

Asynchronous Social Search: Towards better Internet and Enterprise Search

Marco Buijs

11-05-2014

Contents

1	Introduction	3
2	Research approach, methods and techniques	7
2.1	Structured literature review design	8
2.2	Competitor analysis design	9
2.3	Prototype design	9
2.4	Experiment design	10
2.5	Case study design	12
3	Related work	17
3.1	Information Retrieval	18
3.2	Social Semantic Web	19
4	Competitor analysis	22
4.1	Nine Search methods	22
4.2	Comparison of the nine search methods	25
5	Model development process	35
5.1	Competitive advantages of a new search method	35
5.2	Getting people involved	36
5.3	Critical success factors	38
5.4	Requirements specification of a prototype	41
5.5	Hard and Software requirements	46
6	Experiment: quality of the proposed search method	48
6.1	Ranking quality	48
6.2	Discovery capabilities	52
7	Case study: feasibility in a business environment	53
8	Conclusions and discussion	63
	References	64
A	Interview questions	68
B	Interview answers	69
C	Paper submitted to KDIR conference 2014	78
D	Paper submitted to Palgrave Macmillan KMRP	79

Abstract

A fundamentally different approach to internet and enterprise search is described in this thesis. The approach is based on asynchronous Social Search, a search method in which people collaborate to find the information they are looking for. A Structured Literature Review was performed to get an overview of the most relevant literature related to asynchronous Social Search. Different competitors and implementations of search methods in practice were evaluated and compared in a competitor analysis. Critical Success Factors of asynchronous Social Search were identified and a prototype was built. The prototype was tested using a global experiment for its feasibility for web search. Also, a case study was performed to test the feasibility in a business environment. We conclude that asynchronous Social Search has huge potential, mainly for Enterprise Search as a Single Point of Access to organisational information. Both because the implementation requires no integration with the existing Information Technology infrastructure of organisations and participants were very satisfied with the results provided by the prototype.

1 Introduction

For the past decades Google has been the number one Search Engine (SE) in the world and left their competition far behind. The basic way Google finds webpages on the World Wide Web is by using crawlers. Such a crawler scans every page on the Web it can find using the links that are provided on the webpages that the crawler already knows about. Ranking of the pages regarding certain search terms is based on the content of the page and the amount of incoming and outgoing references of the page (Brin & Page,1998). Although crawlers are very efficient at finding web pages, they cannot find everything that is available on the web. They are dependent on the links they come across during the crawling process and files specifically created for crawlers, often named robots.txt, that present lists of web pages. For good reasons, crawlers are prohibited access from many resources such as your e-mail, calendar and other personal information. To access such personal information authentication is required. Another problem with crawlers is that they often lack the power to derive meaning and concepts from a website (Gupta, Li, Yin & Han,2010). Furthermore, webpages are more and more often generated dynamically, making it harder to find all available webpages. What also is happening at the moment is that more and more information is becoming available via Application Programming Interfaces (APIs). An API is a protocol that can be used to communicate with and access data from other software components (Hinchcliffe,2008). Crawlers are not capable of determining the meaning of queries on those APIs and can therefore not properly index API requests. The last problem we address here is that information is sometimes not available on one particular webpage, but segmented over multiple pages. As a consequence of these problems, expressive queries cannot be answered correctly by the most modern search engines of our times. Examples of such expressive queries that are hard to answer for search engines are:

- My e-mail,
- My colleagues,
- Flight details of my flight to Amsterdam,
- Give me the 10 capitals in Europe that have the most inhabitants,
- Give me a list of all the cities with more than 2 million inhabitants,
- Give me the shortest public transport connection from my current location to Amsterdam and I want to leave now,
- List of amateur soccer teams in Madrid.

Figure 1 shows a result provided by Google on the query "List of amateur soccer teams in Madrid". Notice that the result on the query can differ from person to person because Google makes use of personalization of search results and

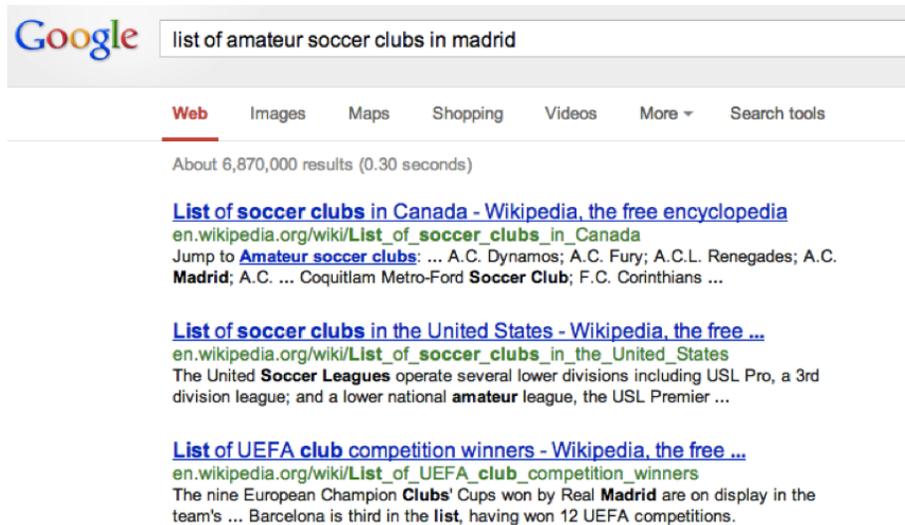


Figure 1: A result on the query "List of amateur soccer teams in Madrid" provided by Google.

other techniques to determine which results should be returned to the user. One particular movement that tries to provide answers to such queries is the Semantic Web. The Semantic Web aims at adding logic to the World Wide Web. The idea behind this is that the Web becomes better readable for machines. This way, machines would be able to get a better understanding of how pages are related to each other and where they have to look for certain information. Furthermore, the Semantic Web enables machines to aggregate data from different pages and present this aggregated data to users in a clear overview (Berners-Lee, Hendler, Lassila et al.,2001).

During this study it was investigated whether a good search engine can also be based on user-generated content instead of crawlers and references. The proposed Search method is an asynchronous Social Search method. Based on work from Evans and Chi (B. M. Evans & Chi,2008) and Golovchinsky, Pickens and Back (Golovchinsky, Pickens & Back,2009), we define asynchronous Social Search as

”Information seeking supported by a network of people where collaboration takes place in a nonconcurrent way”.

Important concepts in asynchronous Social Search are user-generated content and user feedback. The main question in this research is whether people would be better at generating proper search results than computers. Although computers are better at performing a specific task such as playing chess, more sophisticated tasks such as speaking, listening and understanding are still performed better by humans (Russell,2003). In this study we elaborate on using user-generated content and feedback from users as an alternative to crawlers and backlinks. In particular, the proposed system is based on collaborative tagging, which is the concept where you allow anyone to freely attach tags to content. A tag is a keyword assigned to a piece of information to describe that piece of information, i.e., a tag is metadata (Gupta et al.,2010). In our case the piece of information is a web resource. The main research question in this study is:

How does asynchronous Social Search perform compared to existing search methods?

