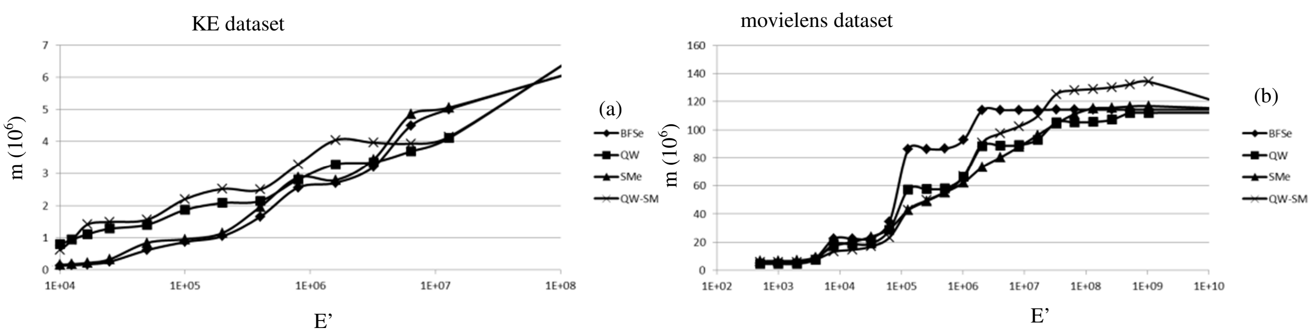


Fig 11. Number of messages in function of effort. For the KE dataset (a) and movielens dataset (b).

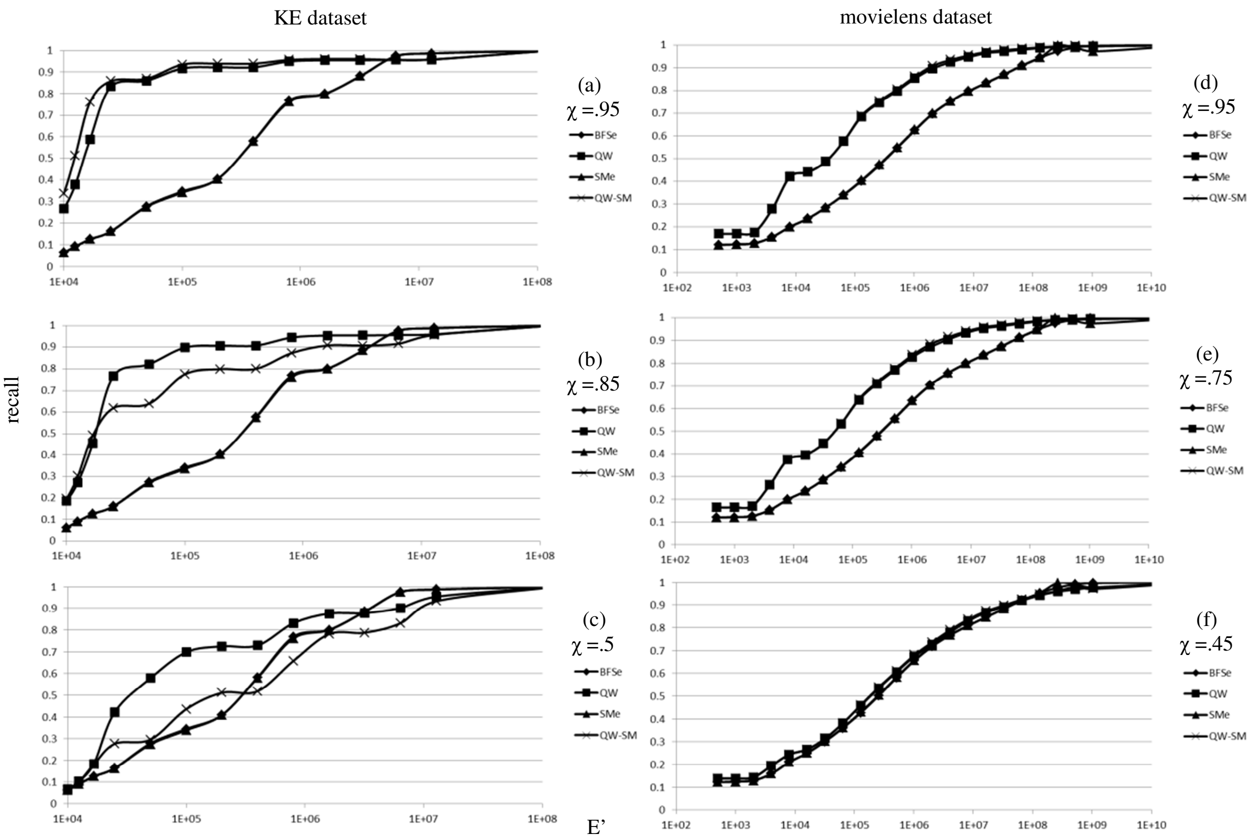


Fig. 12. Recall (y-axis) in function of effort (x-axis) for the KE dataset (a) (b) and (c) with χ of .95, .85 and .5 respectively, and for the movielens dataset (d), (e) and (f) with χ of .95, .75 and .45 respectively.

Considering that we want to obtain a good recall, let us consider it good a recall of at least 75%; the following configurations hit that figure of recalls:

- KE dataset: BFSe and SMe obtain it (the good recall) with an effort of $8*10^5$. QW-SM obtains it at efforts of $\{1.66*10^4, 10^5, 1.6*10^6\}$ with a χ of .95, .85 and .5 respectively, and QW obtains it with efforts of $2.5*10^4$ with a χ of .95 and .85, while it needs an effort of $8*10^5$ to achieve it with a χ of .5.
- Movielens dataset: BFSe and SMe obtain it with an effort of $4.1*10^6$ for the 3 values of χ, while QW and QW-SM depend of χ, they obtain it with efforts of $2.56*10^5$, $5.12*10^5$ and $4.1*10^6$ with a χ of .95, .75 and .45 respectively.

For better explaining the relation between the number of relevant answers found in comparison to the number of messages with the purpose of comparing the configurations, we approximate the relation between the recall and the number of messages as a saturated growth model (Eq. 14), where τ is the time constant and A the saturated (or final) value of y in function of t.

$$y\,(t) = A(1 - \mathrm{e}^{-\mathrm{t}/\tau}) \tag{14}$$

This approximation works well for the recall in function of the number of messages, except for the tail and head of the function; the most relevant behaviour is approximated with Eq. 15, inspired by [27].

$$r\ (m) = 1 - \mathrm{e}^{-\mathrm{m}/\tau} \tag{15}$$

Where:

- m is the number of messages.
- r is the *recall* estimated model. It must converge to 100%.
- τ is the constant that represents the recall model convergence, and it will be referred to as the *number of messages constant*. It has the property that $r = (1 - e^{-1}) \doteq .632$ at $m = \tau$ that indicates that with τ messages we obtain a recall of .632. Dividing the τ of the several configurations we can obtain how many times more messages a configuration needs to obtain a same recall *r* compared to the other configurations.

We used the method of Newton-Rapshon to obtain the τ of the regression model for Eq. 15 that minimizes the sum or squared error (S) of the real and the modelled recalls. The results of the τ of the 4 configurations (BFSe, QW, SMe, QW-SM) at the 3 scenarios (very high, high, and medium χ) and 2 datasets (KE and movielens) are shown in Table 4.

Table 4.
τ and sum of squared error (S) for each configuration in function of χ and dataset.

		KE dataset			movielens dataset		
		χ =.95	χ =.85	χ =.5	χ =.95	χ =.75	χ =.45
BFSe	$\tau\ (x10^6)$	1.84	1.85	1.83	111.11	109.26	101.59
	S	.00621	.00680	.00637	.0521	.0516	.0538
QW	$\tau\ (x10^6)$	1.17	1.35	2.22	35.25	40.12	66.16
	S	.222	.305	.282	.0306	.0316	.0317
SMe	$\tau\ (x10^6)$	2.11	2.13	2.11	71.46	65.9	70.4
	S	.0107	.0118	.0111	.00612	.00654	.00625
QW-SM	$\tau\ (x10^6)$	1.03	1.76	3.8	29.3	33.56	56.73
	S	.0483	.0423	.0686	.0246	.0224	.0199

The models corresponding τ of Table 4 are shown on Fig. 13.

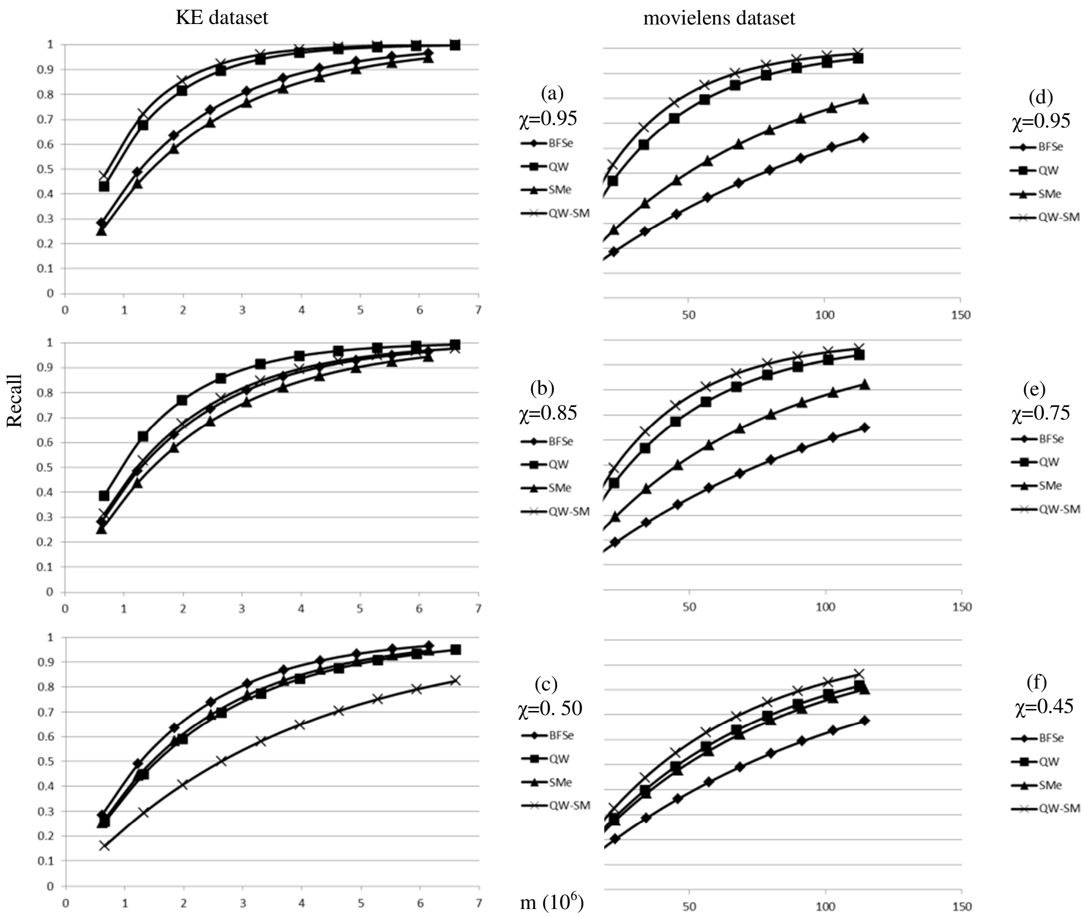


Fig. 13. Modelled recall (y-axis) in function of number of messages (x-axis). For KE dataset ((a), (b) and (c)) and movielens dataset ((c), (d) and (e)), with different χ values.