To answer this question the following sub questions need to be answered.

1. What are the critical success factors of Asynchronous Social Search?
2. What functionality should an asynchronous Social Search engine have?
3. How to design and build an asynchronous Social Search engine?
 - (a) What hard- and software is needed to operate an asynchronous Social Search engine?
 - (b) How to get users involved in generating search results?
 - (c) How to prevent malicious manipulation of search results by users?
4. How to measure search engine quality?

The deliverables of this study are twofold. First, this study will result in a Master thesis answering the research questions and an analysis of the quality of the developed search engine based on the comparison with other search engines. The second deliverable is a functional Asynchronous Social Search engine.

Performing research should be meaningful and relevant with respect to both science and society. From a scientific point of view it is interesting to know how far Artificial Intelligence has evolved. As mentioned before, humans perform some tasks better than machines and vice versa. It would be useful to know

which approach performs better at providing search results to users. This would also give an indication of the complexity of search engines in general. Furthermore, for fields such as Knowledge Management and Data Mining it is relevant to investigate whether there are more efficient and better quality methods for searching information in huge amounts of data. By comparing the efficiency and quality of search results provided by computer-generated content-based search engines and asynchronous Social Search engines, it becomes clear what works better and therefore people can use the best search method. If this happens to be computer based search, this would not have a big impact on society, because this is already the standard way of searching although people might not be aware of it in that sense. When it would be found that asynchronous Social Search or a combination of asynchronous Social Search with computer based search provides better search results, this would mean that digital search could be made more efficient for everyone around the world, providing people with better search results than is the case today.

2 Research approach, methods and techniques

To structure the research project Design Science and Behavioural Science was used. Design Science is a model that can be used to create innovative artefacts that serve humanity, or more specifically, organisations and people. Behavioural Science seeks to develop and verify theories regarding behaviour of people and organisations (Hevner, March, Park & Ram,2004). The Design Science research method suits the project well since the goal of the project is to investigate whether a different approach to retrieval of information can lead to better search results in a search engine. To be able to validate this different approach an artefact was created. Behavioural Science is focused on the development and justification of theories, whereas Design Science is focused on building and evaluating artefacts. Where Behavioural Science has truth as goal, Design Science has utility as goal. As with most research projects, this research project needs both utility and truth. The artefact that was built could be used to validate the main research question using Behavioural Science. So in short, used Design Science to develop an Asynchronous Social Search engine. Then, we used Behavioural Science to validate the usefulness of Asynchronous Social Search. This research project was structured according to the framework as proposed by Hevner, March and Park as shown in Figure 2. This framework addresses both Behavioural Science and Design Science.

The research project started with an explorative literature study to gain knowledge of the field. Success factors of search engines were defined and mapped to Asynchronous Social Search engines. A competitor analysis was performed to investigate the state of the art techniques that are currently in use. Also, a requirements analysis was performed to derive the desired functionality for the Asynchronous Social Search engine. During the requirements analysis phase, prototypes of the system were made to validate decisions regarding design and functionality of the search engine. A small community of users was created that could provide constant feedback on the system. Furthermore, a method was found to measure quality of search engines. The prototype was tested with an experiment and a case study. The following Subsections elaborate on the design of the structured literature review, competitor analysis, prototype, experiment and case study.

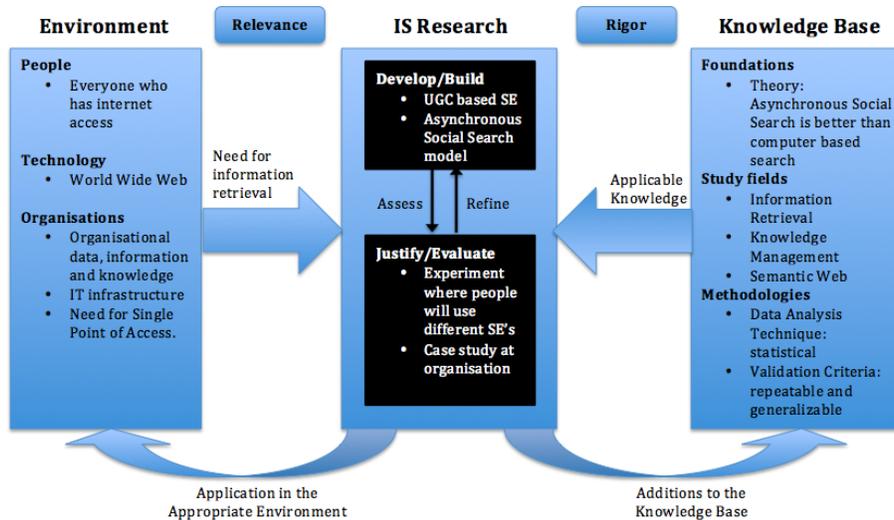


Figure 2: Information Systems Research Framework, adapted from Hevner, March and Park (Hevner et al.,2004).

2.1 Structured literature review design

To find relevant information with respect to asynchronous Social Search a structured literature review was performed. Next to the main and sub research questions, the following questions were formulated:

1. What are the existing methods used in web search?
2. How do these existing methods perform compared to asynchronous Social Search?
3. What are the strength and weaknesses of the different search methods?
4. What implications will asynchronous Social Search have?

To be able to reproduce the results of this review, a protocol was followed while searching for relevant literature. Automated search via Google Scholar was used since all articles that could be relevant, and deliver a significant attribution to science, should be available through there. Notice that the standard settings of Google Scholar were used during the search process, with the exception that citations were excluded from the search. Exclusion criteria were applied to reduce the number of results. First of all, the result must have been cited at least ten times a year, meaning that an article written in 2005 should have been cited at least 80 times to be included in the research. As the base year, 2013 was taken and articles written in 2013 and 2014 were always included. Papers written in 2012 needed at least ten references to pass the selection criteria. Second, articles

that are not about search, or do not have any analogy with search were also excluded from the research. Third, articles that haven't been peer reviewed were also excluded. Last, threads to construct validity, internal validity and external validity of the results was assessed. No restrictions to the date of publishing were used to include or exclude resources. Note that also duplicate results were removed during the process. Because the retrieved resources differed in many ways, such as how they were designed and which methods and outcome measures were used, narrative synthesis was used to analyse. In total, nine competitors were compared. They were not only selected based on market share, but also based on whether they are making use of unique concepts. This is, because it does not make sense to compare multiple Search Engines that are based on the same principles.

2.2 Competitor analysis design

A Search Engine that makes use of the asynchronous Social Search method can only add value to society if it can outperform its competitors. Therefore, a competitor analysis was performed to identify and compare competing search methods that are actually in use in different Search Engines. We made use of competitor arrays to analyse, evaluate and compare the potentials of the competitors (Gordon,1989). We also looked at the competitive advantages that can be achieved by making use of the asynchronous Social Search method. Although usually weightings are assigned to the different factors that are evaluated in a competitor array, we did not do this because we did not find any scientific proof to distinguish relative importance of different factors in search. Competitors and relevant factors to compare were identified based on literature, common sense reasoning, testing the methods in practice and information provided by the owners of the Search Engines.

2.3 Prototype design

During the design and build face it was very likely that small changes to the requirements of the system would be made. The design and build face was structured using the Scrum method. Scrum was chosen because it is an agile development method that has proven itself in the recent past (Schwaber & Beedle,2002). The reason for using an agile development method is that it suits the project on the following aspects:

- Number of developers: Agile development methods are suitable for small development teams. In the case of this project the amount of developers will not exceed eight.
- Requirement changes: Agile development methods provide space for change of requirements during the development face. A potential danger of using agile development methods is the concept called feature creep. This term refers to the problem of adding too many new features to the require-

ments of the product (Keith,2010). This can lead to delays and missing deadlines.

- **Criticality of the application:** The system that was built is not a mission critical system such as an air traffic control system or bank transaction system. Agile development methods are particularly suitable for projects of low criticality.

There are quite some agile development methods available to choose from. Advantages of using Scrum over other agile development methods are the use of a backlog for keeping track on progress, use of short iterations (sprints) and daily measured progress (Schwaber & Beedle,2002).

To model the requirements the Unified Modeling Language was used. In particular, Use Case Diagrams and Component Diagrams were created. Use Case Diagrams are used to model the interaction that people should be able to have with a system, whereas Component Diagrams are used to show the software architecture and the flow of messages within that architecture, also showing the interfaces that are available in the architecture (D'souza & Wills,1998).

2.4 Experiment design

Methods that measure search engine performance found in the literature study were listed and compared. From the available methods a measurement and comparison method for search engines was derived. This method was then used to compare the performance of existing search methods with the Asynchronous Social Search method. Most important in information retrieval systems is to present the most relevant results to the user regarding certain queries that are posed by the user. There are two candidate methods described in literature to evaluate search method quality. The first method comes from the field of information retrieval and is concerned with measures such as precision and recall (Manning, Raghavan & Schütze,2008). According to Manning, Raghavan and Schütze information retrieval is

”finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)” (Manning et al.,2008).

The method requires a predefined test set of documents with relevance judgements regarding certain queries. Furthermore, a test suite of information needs is required. A document is called relevant if it addresses the information need, so not when it just happens to contain all the words in the query. This quality measurement method is most often used on one or more standard test data collections to be able to compare results with other search methods. Notice that this method assumes that the search method is able to retrieve any possible resource. Quality of search methods however, does not only depend on the quality of the ranking method but also on its retrieval capabilities. Retrieval capability refers to the set of resources that can be found by the search method.

A system based purely on PageRank for example, will not be able to find a webpage that has no incoming links referring to it. Although this method is very suitable for the evaluation of both ranked and unranked retrieval results, it is less suitable for the evaluation of a complete search method since retrieval capabilities are not taken into account. A more feasible method to compare search methods is described by Joachim (Joachims,2002). The method is based on interweaving results from two search methods and click through data. When a user types in a query, the top K of two search methods you want to compare are interweaved and presented to the user. Based on the assumption that results on which the user clicks are more relevant than results where the user does not click on and the assumption that the user scans the search results from top to bottom, one can determine which search method performs better using statistical research methods. An additional advantage of this setup is that you do not need any relevance judgements anymore, which is often a time consuming process to acquire, especially when dealing with search methods in which the scope is virtually unrestricted. The proposed setup leads to a blind test in which the clicks of users on links indicate the relative preference of the user in an unbiased way. There is one difference in measurement in our experiment with respect to the experiment as described by Joachim. Joachim recorded for every query which search method performed better. When method A received more clicks for that query than method B, A was considered to be the winner for that query. It was also possible that a user clicked on zero links or clicked equally often on links from method A and B. The latter two cases were both considered to be a tie. In our experiment we did not keep track of individual queries. Instead, we kept track of clicks. We recorded clicks via search method A and clicks via search method B. In this situation, ties are handled as follows: when both search methods received 0 clicks for a query, nothing was recorded. When both search methods received the same number of clicks and this number was higher than 0, both search methods are assigned one click. This comes down to the same as with Joachim's measurement model in which we assume a unique query for every click. One reason to choose for this alternative approach is because it is hard to distinguish unique queries. Moreover, one would need to determine what exactly is a single query. For example, would clicking on a result, pressing the back button in the browser and clicking a second result be considered to still be one unique query? Intuitively we think it should, but how could this be effectively distinguished from typing the query again one minute later? Also, because instant search was used, the number of queries without any clicks would turn out very high. Fact is, that for the statistical tests we perform, ties are not taken into consideration so we don't necessarily keep track of them. Keeping track of ties does give an advantage in such a way that equal numbers of clicks assigned to both search methods are not taken into account in Joachim's model. This has an advantage, because statistical tests will show significant differences with less data. For example, let's assume we would have recorded 100 clicks, from which 40 were on search method 1 and 60 were on search method 2. Now, using a two-tail sign test results in a p-value is 0.0569. However, now let's assume that 40 of those clicks were actually ties in queries.

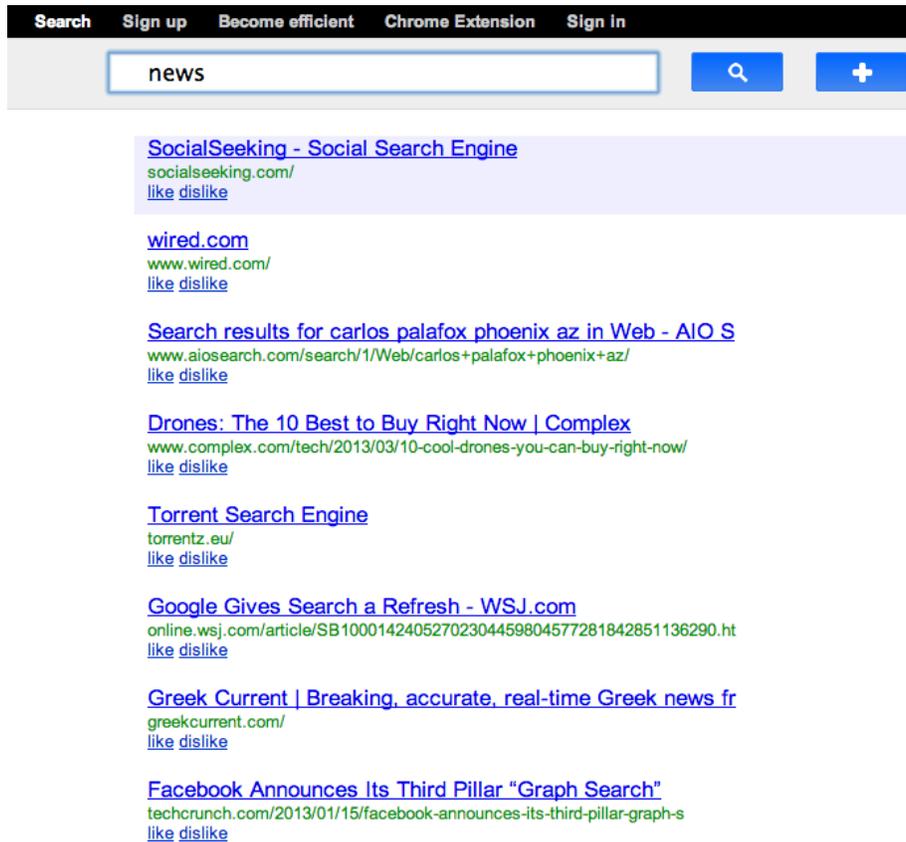


Figure 3: Example of the search interface with results shown for the query "news". It is not possible to distinguish which results come from which search method from a user's perspective.