In Table 5 we show the quotient q of the τ of pairs of configurations. Quotient $q = \tau_i/\tau_j$ means that configuration i hits a same recall in q times the number of messages of configuration j, thus the lower q the lower number of messages are necessary for configuration i to converge compared to configuration j. We can observe that, in the KE dataset scenarios of χ={.95, .85}, QW needs the least number of messages to provide a same number of relevant answers (that is it has the lowest τ and the lowest quotient q), and that SMe is the configuration that needs to use more messages to provide the same number of relevant answers; in the other hand with χ=.85 BFS needs fewer messages to provide the same number of relevant answers.

Observing the outputs of the movielens dataset, we can observe that with all χ values the quotient of τ (QW-SM) divided by the τ of the other configurations is the lowest, thus QW-SM requires lesser messages to provide a same number of relevant answers than the other configurations. However, this advantage diminishes when the χ is decreased; this is due to the effect of σ, because in QW the answers arrive sorted by relevance and the agents decide that they do not need more answers sooner that in consequence the agents are satisfied with lower recall.

Table 5.
Quotient between the τ of the different configurations in function of the χ and dataset.

	KE dataset						movielens dataset				
	quotient	**BFSe**	**QW**	**SMe**	**QW-SM**		***quotient***	**BFSe**	**QW**	**SMe**	**QW-SM**
	BFSe	1	1.57	.871	1.79		**BFSe**	1	3.15	1.56	3.79
χ =.95	**QW**	.637	1	.555	1.14	χ =.95	**QW**	.317	1	.493	1.2
	SMe	1.15	1.8	1	2.06		**SMe**	.643	2.03	1	2.44
	QW-SM	.56	.877	.487	1		**QW-SM**	.264	.831	.410	1
	quotient	**BFSe**	**QW**	**SMe**	**QW-SM**		***quotient***	**BFSe**	**QW**	**SMe**	**QW-SM**
	BFSe	1	1.37	.87	1.05		**BFSe**	1	2.72	1.55	3.26
χ =.85	**QW**	.732	1	.637	.771	χ =.75	**QW**	.367	1	.57	1.2
	SMe	1.15	1.57	1	1.21		**SMe**	.644	1.76	1	2.1
	QW-SM	.949	1.3	.825	1		**QW-SM**	.307	.837	.477	1
	quotient	**BFSe**	**QW**	**SMe**	**QW-SM**		***quotient***	**BFSe**	**QW**	**SMe**	**QW-SM**
	BFSe	1	.827	.869	.483		**BFSe**	1	1.54	1.54	1.79
χ =.5	**QW**	1.21	1	1.05	.584	χ =.45	**QW**	.651	1	1	1.17
	SMe	1.15	0.951	1	.556		**SMe**	.649	1	1	1.16
	QW-SM	2.07	1.712	1.8	1		**QW-SM**	.558	.857	.861	1

A new measure has to be analysed for engineering reasons: Precision. It is easier and more practical to be measured than recall. Recall is in fact only measurable in simulation. Precision is measured as the fulfilment out of a window of answers. We took a window size of 3 answers. The results obtained shown in Fig. 14; (a) and (b) are the proportion for the KE dataset with χ={.95, .85} respectively while (c) and (d) are the values for the movielens dataset with χ={.95,.45} respectively. In all cases, QW and QW-SM show lower quotient than BFSe and SMe; the explanation is that with BFSe and SMe the answers do not come sorted by relevance, as it does happen with QW and QW-SM. In the KE dataset, we can see that stop messages help to increase this proportion while using QW, although that the convergence value is the same.

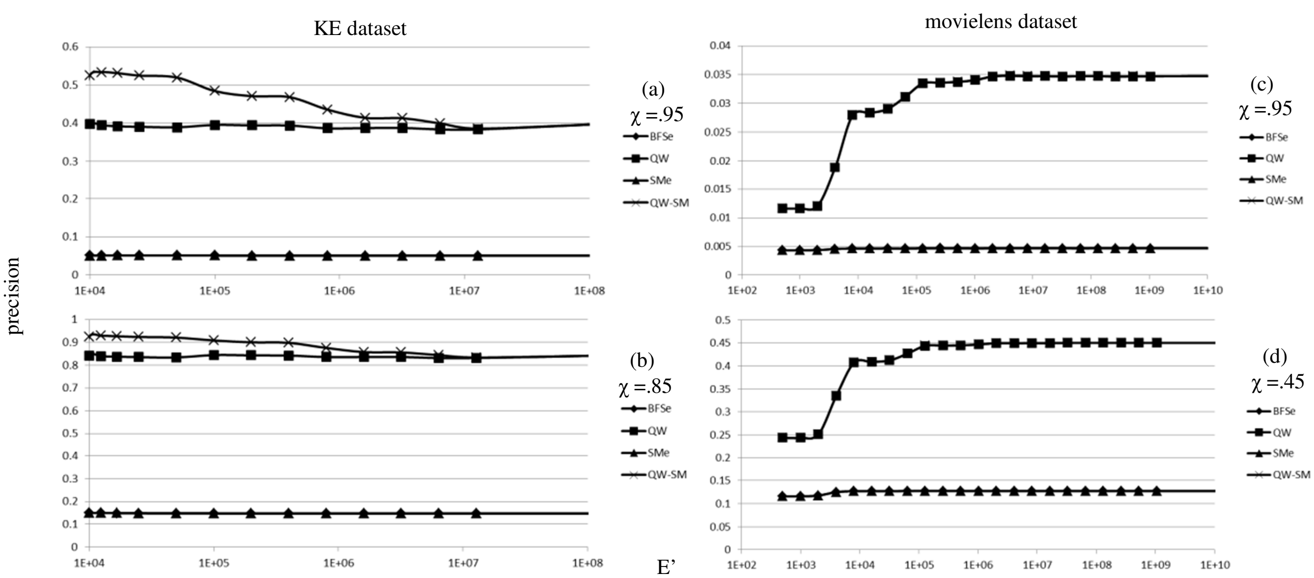


Fig. 14. Precision (in a window size of n=3) in function of effort for KE dataset ((a) and (b), with χ=.95 and χ=.85 respectively) and movielens dataset ((c) and (d) with a χ=.95 and χ=.45 respectively).

Considering Fig. 14, the best performance of KE dataset is achieved by QW-SM with an effort of 10^4, while in the movielens dataset it is achieved by QW and QW-SM with an effort of 10^5 when χ is .45 and with and effort of $4*10^6$ when χ is .95.

QW and QW-SM results can be modelled using Eq. 14 resulting into a τ (which is the *effort constant*) of $2.77*10^3$ when χ is .45 and a τ of $5.35*10^3$ for χ is .95. While the precision obtained by algorithms as BFSe and SMe does not depend on E', QW and QW-SM increase the precision obtained when the effort is increased until saturation of the precision is gotten. Furthermore, it is important to consider that for χ=.95 there are only 4,006 relevant answers in total available for the 19,463 questions; in case of χ=.45 there are 105,514 relevant answers available in total.

5. Conclusions and Future Work

Although Travers and Milgram [34] showed that it is feasible to reach a node using unicast query-routing algorithms, there is high probability that a message may be dropped before the message reaches its destination. Banerjee and Basu [3] developed a probability model that takes into account the probability that a message is answered using unicast query routing, considering that each node has enough expertise to answer the question but also some probability of dropping the question. Several P2P systems use multicast query routing, in which a question is sent to many contacts instead of sending it to one acquaintance at a time. Although the probability of obtaining an answer is increased with multicast query routing, there are more nodes requested, and as a consequence, system scalability can be threatened.

Several state-of-the-art techniques have been proposed to reduce the number of messages generated by multicast query-routing algorithms. For example, the usage of time to live (TTL) limits the number of hops for a given question. Recently, in [38], we proposed Asknext, an agent protocol for social search that uses stop messages. Using stop messages allows flooding to be stopped at the same number of hops at which the answer that satisfies the question is found. With these two techniques (TTL and stop messages), the best answer that is present somewhere in the network is not found due to being out of the distance or time range set in the search process.

In this paper, we presented Question Waves (QW). QW generates several attempts (waves) to find the information needed. Initially, agents send the question to their acquaintances with the highest probability of providing a good answer, i.e., a subset of their contact list. If, after some time, there is no satisfying answer, then the agents send the request to the next best contacts on the list. With QW, the answer relevance is correlated with the answer arrival order. This feature is key for stopping the search process with a lower probability of losing answers that are more relevant than the received ones. Furthermore, the process can save user's time because users can read the answers as they come, and the first answers are the most likely to be the relevant ones. Finally, the number of messages generated by QW is fewer than those generated by other methods, such as Asknext, yet even better relevance is achieved.

With QW inspired from physical waves, we used the ideas of *effort* to answer a question, effort distribution among acquaintances, effort that is dissipated when it is propagated to other agents due to friction, different speeds to send question messages to several acquaintances, and constructive and destructive interferences, which modify the effort that an agent uses to search for answers.