Joachim could now use the following observed data: 20 ties, 20 clicks for method 1 and 40 clicks for method 2. Now using the same two-tail sign test results in a p-value of 0.0135. So by only taking into account clicks instead of queries, results will be weakened, particularly when there are many clicks per query. However, we did not expect many clicks per query and therefore not much loss of statistical significance. To attract a sufficient number of visitors, an online advertisement was placed.

2.5 Case study design

An embedded, closed, single-case holistic study as described by Yin, was performed with the proposed search method (Yin,2009). With the case study we evaluated the capabilities of asynchronous Social Search methods to function as

a Single Point of Access within an organisation. The search engine was hosted internally on the intranet such that knowledge became only available within the organisation. Employees used the Search Engine for five weeks and in that period more and more resources became available through the search engine. At the end of the five weeks semi-structured interviews were conducted to get information about how they experienced using the Search Engine.

The object of study was the proposed social search method. The main research question during this case study was:

Can asynchronous Social Search function as a proper Single Point of Access to information within an organisation?

A second question we wanted to answer was whether there is actually a need within the organisation for a Single Point of Access to information. To be able to answer the main research question in this case study, people from the organisation were interviewed that used the search method. Furthermore, quantitative data was assessed. The speed with which the corpus indexed by the search engine grew in combination with the number of queries and clicks could tell how long it took before the search engine had indexed a sufficiently large part of resources that are used on a daily basis. Ideally we would have liked to see the number of visits, queries and clicks increase over time and the index growth decrease over time. We would have liked to see the index growth decrease since it would indicate that the majority of regularly used resources are already in the index. We would have liked to see the number of visits, queries and clicks increase over time because it would indicate that people are making more and more use of the search method. We think that the tool should become more useful over time, since more documents are indexed and more implicit and explicit feedback on results can be used to optimise top-k ranking.

Possible outcomes of the interviews are that people would be 1) enthusiastic about the new search method, 2) negative about the quality of the search method or 3) did not use the search method at all. Possible outcomes for the development of the size of the corpus over time were constant, linear, exponential, logarithmic or even random. Constant would indicate that the size of the index would stay the same, basically that would mean the size of the indexed corpus would stay 0. Linear growth would indicate that people often use different documents during their daily work. Exponential growth would indicate that people are using more and more resources during their daily work. Last, a logarithmic growth function over time would indicate that the search method has indexed the most important documents that are used by the participants on a daily basis. Possible outcomes for the number of visits, queries and clicks would be increase over time, constant over time and decrease over time. Increase over time would indicate more users on a more frequent interval. This would suggest that people are enthusiastic about the prototype and find it useful. Constant use would suggest not much change over time in the interest of the new tool. Decrease would suggest that people do not find the tool useful and forget about it. Another important question during case study design was whether we were measuring what we wanted to measure. We guaranteed construct validity by

making use of both qualitative and quantitative data. In the case study, the cause was the introduction of the new search method in the organisation. We measured the effects of this introduction using both qualitative and quantitative data. This way we tried to guarantee internal validity. External validity based on a single case study is not really possible. However, if the results of the case study would indicate that the social search method is seen as added value to the organisation, this would be consistent with literature found that states that there is a need for a Single Point of Access to organisational information within organisations. Furthermore, there is consistent literature that emphasises the need for data lakes instead of data silos (Nanterme & Daugherty,2014). Three criteria were used for the selection of an organisation to perform the case study at:

- the employees make use of web resources in their daily work, both internally hosted as externally hosted,
- the organisation should have an information infrastructure with multiple data sources, and
- the organisation has more than 100 employees, full filling over 100 Full-time equivalents (FTEs),
- the organisation must be, at least partially, physically located in the Netherlands since it would be impractical to perform the research further away from the researchers and university given the time and budget limitations of this research.

During this case study the Technology Acceptance Model 2 (TAM2) was used to determine the usefulness of the prototype and the potential of Social Search to function as a single point of access within organisations (Venkatesh & Davis,2000). TAM2 is mainly based on TAM (Davis,1989). TAM2 was chosen over TAM because it explicitly defines external factors that influence the perceived usefulness. Perceived usefulness was used as a surrogate for usefulness. Another option would have been to use the Unified Theory of Acceptance and Use of Technology (UTAUT) (Venkatesh, Morris, Davis & Davis,2003). We chose to use TAM2 over UTAUT because UTAUT is more complex with 41 independent variables and 8 dependent variables (Bagozzi,2007). UTAUT has also been criticised for being less parsimonious than TAM2 (Van Raaij & Schepers,2008). Therefore, TAM2 is more practical to use for this case study in which only a small number of people participated and no conclusions could be drawn solely based on quantitative data. TAM2 is shown in Figure 4.

Factors directly influencing the perceived usefulness of a system are described in more detail here. Fishbein and Ajzen define **Subjective Norm** as a

”persons perception that most people who are important to him think he should or should not perform the behavior in question” (Ajzen,1991).

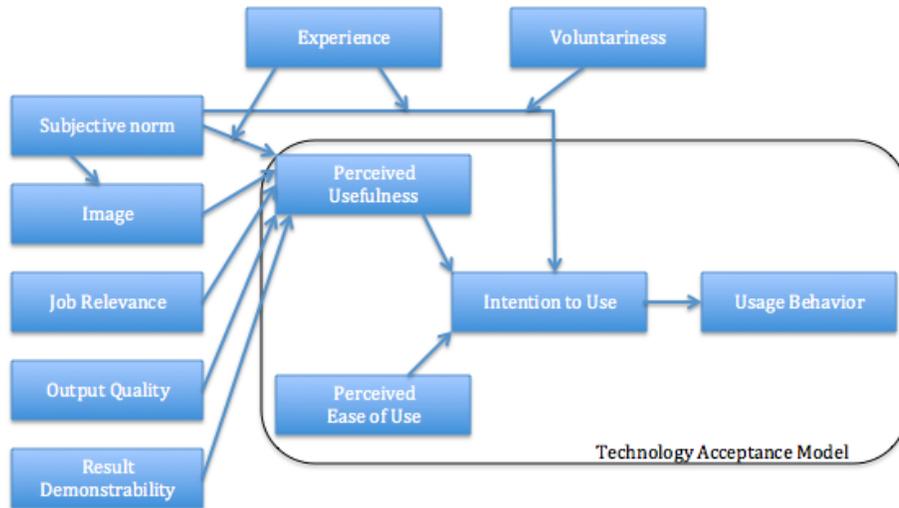


Figure 4: Technology Acceptance Model 2 as proposed by Venkatesh and Davis (Venkatesh & Davis,2000)

According to the TAM2 model, Subjective Norm only has an influence on Perceived Usefulness when people do not have much experience with the system yet. Moore and Benbasat define **Image** as

”the degree to which use of an innovation is perceived to enhance ones . . . status in ones social system”(G. C. Moore & Benbasat,1991).

The higher this degree, the more positive the influence on Perceived Usefulness according to the TAM2 model. **Job relevance** is the extent into which the system supports tasks related to the work of an employee. The better and bigger a set of tasks, that need to be performed by an employee, is supported by a system, the more positive the influence on Perceived Usefulness is according to TAM2. Moore and Benbasat define **Output quality** as

”the degree to which an individual believes that the system performs his or her job tasks well” (G. C. Moore & Benbasat,1991).

In our prototype the output is the results you get when posing a query. Participants were asked whether they were satisfied with the results they got from the system, what percentage of queries lead to success and whether they could find back recourses they added to the system. Moore and Benbasat define **Result Demonstrability** as the

”tangibility of the results of using the innovation” (G. C. Moore & Benbasat,1991).

In other words, Result Demonstrability is the degree into which positive results can be attributed to a system or innovation at hand. When people understand how a system works, they can demonstrate the results instantly. In this case study, the result should be to find information faster than you could before. If people understand how results are ranked and how they can add results to the prototype, they are able to demonstrate the potential efficiency gain of the system. **Perceived Ease of Use** is the degree into which people think that a system is easy to work with (Davis,1989). According to the TAM2 model Perceived Ease of Use has a direct positive influence on Perceived Usefulness.

3 Related work

Developing an Asynchronous Social Search engine involves a lot of work and requires knowledge from many study fields. A subdiscipline of Information Science, Information Retrieval, provides a broad base of information on web search. Next to Information Retrieval, the most relevant areas of study are Data Mining, Knowledge Management and Social and Semantic Web. Related keywords to developing an Asynchronous Social Search engine are: world wide web, search engines, pagerank, searching, indexing, crawling, user-generated content, sorting, social tagging, bookmarking, tagging, social indexing, social classification, collaborative tagging, folksonomy, database management and user interface. A systematic literature study was performed following the protocol specified in Section 2.1. This literature study was performed on November 12, 2013 and since Google updates its indexes regularly it could very well be that reproduction of this review would lead to slightly different results. The following queries were used to collect a set of resources:

1. (33) "like economy" "social web"
2. (102) allintitle: introduction information retrieval
3. (7) allintitle: large scale Web search engine
4. (8) allintitle:"social bookmarking" "web search"
5. (20) "personalized search" "user browsing behavior" "web search" -ontology
6. (26) "search engine" "collaborative web search" asynchronous implicit explicit browser -ontology -peer-to-peer -algorithm
7. (4) allintitle:social Web Search personalization
8. (3) allintitle:"browsing history" "web search"
9. (21) allintitle:user profile web search
10. (9) "user generated content" "web search engine" aardvark
11. (15) "web search" "implicit relevance feedback" "user browsing" -mouse -video -cursor
12. (26) allintitle:collaborative tagging structure

A total of 172 results was returned based on those queries. After having applied the exclusion criteria and removed duplicates, 24 results were left. The following Subsections describe the most relevant literature to this research acquired via the structured literature review.

3.1 Information Retrieval

In Information Retrieval research it has been extensively investigated how, based on a query Q , the relevant document set R can be returned. Traditional techniques look at the content of both the query Q and complete document set D and try to find matches based on this content. This way, a subset of D is returned, namely the relevant document set R . It is interesting to note that this problem does not take into account the issue of finding the document set D . Most often it is assumed that document set D is completely available. On the World Wide Web however, this is not the case and other methods have to be used to actually find an as large portion as possible of all the documents available. In these traditional techniques two methods are considered to be very useful in practice, particularly when they are used in combination.

- Term Frequency (TF): Based on the number of times a word, that occurs in the query Q , co-occurs in a document d , the document d is assigned a certain score for the given query Q . So when a query Q consists of one word, the document in which Q occurs most often is assigned the highest TF score.
- Inverted Document Frequency (IDF): looks at the words in query Q and assigns different values to the words based on whether they are common or rare across all documents in the document set D . Rare words are considered to be more important and are therefore scored higher.

By multiplying TF and IDF we get $TF-IDF$ weighting. $TF-IDF$ scores a document d for term t lowest when term t occurs in nearly all documents. It scores document d higher when the term t occurs frequently in document d . It will also score higher when term t occurs in fewer documents. $TF-IDF$ will score highest for document d when term t occurs in a small number of documents and occurs frequently in d . $TF-IDF$ as described above can be directly applied to queries Q that consist of one term t . When query Q consists of multiple terms $\{t_1, t_2, \dots, t_i\}$, simply the sum is taken of all individual $TF-IDF$ scores per term to return an ordered result set (Manning et al.,2008). To be able to measure Search Engine quality of methods like $TF-IDF$, often quality measures like precision, recall and the F-measure are used (Manning et al.,2008). However, as mentioned in Section 2.4, we will make use of a different method that was proposed by Joachim to compare Search Engine quality (Joachims,2002).

Of major influence to the field of Information Retrieval in hypertext on the web was a paper written by Sergey Brin and Lawrence Page (Brin & Page,1998). In their paper they build on technologies that were already present and added a new unique concept to determine the importance of a page. This concept, named Pagerank, calculates the importance of a webpage based on the number of references to that page, also called inlinks or backlinks. In this concept, the World Wide Web is seen as a huge graph where vertices are webpages and edges are references between pages. Just like in the scientific community, the more references you get as a page, the more valuable you are ought to be. Pagerank

also takes into consideration that it is more important to be referred to from an important webpage than from an unimportant webpage. Brin and Page showed that you can calculate the relative value of webpages in an iterative way. In practice, applying the idea of Pagerank in a search engine lead to significantly better results than before and this is most likely the major reason why Google outperformed their competitors during the millennium. Because PageRank has become so important for ranking, people started to take drastic measures to exploit it. So called link farms were setup to generate a network of links to get their pages higher in the rankings (Wu & Davison,2005).

One issue regarding retrieval in hypertext remains the identification and discovery of the webpages that are out there. A huge part of the web, referred to as the deep web, has never been indexed by any Search Engine (He, Patel, Zhang & Chang,2007). Examples of deep web sources are dynamic pages that are only generated dynamically based on queries posed by users, unlinked pages and private pages.

Another problem with respect to the ranking methods described before is that they do not take into account a user U who poses the query Q . Every user generally has different information needs, even when they are posing the same query (Sugiyama, Hatano & Yoshikawa,2004). Sugiyama, Hatano and Yoshikawa describe how search can be improved without any effort from the user by taking the browsing history of a user and constructing a user profile based on that browsing history (Sugiyama et al.,2004). They also describe shortcomings of existing personalised search systems, namely (1) the effort required from the user to make the system work is too high and (2) the system is unable to adapt to preferences from the user without explicit feedback from the user.