These ideas can be implemented with continuous or piecewise functions. We implemented and used them in two types of simulations. The first type of simulation consisted of using randomly generated knowledge bases (KE dataset) in which all of the agents were able to answer all of the questions but with different relevance due to their expertise, as a type of social question answering. In this set of simulations, we used 2 social networks with 100 agents, 495 and 395 links, .521 and .386 clustering coefficients, and .0736 and .0568 assortativity coefficients, respectively. The objective of this type of simulation was to show that it is possible to obtain answers ranked by their arrival time by using only local information (as in a P2P environment). We obtained as a best correlation .953, which is a high value for using local information; this value is clearly higher than the correlation of answer quality with the reputation of the mediator or the transitive reputation, which were the highest reputations of .133 and .776, respectively. A second objective was to determine the answer relevance when the search process was stopped. When a search is stopped with fewer answers, the correlation with time is lower, but the mean answer quality is higher. We consider the trust model we used in this experiment as the main reason for this decrease of correlation coefficients; there is not enough discrimination between answerers of high expertise and the usage of random values to compute the answer relevance, which may affect when they have similar expertise values. This set of simulations can be seen as the application of QW in a social answering system. In summary, the highest correlation is achieved by using QW with long answering sets, although even in small answering sets, the correlation is high enough to consider that the first answers are the most relevant.

The second type of simulation was in the context of recommender systems and consisted of using data from movielens, a highly reputable and widely used movie-recommender dataset. The contribution of this type of simulation is a step forward with real data in the experimentation with the aim of showing that the method can be applicable in a real domain: recommender systems can be seen as a kind of social search system, concretely a social feedback system [10]. The movielens dataset does not include a social network; thus, with the aim to simulate QW, we created 2 social networks of 387 agents out of the movielens dataset, with 3,089 and 1,933 links each one and the following features of every network: .53 and .515 clustering coefficients and .0401 and .0277 assortativity coefficients, respectively. The objective of these simulations was to show that the QW method can generate a smaller number of messages than other algorithms applied in P2P social networks, and the answer quality is even higher. Because we are using data from a recommender system, we understand that the answers of higher quality are those with lower prediction errors (MAE and MSE). The results demonstrate that the system uses fewer messages than Asknext, and the answer quality is higher. We also compared the correlation of the answer quality with the arrival time, and it was comparable with the correlation of the answer quality with answerer similarity (approximately .6), on which the agents' behaviour is based. This set of simulations can be considered an appropriate application of QW in a social feedback system.

Finally, in Subsection 4.3 we studied the recall and the precision in a window of the 3 first answers per question, in function of the initial effort. The simulation results show that to achieve a target recall, QW needs less effort than BFSe (BFS with usage of effort) and SMe (using SM with effort); this behaviour is accentuated for more strict classifications of relevant items. We modelled the relation of recall and number of messages as a saturated growth model, obtaining that QW-SM needs approximately 3 times less messages to obtain the same recall than BFSe and approximately 2 times less messages than SMe (for χ =.75 or higher with the movielens dataset). About precision, considering the first three answers per question, in movielens dataset, considering QW and QW-SM reach 3.5 times higher precision than BFSe and SMe; and at the initial values ($E'=10^3$) the precision is at least 2 times higher with QW.

In future efforts, we think that it would be interesting to study the effects of small-world network topology characteristics in the scalability and correlation (or correctness) of answers because these factors can have a considerable impact and may affect the design of functions. Extracting which network features contribute to higher scalability, speediness of relevant answers or higher correlation with answer arrival and answer relevance can help to create local methods to modify the network dynamically to obtain these objectives. We also believe that allowing agents to modify their contact list would improve the correlations because the agents will be able to correct some shortcomings of the arrival order of the answers. The trust model is the base of QW, and studying the effect of different trust models can be interesting. A better understanding of the effects of trust models may help to implement an adaptable delay of the question time. Because the agents are autonomous, each one can have different implementations of the functions of the model and can adapt them based on interactions, and an interesting future work would be to study the behaviour of agents with heterogeneous behaviours. Also, studying question-message propagation in an epidemic model could be interesting. Additionally, once the benefits of the algorithm have been shown, it is needed to use a more complete and real dataset that include natural language processing, a dataset like that can be extracted from Quora, Twitter, etc. Finally, it would be interesting compare QW results with traditional search systems as search engines and Q&A portals.

6. Acknowledgements

This research is funded by the EU project Num. 238887 (iSAC6+), the IPT-430000-2010-13 project (SAKE), the EU DURAFILE num. 605356, FP7-SME-2013, BSG-SME (Research for SMEs) Innovative Digital Preservation using Social Search in Agent Environments, TIN2010-17903 *Comparative approaches to the implementation of intelligent agents in digital preservation from a perspective of the automation of social networks*, the Universitat de Girona research grant BR09/10 awarded to Albert Trias, and the AGAUR grant for the CSI-ref.2009SGR-1202.

References

[1] E. Agichtein and Y. Liu, Modeling Information-Seeker Satisfaction in Community Question Answering. ACM Transactions on Knowledge Discovery from Data 3(2) (2009). doi:10.1145/1514888.1514893

[2] H.J. Ahn, A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem, Information Sciences 178 (2008) 37-51. doi:10.1016/j.ins.2007.07.024

[3] A. Banerjee, S. Basu, A Social Query Model for Descentralized Search, In: Proc. 2nd ACM Workshop on Social Network Mining and Analysis, SNAKDD-08, Las Vegas, 2008.

[4] R. Baraglia, M. Mordacchini, P. Dazzi, L. Ricci, A P2P REcommender System based on Gossip Overlays (PREGO), In: Proc. 10th Int. Conf. Computer and Information Technology, CIT, 2010, pp. 83-90. doi:10.1109/CIT.2010.502

[5] M.K. Bergman, White Paper: The deep web: surfacing hidden value, The journal of electronic publishing 7(1) 2001 doi:10.3998/3336451.0007.104

[6] J. Bian, Y. Liu, E. Agichtein, H. Zha, Finding the right facts in the crowd: factoid question answering over social media, In: Proc. Int. Conf World Wide Web, WWW'08, New York, 2008, pp. 467-476.

[7] J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, In: Proc. 14th Conf. Uncertainty in Artificial Intelligence, Madison, WI, 1998, pp. 43-52.

[8] S. Buchegger, A. Datta, A case for P2P infrastructure for social networks - opportunities & challenges, In: Proc. 6th Int. Conf. Wireless On-Demand Network Systems and Services, Utah, 2009, pp. 161–168.

[9] E. Bulut, J. Wang, B.K Szymanski. Cost Effective Multi-Period Spraying for Routing in Delay Tolerant Networks. IEEE/ACM Transactions on Networking, 18(5):1530-43. 2010

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

[11] J. Golbeck, Generating predictive movie recommendations from trust in social networks, In: K. Stølen, W.H. Winsboroughs, F. Martinelli, F. Massacci (Eds.), Proc. of the 4th Int. Conf. on Trust Management, iTrust 2006, LNCS 3986, 2006, pp. 93–104.

[12] M. Gomez, J. Sabater-Mir, J. Carbo, G. Muller, Improving the ART-Testbed, thoughts and reflections, In: Proc. Workshop on Competitive agents in: Agent Reputation and Trust Testbed, Salamanca, 2008, pp. 1-15.

[13] L. Guo. E. Tan, S. Chen, X. Zhang, Y.E. Zhao, Analyzing patterns of user content generation in online social networks, In: Proc. 15th ACM Int. Conf. Knowledge Discovery and Data Mining, KDD'09, 2009, pp. 369-378.

[14] M. Hackworth. Waves- Transverse and Longitudinal Waves, http://www.physics.isu.edu/~hackmart/waves100.PDF

[15] F.M. Harper, D. Moy, J.A. Konstan, Facts or friends?: distinguishing informational and conversational questions in social Q&A sites, in: D.R. Olsen. Jr., R.B. Arthur, K. Hinckley, M.R. Morris, S.E. Hudson, S. Greenberg (Eds.), Proc. 27th Int. Conf. Human Factors in computing systems, CHI'09, New York, 2009, pp. 759–768.

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

[17] D. Horowitz, S.D. Kamvar, The anatomy of a large-scale social search engine, In: Proc. 19th Int. Conf. World Wide Web, New York, 2010, pp. 431–440.

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

[19] P. Jurczyk, E. Agichtein, Discovering authorities in question answer communities by using link analysis, In: Proc. 16th ACM Conf Information and Knowledge Management, CIKM'07, New York, 2007, pp. 919-922. doi:10.1145/1321440.1321575

[20] J.A. Konstan, N. Kapoor, S.M. McNee, J.T. Butler, Techlens: Exploring the use of recommenders to support users of digital libraries. Task Force Meeting, Phoenix, AZ. 2005. Available at <http://www.grouplens.org/papers/pdf/CNI-TechLens-Final.pdf>.

[21] H. Kwak, C. Lee, H. Park, S. Moon. What is Twitter a social network or a new media? In: Proc. 19th International Conference on World Wide Web. pp. 591-600. 2010 Doi:10.1145/1772690.1772751

[22] Q. Liu, E. Agichtein, Modeling Answering Behavior in Collaborative Question Answering Systems, In: P. Clough et al. (Eds.), Proc. 33rd European Conf. Advances in Information Retrieval, ECIR 2011, LNCS 6611, 2011, pp. 67-79.

[23] X. Liu, W.B. Croft, M. Koll, Finding experts in community-based question-answering services, In: Proc. 14th ACM Int. Conf. Information and Knowledge Management, CIKM'05, New York, 2005, pp. 315-316. doi:10.1145/1099554.1099644