3.2 Social Semantic Web

A truly personal approach to Information Retrieval on the WWW has been taken by Delicious. On Delicious people can create an account, add bookmarks to it and retrieve those bookmarks later on based on tags that can be assigned to bookmarks. They can also befriend people and search in the bookmarks of their friends. Several studies were performed on whether such an approach could improve web search (Heymann, Koutrika & Garcia-Molina,2008;Yanbe, Jatowt, Nakamura & Tanaka,2007;Noll & Meinel,2007). Heymann, Koutrika,Garcia-Molina discovered that a significant part of bookmarks are tagged with terms that also occur in the title, content or metadata of the bookmarked source (Heymann et al.,2008). This suggests that it is unlikely that using bookmarks in web search would lead to much better results than a full text search. Another problem with bookmarking they discovered is coverage of the Web. Only a small portion of the web is bookmarked compared to the portion of the Web that is indexed by large search engines. However, they also conclude that pages that are bookmarked are interesting pages and 25 per cent of these pages have not been indexed by search engines. Furthermore, they conclude that bookmarked webpages are disproportionately common in search results compared to their coverage.

Bookmarking on Delicious is a form of collaborative tagging. Golder and Huberman performed research in this field of study and they define collaborative tagging as

”the process by which many users add metadata in the form of keywords to shared content” (Golder & Huberman,2006).

During their research they observed that people use a great variety of tags, but also consensus is reached in such a way that stable patterns emerge in tag proportions with respect to tagged resources. They also identify the main reason behind tagging, which is personal use. They conclude that the stable patterns in tagging can be used to organize and describe how web resources relate to each other.

Morris, in combination with several other researchers, performed studies with respect to the social aspect of search (Morris & Horvitz,2007;Morris,2008;Hecht, Teevan, Morris & Liebling,2012;Morris,2013). Morris and Horvitz created a prototype which supported both synchronous and asynchronous collaborative web search. One feature of their prototype was particularly found useful: insight in the browsing history of their own and their collaborators queries. They conclude that although there is a desire for active, small-group collaborative search, existing technologies do not adequately support this (Morris & Horvitz,2007). In 2008, Morris wrote a paper in which he explicitly concludes that new interaction techniques and new interfaces are necessary to support collaborative search (Morris,2008). In addition, he wrote another paper in 2013 in which he concludes that there is great potential for technological innovation in the field of collaborative search (Morris,2013). In 2012, Hecht,Teevan,Morris and Liebling published an article on another search engine prototype named SearchBuddies. SearchBuddies is integrated into Facebook which provides a social environment to operate in. This enabled them to make use of alternative rich feedback mechanisms such as likes. They observed that such a socially embedded search engine enables people to get answers to questions that couldn’t be answered by search engines before (Hecht et al.,2012).

Models have been created to understand the process of Social Search and to get insight in the anatomy of social search engines (B. M. Evans & Chi,2008;Horowitz & Kamvar,2010;B. M. Evans & Chi,2010). In this context, Social Search is defined by Evans and Chi as

”information seeking and sense-making habits that make use of a range of possible social interactions: including searches that utilize social and expertise networks or that may be done in shared social workspaces. This notion certainly encompasses collaborative co-located search, as well as remote and asynchronous collaborative and collective search” (B. M. Evans & Chi,2008).

Golovchinsky, Pickens and Back identified different sorts of systems for collaboration in online information seeking (Golovchinsky et al.,2009). They proposed a model which can be used to classify Social Search support tools. In this

model, different kinds of collaboration are distinguished and four dimensions are provided: intent, depth of mediation, concurrency and location. Intent identifies into which extent collaboration is explicit or implicit. Depth of mediation refers to the level on which people collaborate, where level refers to for example the User Interface or the underlying algorithms. In the concurrency dimension it is indicated whether a search method supports synchronous or asynchronous collaboration. Location identifies whether collaboration is co-located or distributed, i.e., are people physically in the same room or not. Evans, Kairam and Pirolli conducted research in which they compared an explicit, user interface level, synchronous, distributed search method to traditional search methods (B. M. Evans, Kairam & Pirolli, 2010). They observed that collaboration can help people with information seeking tasks although this type of support tool for collaboration did not perform better than traditional search methods. Such collaboration should therefore be used in combination with existing search methods they concluded.

Just like Morris, Bilenko and White found that browsing history from multiple Web users can be useful input to search methods. They took things a step further by not only looking at browsing history, but to user activity in general (Bilenko & White, 2008). They state that user activity as input for a search method can lead to improved precision and recall. Matthijs and Radlinski address the need of personalised search results and use browsing history to create a user profile, which in turn is used to rerank the top 50 search results returned by existing search engines (Matthijs & Radlinski, 2011). They conclude that their reranking outperforms the Google's default ranking.

In 2013, Gerlitz and Helmond describe a phenomenon they refer to as the Like economy (Gerlitz & Helmond, 2013). They describe what metrics were used to determine the value of webpages through the years. In the early days of the World Wide Web, the number of hits, webpage visits, was used for this. This is what the authors refer to as the Hit economy. In the late 1990s, Google introduced the hyperlink as a metric, which led to great success. This was a first step in including relational value and social validation in search engine algorithms. Gerlitz and Helmond refer to this as the Link economy. A shift from the Informational web to the Social web led to the introduction of more user-focused web metrics to determine the value of webpages. A key feature in the Social web is the use of Social buttons. Social buttons allow users to like, recommend, share and bookmark web resources. They enable new ways of information exchange and value determination. This is what Gerlitz and Helmond call the Like economy. In short, the emergence of the Like economy puts the users in charge by enabling them to provide relevance feedback on web resources.

4 Competitor analysis

This Section gives an overview of the different types of search methods that are in use nowadays. Based on the functionality of those search methods, their major strengths, weaknesses and limitations are identified. A new search method is proposed that can overcome those weaknesses and limitations. In Section 4.1, nine search methods are described. Those search methods are Google search, Google Knowledge Graph, Wolfram Alpha, DuckDuckGo, Graph Search from Facebook, Bing based on Semantic technology from Powerset, Sindice and FAROO. Furthermore, the functionality of the social bookmarking service Delicious is described because it is closely related to the search method proposed in this document. Then, in Section 4.2, the nine search methods are compared to each other. The newly proposed search method and its competitive advantages are described in Section 5.1.

4.1 Nine Search methods

In the following paragraphs every search method is briefly described. As you read through the list of search engines that are described here you might miss Yahoo! Search. This is because Yahoo! Search is powered by Bing nowadays.