[24] M. Luck, P. McBurney, O. Shehory, S. Willmott, Agent technology: computing as interaction roadmap for agent-based computing. A roadmap for agent based computing, AgentLink 2005, ISBN 085432 845 9.
[25] E. Michlmayr, A. Pany, G. Kappel, Using Taxonomies for Content-based Routing with Ants, Journal of Computer Networks 51 (2007) 4514-4528.
[26] M. Montaner, Collaborative Recommender Agents. State-of-the-art and New Techniques. Lambert Academic Publishing. Saarbrücken, 2011. ISBN 978-3-8383-5426-2.
[27] A. Moreno, J. Ll. de la Rosa, B.K. Szymanski, J.M. Bárcenas. Reward System for Completing FAQ, Frontiers in Artificial Intelligence and Applications – AI Research & Development ISSN 0922-6389, Vol: 202, pp: 361-370, Nov 2009, IOS Press
[28] B.A. Nardi, S. Whittaker, H. Schwarz, It's not what you know, it's who you know: Work in the information age, First Monday 5 (5) (2000).
[29] G. Pitsilis, L. Marshall, A Trust-enabled P2P Recommender System, In: Proc. 15th Int. Workshop Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE '06, 2006, pp. 59-64, doi: 10.1109/WETICE.2006.12
[30] P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, J. Riedl. GroupLens: An open architecture for collaborative filtering of netnews, In: Proc. Conf. Computer Supported Cooperative Work, CSCW'94, 1994, pp.175-186.
[31] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Item-based Collaborative Filtering Recommendation Algorithms, In: Proc. 10th Int. Conf. World Wide Web, WWW'01, 2001, pp. 286-295.
[32] U. Shardanand, P. Maes. Social information filtering: Algorithms for automating 'word of mouth', In: Proc. Conf. Human Factors in Computing Systems. CHI'95, 1995, pp. 210-127.
[33] B. Smyth, P. Briggs, M. Coyle, M.P. O'Mahony, Google shared a case-study in social search, In: G.J Houben et al. (Eds.), Proc. 17th Int. Conf. User Modeling, Adaptation, and Personalization, UMAP'09, LNCS 5535, 2009, pp. 283-294.
[34] J. Travers, S. Milgram, An Experimental Study of the Small World Problem. Sociometry 32 (4) (1969) 425-443.
[35] A. Trias, J. L. de la Rosa, B. Galitsky, G. Drobocsi, Automation of social networks with QA agents (extended abstract), In: van der Hoek, Kaminka, Lespérance, Luck and Sen (Eds.), Proc. 9th Int. Conf. Autonomous Agents and Multi-Agent Systems, AAMAS'10, Toronto, 2010, pp. 1437-1438.
[36] A. Trias, J.L. de la Rosa, Propagation of Question Waves by Means of Trust in a Social Network, In: H. Christiansen et al. (Eds.): Proc. 9th Int. Conf. Flexible Question Answering Systems, FQAS'11, 2011, LNAI 7022, 2011, pp. 186–197.
[37] A. Trias, Relevance and Speed of Answers: How Can MAS Answering Systems Deal With That?, Online Proc. 13th European Agent Systems Summer School, EASSS 2011, Girona, 2011, <http://eia.udg.edu/easss2011/resources/docs/paper9.pdf>.
[38] A. Trias i Mansilla, J.L. de la Rosa i Esteva, Asknext: An Agent Protocol for Social Search, Information Sciences 190 (2012) 144–161.
[39] T. Vu, A. Baid. Ask, Don't Search: A Social Help Engine for Online Social Network Mobile Users. In: Proc. 35th IEEE Sarnoff Symposium 2012.
[40] F.E. Walter, S. Battiston, F. Schweitzer, A model of a trust-based recommendation system on a social network, Autonomous Agents and Multi-Agent Systems 16 (1) (2008) 57–74.
[41] L.S. Wu, R. Akavipat, A. Maguitman, F. Menczer, Adaptive peer to peer social networks for distributed content based web search, in: Social Information Retrieval Systems: Emergent Technologies and Applications for Searching the Web Effectively, IGI Global, 2007, pp. 155-178.
[42] B. Yu, M.P. Singh, An agent-based approach to knowledge management, In: C. Nicholas, D. Grossman, K. Kalpakis, S. Qureshi, H. van Dissel, L. Seligman (Eds.), Proc. 11th Int. Conf. Information and Knowledge Management, CIKM'02, New York, 2002, pp. 642–644.
[43] B. Yu, M.P. Singh, Searching social networks, In: Proc. 2nd Int. joint Conf. Autonomous Agents and Multi-Agent Systems, AAMAS'03, Melbourne, 2003, pp. 65–72.
[44] B. Yu, M.P Singh, K. Sycara, Developing Trust in Large-Scale Peer-to-Peer Systems, In: Proc. 1st IEEE Symposium Multi-Agent Security and Survivability, 2004, pp. 1-10.
[45] Y. Zhou, G. Cong, B. Cui, C.S. Jensen, J. Yao: Routing questions to the right users in online communities, In: Proc. IEEE Int. Conf. Data Engineering, ICDE, 2009, pp. 700-711. doi:10.1109/ICDE.2009.44

Table A.
Summary of QW terms and concepts.

Concept	Term
Probability of having satisfied question q_j of questioner a_i at a distance d	$P_{i,j}(d)$
Probability of satisfying question q_j of questioner a_i exactly at distance d	$\delta P_{i,j}(d)$
Probability of having satisfied question q_j of questioner a_i at a time t	$P_{i,j}(t)$
Probability of satisfying question q_j of questioner a_i exactly at time t	$\delta P_{i,j}(t)$
Tree of degree α	α
Probability of all agent to satisfy a question with own knowledge (at Subsection 3.1)	β
Expertise	k
Implication	I
Other Factors	R
Weights of k, I and R respectively	w_k,w_I,w_R
Effort	E
Initial Effort	E_0
the effort that a_i can transmit	E_i
Effort that a_i transmitted to a_j	$E_{i,j}$
Value that determines that an agent contribution for a question has been enough	σ
Modificator of "agent sensibility"	φ
Answer Quality	ϑ
Constant that indicates the maximum lost of energy due friction	μ
Probability of having an own question	Ψ
Interest Vector	$It=\{i_1, i_2, i_3, i_4, i_5\}$
Set of Agents	$A=\{a_0, a_1, \ldots, a_i\}$
Answer Quality mean	$\bar{\vartheta}$
Number of answers	a
Number of questions	q
Number of messages	m
Answers per question	a/q
Threshold that determines answers above it are considered relevant	χ
Regression model constant (applied as a message constant and as an effort constant)	τ
similarity of users t and v	$u(t, v)$
rating given by user x to item y	$r_{x,y}$
mean of the ratings of user x	$\overline{r_x}$
set of items	Im
Collaborative filtering constant	K
Question delay from agents a_i to a_j	$t_{i,j}$
Trust agent a_i has to agent a_j	$T_{i,j}$
Sum of squared Error	S

7. Main Results and Discussion

In chapter 3, we analysed the village and library paradigm; we compared them and dissected the village paradigm. The dissection of the village paradigm, which includes social search, allowed us to observe that there are three important aspects that can be automated. These aspects are the *propensity* to answer, the generation of *answers*, and the selection of *candidates* to ask the question. We also detected other aspects that can be automated in the future, such as *identify the information need*, *answer evaluation*, *learning*, and *rewarding*. In the following sections, we review the literature to study the current automation level of those aspects.

In the case of *answers*, there are several studies that reuse the previous content; the first step is to determine the similarity or equivalence between the questions [Jeon 2005]. Then, there are the options of reusing the previous answers [Jeon 2005, de la Rosa 2010, Wenyin 2010] or suggesting similar questions to the users [Cao 2008]. There are other systems that use Frequently Asked Questions (FAQs) or that attempt to extract an answer from a repository [Hirschman 2001, Simmons 1965.]

In the case of *candidates*, we found 8 aspects to select them: *Expertise*, *similar tastes*, *demographic profile*, *trust*, *sociability*, *activity*, *authority,* and *answerer interests*.

Another important aspect of the *candidates* is the *query-routing*, especially when the question can be sent directly to acquaintances only. Most of the centralised environments allow us to ask to any user, while the environments based on the P2P architecture allow only requests to the acquaintances of the social network, in which case it is possible to answer with a referral or to forward the question to an acquaintance. In this thesis, we focused on the P2P mechanisms for forwarding questions, in unstructured P2P social networks, given that social networks are bottom-up and that structured architectures use methods to organise nodes and use distributed hash tables (DHT) for content search.

With respect to *propensity*, we have seen 6 characteristics to filter (to show or not) one question to a user. These characteristics are: *Expertise*, *Interest*, *Who requests*, *Effort*, *Reward*, and *Answerer Context*.

In chapter 4, we proposed the Asknext protocol, which uses 3 types of messages: the question, answer and stop messages. The idea behind the protocol is the following:

We assume an unstructured P2P social network in which each user u_j is represented by a multi-agent system a_j; in this thesis, we assume that each of these multi-agent systems is composed of exactly one agent; as a result, we will refer to these multi-agent systems as *agent*; the user u_j is the owner of agent a_j. Each agent a_j has a knowledge base that has been provided by the owner u_j. These assumptions are not minor; in practise, only a few users provide their knowledge to their agents. However, we consider that currently, with the advent of social networks (and web 2.0), there is intense communication among users in the form of questions and answers; comments on blogs; cross comments in micro-blogging that generate conversation threads; exchange of documents (usually pictures); and comments on content that can be considered as tags that the semantic web has required for many years. In the same way that users communicate with each other through online social networks, agents can perform the same communication; an agent can communicate with its owner and with the agents owned by the acquaintances of its owner, using the same question-answering channels and the same micro-blogging threads. In the state of the art of FAQ and QA, there are studies that explain who to extract frequent questions from [Liljenback 2007] (which can represent 80% of the total questions), to answer them automatically without loss of quality or loss of answer relevance. In this thesis, I did not adopt any specific FAQ or QA technique; instead, we used techniques from recommender systems in which the interactions are questions and the answers are in the form of evaluations (ratings).

Thus, the starting point is that an agent has a contact list that comprises the list of agents owned by acquaintances of its owner. Every agent can generate a knowledge base using the social interactions of its owner and can use its knowledge base to answer questions that are asked to its owner.

In the process of answering a question, each agent can play one of the three following roles:

- *Questioner*: In the Questioner role, the agent has the objective of satisfying the information needs of its owner.

- *Answerer*: In the Answerer role, the agent answers using its own knowledge or the knowledge of its owner.
- *Mediator*: In the Mediator role, the agent that was delegated to answer a question (thus, it is not a questioner) delegates answering it to another set of agents.

In short, the protocol works as follows:

- When a user u_j has an information need, she asks her agent a_j (the agent will then play the role of *questioner*); the agent will check its knowledge base and will attempt to answer the question (according to a study, 30% of the times, a user is searching for information that was found previously [Smyth 2009]). If it is not possible to answer the question by using its knowledge base, then the agent will send the question to a subset of its acquaintances (according to the same study, 70% of the time that a user is searching for information, it was found by their acquaintances previously).
- When an agent a_j receives a question that was sent by another agent, it will first attempt to answer it by using its knowledge base; if it is possible, then the agent will play the role of answerer and will send the set of answers to the requester agent; otherwise, the agent can show the question to its owner (as occurs in MARS [Yu 2002]) and, in the case that the owner answers, will forward the answer to the requester (again taking the role of answerer). A last chance is forwarding the question to a subset of its acquaintances and becoming a mediator.
- When a *questioner* or a *mediator* receives an answer, it must consider whether the answer will satisfy the information need; are the answers available enough to contribute to satisfying the information need? In that case, it will send a stop message to all of the agents that it delegated the task to.
- When an agent receives a stop message, it will not forward the remaining question messages that are associated with the stop message and will forward the stop message to the agents that have been delegated by it.

Another important aspect is allowing that the stop messages be able to stop the propagation of the question messages. We modelled the stop message process as a kinematics problem in which the relative speed of the stop messages and question messages is positive; in other words, the stop messages are faster than the question messages. This relative speed is implemented by slowing the question messages, adding a delay in their propagation from agent to agent. We set the delay of the question messages to be a function of the distance from the questioner, stopping the question at the same distance that the answer was found. With this model, it is also possible to study the time needed to obtain an answer on the basis of the delays and the distance at which the answer is found. In the studied cases, we found that stop messages work well in general.