Google search

Google is by far the search method that is most used for the last decade, although it was launched only in 1997. Because the owners, Brin and Page wrote a scientific paper on the functionality of the initial prototype of their search method, relatively much is known about the techniques that were used (Brin & Page,1998). It has to be noted that Google has added many secret criteria for determining the ranking of webpages since then. However, for as far as known, Google is heavily based on PageRank, which is a method proposed by Brin and Page to rank webpages according to the amount of incoming links a webpage has. Furthermore, Google search makes use of Term Frequency (TF) combined with Inverted Document Frequency (IDF) techniques. As mentioned in Section 3.2, TF analyses which words are used in which frequency in a document and IDF analyses relevancy of words based on in how many documents the words occur. As also mentioned in Section 3.2, the combination of these techniques is referred to as $TF-IDF$. Without going into too much detail, this appears to be a popular technique to match queries to documents (Aizawa,2003). To discover new web sources, Google makes use of crawlers. In 2012, Google claimed to have indexed 30 trillion webpages. After crawlers find webpages, Google inspects the HTML document including Meta tags and takes it all into account during the indexing process. Another technique that Google uses in trying to achieve as accurate results as possible is providing people with personalised results. This entails that Google tries to identify users and then provides specific results to those users based on the profile it maintains of that user. Although the techniques that Google uses are vulnerable to spam in many ways, the organisation

has taken measures to prevent spamming and the search engine provides decent results in many cases. However, the more commercial topics are often manipulated by companies that hire Search Engine Optimisation organisations and thereby paying to be on the first page given certain keywords. A study regarding the effectiveness of several methods to turn up higher in the rankings of Google was conducted by Evans (M. P. Evans,2007). One simple example of malicious use is making many webpages link to your website to increase the PageRank of your website.

Bing based on Semantic technology from Powerset

Microsoft launched Bing in 2009. In many ways, Bing functions the same way as Google search does. One notable additional feature that has been added to Bing is so called semantic search featured by Powerset, a company that Microsoft acquired in 2008 (Hendler,2009). In this context, semantic search refers to search that tries to understand what your query actually means and in which context it should be placed.

Graph search from Facebook

Facebook Graph Search works fundamentally different than Bing and Google. First of all, Facebook uses the knowledge it has about you as a Facebook user to provide results. It is a highly specialised search engine focused on social queries. Furthermore, the knowledge about others in the network is used. A typical query that Facebook would be able to answer where Bing and Google would miserably fail would be: "photos of my friends taken in Amsterdam". Just like Bing, Facebook also makes use of semantic technology to understand such queries. Facebook also takes into account likes and tags to provide search results. Facebook works together with Bing to be able to also answer more traditional queries. Notice that in this research Bing results on Facebook will not be taken into account because it is a completely different search method.

Google Knowledge Graph

In 2012, Google added another way of search to its search engine, named the Google Knowledge Graph. We already elaborated on Google search, which is the more traditional way of how Google answers queries. The two methods are currently working together if you are posing queries to Google. However, a clear distinction can be made about which method has returned which answer. Because its a completely different way of handling queries than Google search, we treat and describe it as a different search method here. The distinction between the two methods can be identified as follows: Google search will provide you links to information on other webpages around the web whereas Google Knowledge Graph provides you information directly. The majority of queries only returns Google search answers, but an example of a query that returns answers using both methods is "paintings from Picasso". Notice that Google does not

provide the same search results to everyone and search results are continuously changing so it could be that Google wont show you Google Knowledge Graph results for this query. Just like Graph search, Google Knowledge Graph heavily relies on semantic search. Googles Knowledge Graph contained 570 million objects and over 18 billion facts about and relations between those objects in 2012. Google has acquired this knowledge from many sources including the CIA World Factbook, Freebase, and Wikipedia. However, crawlers are also used to search for documents with semantic markup. Google also enables people to provide feedback if facts or relations are not correct.

Wolfram Alpha

Wolfram Alpha answers factual queries by computing answers based on externally sourced curated and structured data. Most of the data that Wolfram Alpha uses to answer queries comes from systematic primary resources. This means that no crawlers are used to search for information to put into the internal knowledge base of Wolfram Alpha, but a selected and verified group of data providers is consulted using more structured ways to communicate data such as using APIs instead of HTML pages. Although Wolfram Alpha is good at answering queries that are fact based, it is weak at answering queries that require opinions and social thinking.

Delicious

Delicious tries to enable people to store and to share links to all their relevant web resources. It also tries to enable people to discover new relevant web resources. Delicious was launched in 2003 and enables people to tag webpages and discover them later on. In other words, Delicious is an online bookmarking service (Golder & Huberman,2006). When people bookmark webpages, users can choose whether this bookmark should be private or public. An advantage of tagging is that also webpages with restricted access can be indexed, whereas crawlers do most often not have this capability. A disadvantage of this method is that the system is dependent on the amount of people that use it for discovery and searching of relevant web sources.

DuckDuckGo

In 2008, Gabriel Weinberg launched DuckDuckGo. DuckDuckGo primarily distinguishes itself from other search engines by not storing any personal information of its users, providing the same search results to everyone and thereby preventing the so-called filter bubble. The Filter bubble is a result state in which users are separated from viewpoints they disagree on, isolating them into their own cultural or ideological environment (Pariser,2011). Consistently, the organisation claims to protect the privacy of its users. DuckDuckGo makes use of other search engines such as Bing and Wolfram Alpha. Information it acquires itself is mainly retrieved from crowd-sourced websites such as Wikipedia.

Furthermore, DuckDuckGo makes use of instant answer plugins. Such plugins provide instant answers to queries such as "weather Netherlands" instead of links to external webpages. Such instant answers are also called "Zero-click Info" boxes. Instant answer plugins discover data from external sources in a structured way just like Wolfram Alpha does.

Sindice

Sindice is a search engine that works by crawling the Web of Data, which is composed of webpages that have semantic markup such as Resource Description Framework (RDF) triples. Such triplets consist of a subject, object and the relation between the subject and the object. Sindice crawls the Web of Data and because of the semantic markup that is contained on the pages of the Web of Data, it is capable of reasoning about concepts and relations between concepts rather than keywords. Sindice contains several billions of triples in their dataset. Just like Facebook Graph Search and Google Knowledge Graph the search method heavily relies on semantic interpretation of queries and data in the database.

FAROO

FAROO is a search engine that is fully based on peer-to-peer technology (*FAROO*,2007). Crawling, indexing and searching are all decentralised. Data is stored on computers of consumers and pages that are visited by the peers are automatically indexed and distributed. Ranking of results is based on user behaviour, also called attention based ranking. FAROO also crawls webpages based on user behaviour, thereby discovering parts of the deep web, which cannot be found by crawlers that are dependent on links and input from websites to notify them about the existence of those pages. The part of the deep web that can be discovered by FAROO is the part that is presented to users in a to humans understandable way. A good example of a human understandable format is an HTML page processed by a browser resulting in a clear overview. An example of deep web that is not directly understandable to humans is a rawly presented JSON file. Because of the peer-to-peer approach FAROOs search method scales very well. Therefore, the more people who use it, the more web resources can be indexed and the better the results can become. Currently, FAROO has indexed over 2 billion webpages.