BFS and Asknext use the closest answers that are available for satisfying the information need; the algorithms do not consider that better answers can be found at a distance that is even farther beyond the propagation limits set up by the algorithm. Better relevance can be achieved with the QW algorithm, as we will see in chapter 6.

Another technique proposed in the protocol is the CQT (Check the Question Thread), which comprises checking before forwarding a question to the previous receivers; in this way, a question will not be sent to an agent that is known that received it previously.

The protocol has been formalised using the framework of Hussain and Toni [Hussain 2008].

In our simulations, we used data from a Recommender Systems dataset from Movielens. It has been necessary to generate social networks out of that dataset; the networks have been randomly generated by adding a number of acquaintances from a fixed interval to each agent. In the simulations, we have seen that the protocol reduces the number of messages (approximately 20 times with the dataset and networks used). Fig. 1 shows the number of messages generated for social networks 1 and 2, when we used stop messages (C1-S, C2-S) and when stop messages are not used (C1, C2), as a function of Δ, which is directly proportional to the combination of the number of answers and how relevant the answers should be to stop a search. The time that is required to satisfy a question need is slightly increased (in network 2, which was less dense, the mean was 2.5 simulation steps, and with stop messages, the mean is 3.2, which is approximately 25% slower). We also found that the protocol does not affect the relevance.

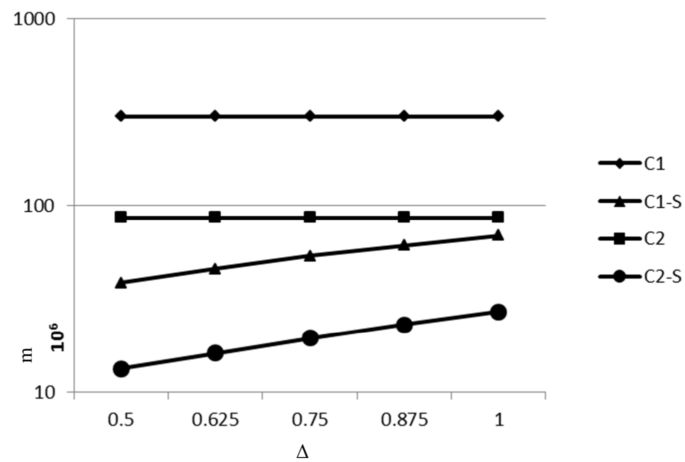


Fig. 1. Number of messages with Asknext and BFS for 2 social networks, applying different Δ values. (From [Trias 2012: figure 12c])

There are a few exceptions when using SM does not reduce the number of messages. For example, once most of the agents have received the question, sending stop messages will not reduce the total number of messages. We are focusing on reducing the number of messages to measure the scalability, but it will be more realistic to associate a cost for processing each message (and their associated tasks) because each type of message will have a different processing cost: stop messages would be faster to process; it is not studied, yet.

With Asknext, the answers follow the inverse path of the questions. We described the following reasons for using this path instead of answering back directly: *delegation*, *traceability*, *privacy*, *social behaviour*, *comprehension*, *automatic learning*, *reduction of answers received*, and *direct dialog*. However, we expect to justify that decision experimentally in our future work.

In chapter 4, we did not study the effect of social network topologies in the Asknext protocol; however, we addressed it in chapter 5, where we detailed the properties of 21 social networks that we generated with the aim of confirming that the conclusions from chapter 4 are valid regardless of the social network topologies and, at the same time, can study the effect of the network properties in the Asknext protocol. Social networks are complex networks that have a high *clustering coefficient* (c), a relatively short *average path length* ($\bar{d}$), and most of them are assortative, although most networks are disassortative. A network is assortative when the assortative coefficient (r) is positive. In this chapter, we decided to use networks with c≥0.1, based on the properties of the social networks studied by Newman and Park [Newman 2003b], Mislove et al. [Mislove 2007] and Newman [Newman 2001]. We grouped the 21 social networks that were generated into 4 groups, based on the number of links. We implemented combinations of the mechanisms SM, CQT and TTL over 6 algorithms, and we evaluated the *average time needed to obtain an answer* ($\overline{T_a}$), the *mean of equal question messages that each agent receives* ($\overline{qma}$), and the *number of messages sent* (*m*). We observed that, when there is a high variation in the number of links (L), L and the *largest eigenvalue* (λ) explain very well the $\overline{T_a}$, $\overline{qma}$ and *m*. When there is low variation of L (in each group), other properties have a larger effect, and L and λ are worse predictors. The algorithm BFS with a TTL of 7 (Alg6) is very sensitive to variations in L.

With these 21 networks (and the Movielens dataset that was used), the number of messages reduced by Asknext is between 94.6% and 95.5%, which indicates that there are between 18.5 and 22.22 times fewer messages, which proves the conclusions obtained in chapter 4. With respect to the reduction in the messages, we observed that between 94.5% and 95.3% are due to SM, while the CQT reduces the number of messages by only an additional 2.55% to 4.52%. The impact of CQT reducing the number of messages is very low considering the impact of SM.

The analysis of the network topology properties over these inputs can provide further information if, instead of using normalised values between 0 and 1, we use other values for the number of messages that are more representative with respect to the output values that are not normalised. For example, if we consider 2 totally connected networks of 5 and 1,000 nodes, both will have c=1, and c will not show a difference in the number of messages. If, instead of c, we used the number of closed triplets (that is used to compute c), then perhaps it would be possible to explain better the outputs. This issue can be studied in future work.

The most important contributions of this thesis are from chapter 6 onward, where we proposed the algorithm Question Waves (QW), which, in its core form, comprises adding delays to the question messages in such a way that the delays are inversely proportional to the trust in the acquaintances; thus, with QW, **the first nodes requested are the nodes that are more trusted**. As a result, we used Spearman's correlation to check that the answers came sorted by relevance. Each question message corresponds to an attempt to satisfy the information need, and the first attempts have a higher probability of satisfying the information need, which we relate to the trust in the candidates. If the information need is satisfied, then the ensuing attempts are cancelled.

QW uses other mechanisms, such as delaying the answers, with the aim of agents that were delegated with a question could answer before the agent that had delegated the question answering to them; thus, there is a mechanism for the question effort, which can be distributed heterogeneously to different acquaintances, and not all of the effort sent is used to satisfy the information need (friction). This algorithm is compatible with SM (QW-SM).

Effort mechanisms and their distribution allow adjusting the propagation of the question.

In the simulations, we used assortative social networks with low density, positive assortativity, and a high clustering factor. The knowledge base has been modelled as a knowledge exchange scenario (KE dataset) that was generated randomly and as an opinion exchange scenario, using the Movielens dataset that was used previously. With the KE dataset, we obtained correlations of the answer relevance with the arrival order of .953, which we consider to be a very high value. We observed that, using SM, the correlation is reduced, but the mean of the answer relevance is increased; this finding indicates that the implementation proposed has difficulty in differentiating the relevance of the best answers, which could be due to the effect of the random component on the generation of the answer relevance and because the delays are not sufficiently different from each other when the trust is high. We performed experiments with the Movielens dataset to perform in-depth analysis of the behaviour of QW.

With the Movielens dataset, we evaluated the effect of using delays in question messages as a function of trust and delaying the answers based on the self-confidence; we observed that the relevance increases and the number of messages is reduced.

We also studied the relevance as a function of the effort applied and the number of messages sent; specifically, we used as a relevance measure the recall, which indicates the proportion of relevant answers from the algorithm that were retrieved in response to the questioners. The number of messages generated and the recall can be modelled as a saturated growth model. With the aim of using a model that is easier to explain, we used equation 1.

$$y(t) = A\left(1 - e^{-\frac{t}{\tau}}\right) \quad \rightarrow \quad r(m) = 1 - e^{-m/\tau} \tag{1}$$

where:

- $r(m)$ corresponds to $y(t)$ and indicates recall as a function of the number of messages sent (m).
- τ explains the model and indicates that, with $m = \tau$, the recall obtained is $1 - e^{-1} \cong 63.2\%$; when $m = 2\tau$ the recall is $1 - e^{-2} \cong 86.5\%$; and so on, in a form that is multiples of τ.
- If we make a ratio of the τ of the 2 models (τ_a/τ_b), we obtain how many times the number of sent messages of model *b* are needed for model *a* to obtain the same recall.

We explored other saturated growth models that provide fewer errors in modelling the relation between the recall and the number of messages, such as the model in equation 2 (where *a* and *b* are the constants of the model); however, it was difficult to compare the results and thus extract conclusions from them.

$$r(m) = \frac{am}{b+m} \tag{2}$$

We observed that QW provides the highest recall given the same effort and a higher recall given the same number of messages, as can be shown in Fig. 2 and Fig. 3. When QW is used, the higher the χ value (that means we are selecting only the answers of the highest relevance), the faster the recall converges. For example, with χ=0.75, the τ for the recall and number of messages is $109.26 \cdot 10^6$ messages for BFSe (BFS with effort) and $33.56 \cdot 10^6$ messages for QW-SM; this finding represents that BFSe needs approximately 3.26 times more messages to obtain the same recall in all of the algorithms.

Next, we used *precision* to measure the relevance, given that, in real environments, computing the recall is not practical. To measure the precision, we considered the first three answers per question, and we evaluated the proportion of relevant answers out of them. With respect to BFSe and SMe, the precision obtained is constant as a function of the effort; in the case of QW and QW-SM, they have the behaviour of a saturated growth model. With χ=0.45, at the initial values of effort, QW and QW-SM obtain a 2.5 times higher precision than BFSe and SMe, and obtain 3.75 times higher precision for the highest efforts, as can be seen in Fig. 4.

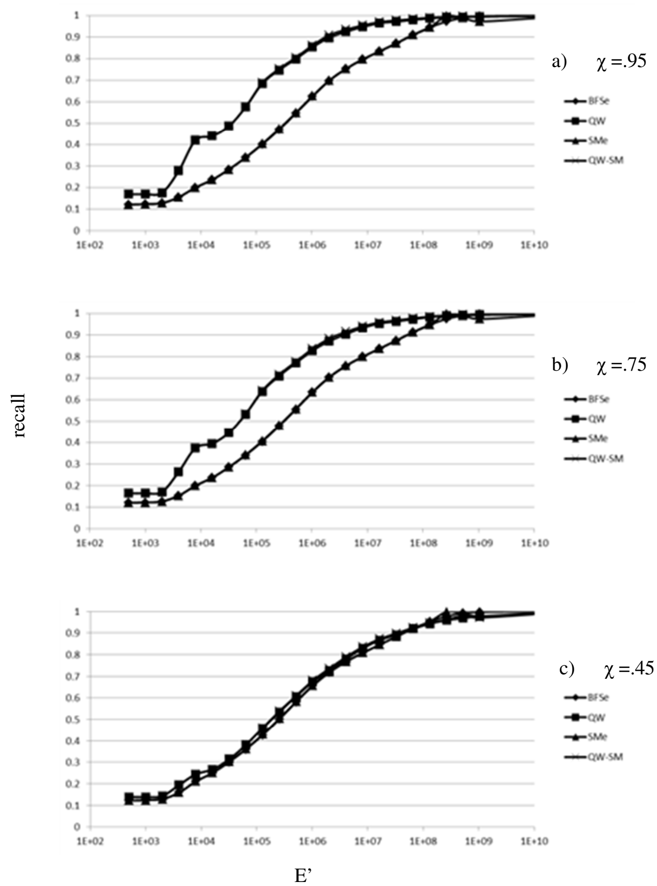


Fig. 2. Recall as a function of the Effort applied for BFSe, QW, SMe, and QW-SM, for different χ values. From [Trias2013c: figure 12d, e and f]

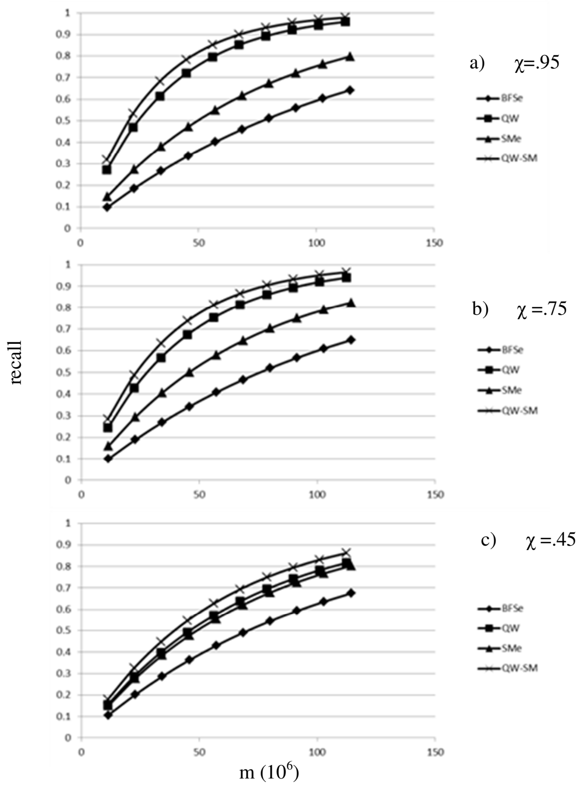


Fig. 3. Modelled recall as a function of the number of messages for BFSe, QW, SMe, and QW-SM. From [Trias2013c: figure 13d, e and f]

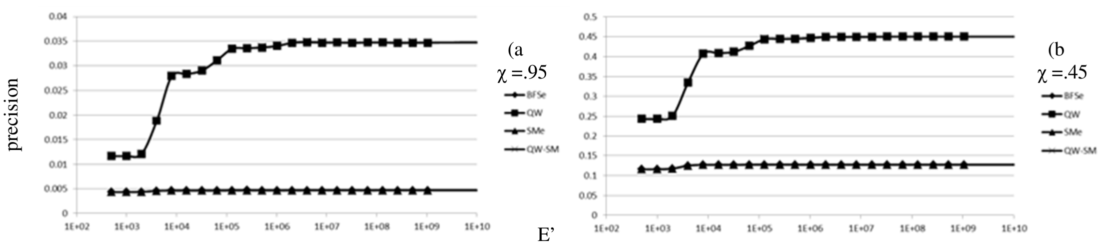


Fig. 4. Precision as a function of Effort for BFSe, QW, SMe, and QW-SM for different χ values. From [Trias2013c:figure 14c and d]

One of the assumptions of this thesis is that all of the agents can answer immediately (thus, if no delay is applied, then the answer message will be received in the following simulation step); however, this scenario will not occur in a more realistic environment. There is a need to study the effects of agent properties (for example, if some agents have more resources or can answer faster) for future work. It is advisable as well to study in depth the time that is required to generate a new answer; for example, an answer entered manually by a user might be more relevant that another answer that was generated automatically (or a reused answer), but the second answer will be provided faster. Consider the following example: in numerical analysis, a computation with further iterations will obtain more precision, but additionally, more expert users could use methods with higher convergence and obtain a higher precision in less time. These aspects must be studied in depth in several applications domains.

We did not apply CQT to QW because it reduced the number of messages by only less than 5%, and we considered that it did not have significance.

It is advisable to consider whether the delay can be annoying for users. Should the users wait additional time to receive an answer if a system uses QW instead of BFS? In the case of QW, we did not study the time needed to satisfy an information need because the time depends on the delay configurations and the effort per question. However, in the case of Asknext, the results indicate that the answers were 25% slower. I think that it is worthwhile to wait 25% more time because the users (and their agents) will receive 95% fewer questions. Furthermore, when QW is applied, answers with high relevance will be provided faster than with Asknext, which is better and more preferable than having fast but less relevant answers. However, it is a drawback that, if there are no available answers that have a high relevance, then the waiting time will increase. This aspect must be studied as future work, which would account for the time that the agents need to answer automatically and the time that is needed to process and send messages and the frequency at which the users ask questions.

8. Main Conclusions

1. The village paradigm can be dissected with respect to the following three main aspects: the answers, the candidates, and the propensity.
   1.1 Additionally, the detection of the information need, the rewarding, learning and answer evaluations can be considered.
   1.2 The candidates include query-routing methods.
   1.3 To any degree, all of these aspects can be automated.
2. Adding delays in the propagation of the questions and answers in the query-routing methods allows improving the system scalability and relevance.
   2.1 Delays in question messages allow using stop messages to avoid propagation of question messages once the information need is satisfied.
   2.2 Delays in question messages as a function of trust (inversely proportional) contribute to improve system relevance. The delays enable the system to obtain the information first from candidates with the highest trust.
   2.3 Delays in answer messages contribute to improving the relevance because answers from further agents can be provided faster than less relevant answers that are closer and share the same path.
3. Adding delays in the propagation of questions and using stop messages allow the number of messages generated to be less sensitive to the number of links in the social network compared with the algorithm BFS.
4. Stop messages are more efficient when they are used in networks that have higher clustering coefficients, as is the case of social networks.
5. The reduction in the number of messages provided by the CQT mechanism is very small compared with the reduction in the number of messages provided by stop messages.
6. In some cases, stop messages can generate more messages, which could occur when the question propagation reaches most of the agents, and sending stop messages will produce more messages compared with allowing the question to continue the propagation.
7. When the effort is applied to multicast query-routing algorithms, the number of messages generated and the recall has the behaviour of a saturated growth model.
8. The QW algorithm, with the usage of delays as a function of trust in the propagation of question messages, allows us to relax the decision of how many acquaintances

must be requested. It allows us to obtain the benefits of unicast models, such as when the information is found fast and thus only uses a few messages. However, additionally it has a higher probability of obtaining a relevant answer because it is less sensible to dropping, and more agents are requested if it is needed.

9. Personal Opinion

I think that, with this thesis, I have performed a very productive work: the proposed mechanisms of social search not only reduce considerably the number of messages but also significantly increase the relevance of the received answers. Furthermore, the answers arrive ranked by order of relevance, in real time; we have no evidence that these goals have been achieved before. The answers arriving ranked by relevance enable users to start reading the answers on the fly without being concerned with whether better answers could be provided later on. This advantage must compensate for the extra time taken by my algorithms to perform the question answering. It is true that the waiting time has not been studied in my thesis, and it deserves further study in the future. However, I think that this idle (extra) time will have no impact on the users because it will most likely represent only some seconds or a few minutes in the worst cases. We should not forget that, with web search engines, we obtain results immediately; however, then the users proceed to a second search (refinement) by seeking information inside the proposed documents, and this second search takes time and is not automated.

After all of the work that has been conducted, there are still some challenges pending. The mechanisms proposed have not been proven in a real environment: it would be necessary to study their behaviour in a real case and allow that agents interact with humans who usually have dynamic behaviour. There are a number of improvements about agent behaviours; in this thesis, we considered that all of the agents had the same behaviour, and we did not consider divergences. Finally, with respect to network topologies, we checked that there is a small effect on the improvements obtained with Asknext, yet further studies should be performed as future work.

This work is a first step toward the automation of the village paradigm by means of intelligent agents. Inside our Agents Research Laboratory (ARLab), there have been studies that were indirectly related to the automation of the village paradigm with agents, which were focused on recommender systems only. The future work proposed in the next chapter indicates some of the future challenges that I would like to address after this PhD.

10. Erratum: Minor Errors Detected and Clarifications

10.1. Erratum: Minor Errors Detected

In chapter 4:

- Fig. 11 caption should be "Number of messages using stop messages…" instead of "Number of answer messages using stop messages".
- Fig. 12-a is MSE, not RMSE, and Fig. 12-b is MAE, not MSE.
- On pg. 56 (numbered as 156), it should be 3.57×10^6 instead of 3.57×10^7.

10.2. Clarifications

In chapter 4:

- On pg. 51 (numbered as 151), 1st paragraph. "In Eq. (6), is it feasible to use a dynamic and adaptive T_F in function of the current distance and T_R can be obtained" should be "With Eq. (6), a dynamic and adaptive T_F in function of the current distance and T_R can be obtained".
- On pg. 52 (numbered as 152), 3rd paragraph. "query(X,Y,Z): The agent X asks Z of the set of agents Y. It should be "query(X,Y,Z): The agent X asks the query Z to a set of agents Y.
- On pg. 52 (numbered as 152), 7th paragraph, 4th bullet point. "DialogStatus(Z,S): Indicates whether S is the agent status for dialogue Z", and other occurrences of dialogue. Dialogue is frequently associated with conflicts, but in this chapter (and the complete thesis) it is about knowledge transfer dialogues.
- On pg. 59 (numbered as 159), 7th paragraph. "agents must have enough with one answer" should be replaced by "agents must be satisfied with one answer".

In chapter 6:

- On pg. 87. 3rd paragraph, 2nd bullet point. "$P_{i,j}(d+1) = P_{i,j}(d) \ or \ \partial P_{i,j}(d+1) = P_{i,j}(d) + \left(1 - P_{i,j}(d)\right) * \partial P_{i,j}(d+1)$", "$or$" is the or operator, it should be "V".

11. Future Work

We organised future work in the following main blocs: agent behaviours, applicability, question wave improvements, and environment properties.

In the category of *Agent Behaviours*, we account for aspects such as agent decisions and their associated algorithms and their resources.

In the category of *Applicability,* we include all of the steps that are needed for a real implementation of these methods, including the usage of real questions and answers and studying the effect of the delays in real time.

In the category of *Question Wave Improvements,* we include formalisation aspects and engineering aspects on how to configure a QW environment to obtain some answer distributions over time.

In the category of *Environment,* we include aspects of social network topologies and how the answers are distributed in the social network.

11.1 Agent Behaviours

The agent behaviours used in this thesis are quite simplistic because our objective was more focused on studying the effects of the mechanisms proposed at a macro-level. Currently, agent technology has the potential to improve the results that we obtained, for example, agents can manage their resources or can have different objectives. Additionally, more types of agents could be included. The following list contains all of the aspects that we consider as future work:

- AB-01 - ALGORITHM TO ADAPT THE WAVES DELAY: The implementation of some of the behaviours of question waves has been arbitrary. One clear example is the function of delaying the question propagation based on trust or the delay of one's own answers based on self-confidence. The aim of our work was to show the concept; we think that there is not the same sub-optimal solution that might work in all of the environments and contexts. An adaptive algorithm could be proposed to provide sub-optimal solutions in several contexts. Such an algorithm would adapt the waiting times as a function of trust based on the local results. Each agent should decide its behaviour based on its objectives and states.

- AB-02 - MAS: Multi-Agent System. The architecture of MAS' can be more complex. Each user can own a MAS rather than only an agent. Agents vertically specialise in answering special questions, agents mine opinions and provide them as answers, and agents can be without an owner or even without a "human owner". Other examples would be having one agent at a job site, which could stay there when the worker leaves the enterprise. For example, if one human dies, then the agent can continue to live. If different agents are available, then virtual organisations can be considered.
- AB-03 -RECONFIGURE NETWORK: Furthermore, agents can automatically do/propose some changes in the network topology as adding or removing links (in other words, adding and removing contacts or acquaintances in the contact list), with the aim of obtaining better results; it can be studied as a local decision or as a global decision with a meta-recommender. Several preliminary studies move in this direction: in our research laboratory, we are studying the effects of adding acquaintances in the contact list, with the aim that agents that are closer to the nodes usually provide the best answers.
- AB-04 -REWARDING: With the aim of providing incentives to agents to fulfil some of the tasks (finding answers to a question), it could be interesting to add some reward mechanisms. This approach could help agents to prioritise some of the tasks when they have a limited bandwidth, and perhaps the rewards should be based on the effort mechanism presented with Question Waves. However, this method needs more specification as to what is achievable with the received effort units and what they represent.
- AB-05 -EFFORT DISTRIBUTION OPTIMISATION: The important points are the following: deciding to send the question to fewer agents; effort efficiency; spread the effort in a more efficient way; agents will not attempt to spend (delegate) more effort than is necessary; and adapting the weights of the amount of effort to attempt to obtain a sub-optimal distribution.
- AB-06 -LEARNING: Agents can learn the answers that they receive when they play the role of mediators. Currently, we are studying the effect of applying

short-term memory in the agents, implementing it as a cache memory, which would allow the agents to reuse the answers that are received, thus increasing the answer speed and reducing the number of messages, with a small loss of relevance.

- AB-07 -ANSWER THREAD: It would be interesting to study the effect of using the inverse path of the question for answering instead of answering directly to the questioner. In chapter 4, we explained some of the benefits that we considered from using the inverse path of the question; however, performing some experiments can show whether there is any difference in the number of messages, the speed of answers and the relevance.
- AB-08 -BANDWIDTH: Once we can pre-process real data, we can start to evaluate the real costs of the actions; the costs can be in terms of the CPU time, network usage, or other considerations. This approach will allow us to specify the real time that agents need to perform their actions. Furthermore, it can help to determine how many actions the agents can perform in the simulations, based on their CPU and network availability. These properties can be different for several agents because they can be situated in computers that have different features, and additionally, they can run different algorithms to attempt to find an answer. Knowing the real cost of the actions can help in determining the effectiveness of the stop messages because the number of messages may not be the most important consideration compared to the cost of the associated actions.

11.2 Applicability

Although I consider that the results of this thesis are a good starting point, simulations are far from being applicable, and further efforts must be made. The most important efforts are to address real data (social networks, questions and answers); additionally, the environment in a more realistic scenario will be dynamic, while in our simulations, it is static. The following list contains all of the aspects that we consider to be future work from the perspective of applicability.

- AP-01 -FACEBOOK EXPERIMENTATION. First experiments of social search applicability in a social network real environment can be done in Facebook, studying how friends can help with knowledge.

- AP-02 -LABORATORY Stage 3 (REAL SOCIAL NETWORK). After working with 2 datasets (KE and Movielens), in the laboratory Stages 1 and 2, instead of creating the networks, it is time to work with samples of social networks. These networks can be extracted from online social networks, such as Twitter, where there are available datasets.
- AP-03 -LABORATORY Stage 4 (REAL Q&A): An ultimate step will be to use real questions and answers. Although for the current experimentation, the usage of recommender systems data has proven to be sufficient, the challenge is to use a system that can address real questions and not be limited to predicting the evaluation that a user will rate an item. In this sense, we will need to use Natural Language Processing (NLP). NLP can work well in a domain-specific scenario, but open domains can threat its precision. NLP state-of-the-art studies can be used with that purpose, but we would need to add this step as future work because adding such a complex tool could provoke poor results, which might be attributed to the algorithm not working well. With the idea of building a laboratory, these data can be pre-processed before the simulations, generating an agents' knowledge model for the simulations.
- AP-04 -ANSWER DELAY: In our framework, agents can answer directly or request an answer of their owner. We assume that the answering times are all equal for all of the agents in our simulations. The time needed to answer should be different for each case: we think that human answers should, in general, be better than the automated answers, as well as that they would need more time than agent-generated answers. The heterogeneous answering time should be studied because it might have a high impact on the Question Waves algorithm.
- AP-05 -EFECTS OF DYNAMIC NETWORK: Social networks are dynamic. Studies of the flexibility of the agents when there are changes in this environment could be performed. The impact of these changes must be studied, such as the aspect of how much time is needed for the environment to stabilise. The main difference from AB-03 is that, in AB-03, agents attempt to improve the social network, and the dynamic aspects of the social networks in AP-04 is about the changes made by the users, for example, when there is a new user or when an agent is removed from the network, or the users decide to add or remove a contact without considering the effects on the search process.

- AP-06-ROBUSTNESS: Studies of the robustness of the algorithm should be performed. First, we can consider the option that agents drop questions (one main reason to drop questions would be that the agents maximise their gains dropping them). Second, we can consider that the messages are lost for reasons other than failures in the network, for example agents drop messages. Three, we can consider that agents are not available at some time, or that the people we consider in the simulations provide new answers, with a higher answering time.
- AP-07 -STUDY THE APLICABILITY OF QW IN CORPORATE SOCIAL NETWORKS: In corporate social networks, Q&A are more frequent, and this environment can be considered as a scenario for partially automating the village paradigm.

11.3 Question Waves Improvements

The Question Waves algorithm can be improved, or adapted, based on the system's needs. Possibly some environments will require faster answers, while others will need higher scalability or more relevance; how to configure the QW environment should be studied with this aim. Furthermore, QW should be formalised more precisely. In the following list, we detail the future work that we consider in this line:

- QWI-01 - CHARACTERISATION: In section 4.3 of chapter 6, we explained that there is an exponential behaviour in the relation between the number of messages and the recall; it can be seen more clearly in the middle of the function. It would be interesting to find models that best fit and explain more the behaviour of the algorithm. In this case, we performed some preliminary tests, which are not yet ready for publication.
- QWI-02 - MODELING: Observing the output of Fig. 8 in chapter 6, and having some constraints fixed, we can think about the desired output distribution, and on that basis, provide an algorithm that specifies a question waves configuration that provides a sub-optimal solution.
- QWI-03 - RECALL VS TIME: In chapter 6, section 4.3, we studied the recall based on the number of messages; it would also be interesting to study the growth that it has as a function of time, with several effort configurations.

- QWI-04 - FORMALISATION AND GENERALISATION: We presented the algorithm of Question Waves; however, a more formal approach is needed. Furthermore, although we focused on the approach of query routing in a P2P agent social network, perhaps it would be possible to generalise and find more applications for the algorithm and to present it as a meta-heuristic algorithm. “Reasoning about actions” should be considered to formalise QW.

11.4 Environment properties

Properties of the environment have a clear impact on the algorithm. A detailed study of their effects is needed to know in which cases QW can be used and when it can provide its best results. As a continuation, there is a list of the future work that we consider about environment properties, specifically in the social network topology and knowledge distribution:

- EP-01 -ANSWER DISTRIBUTION: The way that the knowledge is distributed, such as how many answers are available, has an important impact in the algorithms, which deserves to be studied and quantified in some way. Different answer densities, their relevance and their location could be studied.
- EP-02 -NETWORK: In chapter 5, we studied the effects of the network topology on the techniques of TTL, CQT, and SM. Obtaining only one answer was sufficient: the next step was to study the effect when the number of answers needed is increased. Furthermore, the availability of answers is crucial on the outputs. Comparing several scenarios with the same networks and different answer availabilities would increase the value of these experiments. In the case of Question Waves, it is required to study the effects that network topologies have in Question Waves and to attempt to detect which types of networks boost the Question Waves performance, if any.
- EP-03 -EPIDEMIC MODEL: Epidemic models can be used to study the effects of network topologies on the dissemination of Information [Ganesh 2005]. In computer networks, epidemic models are used to study how cascading failures (as viruses) are spread over a network. Using epidemic models, it is possible to study how the process evolves and the mechanisms for reducing the impact of the

evolving process (the question dissemination in our case) through immunisation [Calle 2010]. An epidemic model can be used to study the process of question propagation and can be used to study mechanisms of immunisation as stop messages.

12. References

[Buchegger 2009] S. Buchegger, A. Datta, A case for P2P infrastructure for social networks – opportunities & challenges. In: Proc. 6th Int. Conf. Wireless On-Demand Network Systems and Services, Utah, 2009, pp. 161-168.

[Calle 2010] E. Calle, J. Ripoll, J. Segovia, P. Vilà, M. Manzano. IEEE Network Magazine. 24(6)(2010): 17-22.

[Cao 2008] Y. Cao, H. Duan, C.Y. Lin, Y. Yu and H.W. Hon. Recommending questions using the mdl-based tree cut model. In: WWW, 2008, pp. 81-90.

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

[de la Rosa 2010] J.L. de la Rosa, A. Trias, R. Ruusalepp, et al. Agents for Social Search in Long-Term Digital Preservation. In: Sixth International Conference on Semantics Knowledge and Grid (SKG), 2010, pp. 363-366 ISBN:978-1-4244-8125-5 DOI: 10.1109/SKG.2010.60

[Galitsky 2013] B. Galitsky. Exhaustive simulation of consecutive mental states of human agents. Knowledge-Based Systems 43 (2013) 1-20. DOI:10.1016/j.knosys.2012.11.001

[Ganesh 2005] A. Ganesh, L. Massoulie, D. Towsley. The Effect of Network Topology on the Spread of Epidemics. In: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communication Societies. 2005.

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

[Hendler 2010] J. Hendler, T. Berners-Lee. From the Semantic Web to social Machines: A research challenge for AI on the World Wide Web. Artificial Intelligence 174 (2010) 156-161.

[Hirschman 2001] L. Hirschman and R. Gaizauskas. Natural language question answering: the view from here. Natural Language Engineering 2001 7(4):275-300. DOI:10.1017/S1351324901002807

[Horowitz 2010] D. Horowitz, S.D. Kamvar. The anatomy of a large-scale social search engine. In: Proc. 19th Int. Conf. World Wide Web, New York, 2010, pp. 431-440.

[Hussain 2007] A. Hussain, F. Toni, Bilateral agent negotiation with information seeking. In: Proc. of 5th European Workshop on Multi-Agent Systems, 2007.

[Jeon 2005] J. Jeon, W.B. Croft and J.H. Lee. Finding similar questions in large question and answer archives. In: CIKM, 2005, pp. 84-90.

[Liljenback 2007] M. E. Liljenback. Contextqa: Experiments in interactive restricted domain question answering. Master's thesis. San Diego University, 2007.

[Mislove 2007] A. Mislove, M. Marcon, K.P. Gummandi, P. Druschel, B. Bhattacharjee. Measurement and Analysis of Online Social Networks. Proceedings of the 7th ACM SIGCOMM conference on Internet Measurement. 2007. DOI:10.1145/1298306.1298311

[Montaner 2011] M. Montaner. Collaborative Recommender Agents. State-of-the-art and New Techniques. Lambert Academic Publishing. Saarbrücken, 2011. ISBN 978-3-8383-5426-2.

[Newman 2001] M.E.J. Newman. The structure of scientific collaboration networks. In: Proc. national academy of sciences of the United States of America (PNAS). 2001. Vol. 98 no.2. pp. 404-409. DOI:10.1073/pnas.98.2.404

[Newman 2003a] M.E.J. Newman. Mixing patterns in networks. Physical Review E 2003; 67. DOI:10.1103/PhysRevE.67.026126

[Newman 2003b] M.E.J. Newman, J. Park. Why social networks are different from other types of networks. Physical Review E 2003; 68, DOI:10.1103/PhysRevE.68.036122

[Simmons 1965] R. F. Simmons. Answering English questions by computer: A survey. Communications Association for Computing Machinery (ACM) 1965 8(1): 53-77.

[Smyth 2009] B. Smyth, P. Briggs, M. Coyle, M.P. O'Mahony, Google shared a case-study in social search, in: G.J. Houben et al. (Eds.), Proceedings of 17th International Conference on User Modeling, Adaptation, and Personalization, UMAP'09, LNCS 5535, 2009, pp. 283–294.

[Trias 2011a] A. Trias, J.L. de la Rosa, Propagation of Question Waves by Means of Trust in a Social Network, In: H. Christiansen et al. (Eds.): Proc. 9th Int. Conf. Flexible Question Answering Systems, FQAS'11, 2011, LNAI 7022, 2011, pp. 186–197.

[Trias 2011b] A. Trias. Relevance and Speed of Answers: How can MAS Answering Systems Deal With That? 13th European Agent Systems Summer School. 2011. http://eia.udg.edu/easss2011/resources/docs/paper9.pdf

[Trias 2012] A. Trias i Mansilla, J.L. de la Rosa i Esteva, Asknext: An agent protocol for social search, Information Sciences 190 (2012) 144-161.

[Trias 2013a] A. Trias i Mansilla, J.L. de la Rosa i Esteva. Survey of social search from the perspectives of the village paradigm and online social networks. Submitted to Journal of Information Sciences.

[Trias 2013b] A. Trias i Mansilla, E. Calle Ortega, J.L. de la Rosa i Esteva, M. Manzano Castro. Studies on Social Networks Topologies and Their Effect in the Dissemination of Information in P2P Social Search. Submitted to International Journal of Communication Systems.

[Trias 2013c] A. Trias i Mansilla, J.L de la Rosa i Esteva, Question Waves: an algorithm that combines answer relevance with speediness in social search. Submitted to Information Sciences (In second review).

[Trias] A. Trias i Mansilla, S. Sethserey, J.L. de la Rosa I Esteva. Applying Short-term Memory to Social Search Agents.Working Paper.

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

[Wenyin 2010] L. Wenyin, T. Hao, W. Chen and M. Feng. A Web-Based Platform for User-Interactive Question-Answering. World Wide Web 2010 12(2): 107-124. DOI: 10.1007/s11280-008-0051-3

[Wooldridge 2002] M. Wooldridge. An Introduction to MultiAgent Systems. John Wiley & Sons, Inc., New York, USA. 2002.

[Yu 2002] B. Yu and M.P. Singh, An agent-based approach to knowledge management, In: CIKM ’02: Proc. 11th international conference on Information and knowledge management. New York: ACM, 2002, pp. 642–644

13. Annex: Update of the Asknext protocol for the deployment of Question Waves

In chapter 3, we presented the Asknext protocol and its formalisation. To use QW with the protocol, some updates would be needed:

- Question delays must be based on trust.
- The method send[answer(Y,S,Z)] must include an answer delay; this method would be send[answer(Y,S,Z), T_A], where T_A is the time instant at which the message is sent.
- When a question is answered, which happens in section 4.5, the question does not always have to be finalised. Then, it would be required to check the status of the question when considering the effort requested and the answers received.
- A method is needed for effort distribution during question propagation.

Adding the following predicates would be required:

- *EffortDistribution*(Effort,E',C'): Indicates the effort E' that will be requested to the acquaintance C' given an available Effort.
- *AnswerDelay*(Z,T_A): Indicates that, for an answer Z, a delay T_A must be applied.
- *QuestionDelay*(Z, T_F,C'): Indicates that question Z to acquaintance C' has a delay of T_F.
- *AnswerSatisfyQuestion*(Z): Indicates whether, with the answer Z (and answers received previously), the associated question is satisfied.

The updated formalisation of the protocol is represented in Fig. 5, Fig. 6, Fig. 7, Fig. 8 and Fig. 9.

A= send[query(X,Y,Z),Time, Effort]
P= {Y $\subseteq$ C , Z.QT ∩ Y = Ø}
Q= {Dialog(Z)}
X={Dialog(Z),DialogStatus(Z,Asked),
MyRecipients(Z,Y)}

Fig. 5. Send question message.

Here, in all of the cases, sending a question includes the fields Time and Effort (Fig. 5).

A= process[query(X,Y,Z), Effort]
P= {C $\in$ X }
Q={Dialog(Z), KnowAnswer(Z), Z.Deadline$\leq$CurrentTime}
X={Dialog(Z),Sender(X,Z), EffortDistribution(Effort,E',C'), QuestionDelay(T_F,C'),send[query(Y,C',Z), CurrentTime+T_F,E'], DialogStatus(Z,Asked)}

A= process[query(X,Y,Z),Effort]
P={C $\in$ X, KnowAnswer(Z) }
Q={Dialog(Z), Z.Deadline $\leq$CurrentTime}
X={Dialog(Z), Sender(X,Z), Answer(Z,A), AnswerDelay(Z,T_A),send[answer(Y,X,A, CurrentTime+T_A),] EffortDistribution(Effort,E'), QuestionDelay(T_F,C'), send[query(Y,C',Z), CurrentTime+T_F,E'], DialogStatus(Z,Asked) }

Fig. 6. Receiving a question message.

Having an own answer does not finalise the dialog, and a delay T_A is applied to the known answers. Furthermore, effort is distributed according to predicate EffortDistribution, and the question delay depends on the predicate QuestionDelay.

A= process[stop(X,Y,Z)]
P= {C $\in$ X }
Q={Dialog(Z), Z.Deadline$\leq$CurrentTime}
X={Dialog(Z), DialogStatus(Z,Finished)}

A= process[stop(X,Y,Z)]
P= {C $\in$ X , Dialog(Z), Sender(Z,X)}
Q={DialogStatus(Z,Finished), Z.Deadline$\leq$CurrentTime}
X={MyRecipients(Z,C'), send[stop(Y,C',Z)]}

Fig. 7. Receiving a stop message.

Fig. 7 contains the formalisation of receiving a stop message, which does not require any update.

A= process[answer(X,Y,Z)]
P= {C $\in$ X , Dialog(Z), AnswerSatisfyQuestion(Z)}
Q= {DialogStatus(Z, Finished), Z.Deadline<=CurrentTime}
X={MyRecipients(Z,R),send[stop(Y,R,Z)], Sender(Z,S), send[answer(Y,S,Z),0])}

A= process[answer(X,Y,Z)]
P= {C $\in$ X , Dialog(Z), !AnswerSatisfyQuestion(Z)}
Q= {DialogStatus(Z, Finished), Z.Deadline<=CurrentTime}
X={MyRecipients(Z,R), Sender(Z,S), send[answer(Y,S,Z),0])}

Fig. 8. Receiving answer message.

When an answer is processed (Fig. 8), the only difference is that stop messages only will be sent when it is considered that the question is satisfied.

A= send[answer(X,Y,Z), Time]
P= {Sender(Y,Z)}
Q= {DialogStatus(Z, Finished)}
X= {DialogStatus(Z,S)}

A= send[stop(X,Y,Z)]
P= {MyRecipients(C',Z), C' $\subseteq$ Y}
Q= {DialogStatus(Z, Finished)}
X= {DialogStatus(Z, Finished)}

Fig. 9. Sending an answer message and a stop message.

Fig. 9 contains the update of the actions of sending an answer and sending a stop message. The only modifications are that the status of dialog Z when an answer is sent can be Finished or Asked and that only the answer will be sent after Time.