4.2 Comparison of the nine search methods

To be able to compare the nine search methods, multiple competitor arrays have been created. In Table 1, for every search engine it is indicated on a scale from 1 to 3 whether the search engine makes use of crawling, tagging, structured data transferring and browser tracking to discover information sources. 1 indicates that the search engine is expected to not use this method or it only has a minor impact on the discovery method, 2 means that the search engine does

use the method, but it is not used extensively and 3 means that it is believed that this search engine uses this method extensively. This information has been acquired by looking at information that has been made publicly available by the organisations behind the Search Engines. Table 2 indicates for all search methods into what extent they make use of several ranking methods. Although detailed information on discovery and ranking methods is often considered as classified by most Search Engines, a general indication of what types of methods they use is often publicly stated on their websites. For example, Google provides information to web authors on how they can show up higher in the rankings and how Google tries to find webpages. Table 3 indicates which sources a search method can discover based on the methods used for discovery as shown in Table 1. Table 4 gives an overview of three characteristics and into what extent each search method has these characteristics. One of those characteristics is the proneness of the search method to spam and malicious use. The other two characteristics are user-independency and the topic scope. Table 5 indicates into what extent every search method supports certain general features. Those features include personalised search, non-personalised search, suggestions during query typing and instant search results during query typing.

Tables 1 to 5 also contain values for the newly proposed search method under the alias "New". Section 5.1 will explain how those values are determined. Notice that numbers in those tables are based on the pure service that a search method is offering itself and support from other search methods within search methods is not taken into account. An example of using a search method within a search method is a so called meta search engine, which discovers its sources from other search engines.

Discovery methods

From Table 1 it can be derived that Google, Bing, Google Knowledge Graph, Sindice and FAROO make the most extensive use of crawling to discover data. Delicious makes use of tagging to discover their data. Users assign keywords to webpages and can retrieve those resources later on by searching for those assigned keywords (Gupta et al.,2010). Wolfram Alpha, DuckDuckGo and Sindice make use of structured data transferring methods to retrieve their data. DuckDuckGo for example, makes use of real-time data from a third party to provide weather forecasts. As the only one, FAROO makes use of anonymous user behaviour tracking to discover new web resources. This means that FAROO anonymously keeps track of the URLs that people visit and uses them as starting points, also called seeds, for crawling the web. Facebook Knowledge Graph does not make use of discovery methods because all the data is already available within their own knowledge base.

Discovery capabilities

The discovery methods that a search method supports determine what resources can be indexed and which can never be found. In particular, using crawling

and structured data transferring limits the resources that can be indexed significantly. Crawlers are limited to the links they know about and the links they come across while crawling. Structured data transferring is also very limited because only a preselected set of sources can be discovered this way. Tagging and tracking browse behaviour on the other hand are way less restrictive. Whenever someone finds a resource that is relevant to him he can tag the resource or in the case of tracking browse behaviour, the discovery method instantly knows about the existence of the web resource when a user accesses the resource. Although tagging and tracking browse behaviour are less restricted in the web resources they can discover, the methods also have clear disadvantages. Tagging requires user effort and tracking browse behaviour can lead to a lot of privacy issues. Crawling and structured data transferring do not have these downsides. Delicious and FAROO are capable of discovering more resources than the other search engines. Table 3 provides an overview of different types of web resources and into what extent every search method is capable of discovering those types of resources.

	Crawling	Tagging	Structured data transferring (e.g. using APIs or Database snapshots)	Tracking browse behaviour
Google	3	1	1	1
Bing	3	1	1	1
Facebook	1	1	1	1
Knowledge Graph	3	1	1	1
Wolfram Alpha	1	1	3	1
Delicious	1	3	1	1
DuckDuckGo	2	1	3	1
Sindice	3	1	3	1
FAROO	3	1	1	3
New	1	3	1	3

Table 1: Discovery methods.

	PageRank	HTML	TF-IDF	Tagging	Semantic	Liking	Click through	Tracking browse behaviour
Google	3	3	3	1	1	1	1	1
Bing	3	3	3	1	2	1	1	1
Facebook	1	1	1	3	3	3	1	1
Knowledge Graph	1	1	1	1	3	1	1	1
Wolfram Alpha	1	1	1	1	2	1	1	1
Delicious	1	1	1	3	1	1	1	1
DuckDuckGo	2	2	2	1	2	1	1	1
Sindice	1	1	3	1	3	1	1	1
FAROO	1	2	3	3	1	1	1	3
New	1	2	2	3	1	3	3	3

Table 2: Ranking methods.

	Public webpages	Webpages with restricted access	Deep web	Any other resource (both on and offline)
Google	3	1	1	2
Bing	3	1	1	2
Facebook	1	2	1	1
Knowledge Graph	3	1	2	2
Wolfram Alpha	2	2	2	2
Delicious	3	3	1	1
DuckDuckGo	2	1	2	1
Sindice	1	1	2	2
FAROO	3	1	2	1
New	3	3	3	3

Table 3: Discovery capabilities.

	Topic scope	Spam	User-independent
Google	2	3	3
Bing	2	3	3
Facebook	1	2	1
Knowledge Graph	2	1	3
Wolfram Alpha	1	1	3
Delicious	3	3	1
DuckDuckGo	2	1	3
Sindice	2	3	3
FAROO	2	2	1
New	3	2	1

Table 4: Three relevant properties of the nine search methods: size of the scope in which queries can be handled, proneness to spam and the extent into which the method is independent of its users.

	Personalised search	Non-personalised search	Suggestions	Instant search
Google	3	1	3	3
Bing	3	1	3	1
Facebook	3	1	3	1
Knowledge Graph	1	3	3	1
Wolfram Alpha	1	3	3	1
Delicious	3	1	1	1
DuckDuckGo	1	3	1	1
Sindice	1	3	1	1
FAROO	1	3	3	3
New	3	3	3	3

Table 5: Four relevant features of the nine search methods: degree into which it supports personalised and non-personalised search, whether it provides suggestions regarding queries while the user is querying and whether instant search results are provided.

Ranking methods

Table 2 gives an overview of which methods are used by each search method to rank results. Bing and Google search make use of Pagerank, HTML content of webpages and TF-IDF. In these cases, importance of pages is determined using the amount of backlinks a page has and the keywords of the pages are determined by the HTML content of the document. The keywords are analysed during query processing using TF-IDF ranking. Facebook, Delicious and FAROO enable users to tag resources and this is taken into account when search results are being ranked. Facebook Graph Search, Google Knowledge Graph and Sindice use semantic methods to determine actual meaning of queries, documents and objects. This way, resources are ranked based on conceptual equality instead of keyword matching. Facebook also takes into account likes. Things that are liked more can be ranked higher. Ranking can also be based on clicks from the user on the provided results. The basic idea is simple: the more often a user clicks on a provided web resource after a query, the more relevant that result is provided that query. As far as known, none of the search engines mentioned here make use of this ranking method. Last, FAROO makes use of browser tracking to rank results. Pages where more time is spent should be ranked higher.

Spam and malicious use

All search methods are prone to spam and malicious use, but in different degrees and in completely different ways. Proneness to spam and malicious use is not only dependent on the ranking method, but also on the discovery method. For every search engine, we will now describe their main vulnerabilities to spam. Google and Bing are mainly vulnerable for spam because they make use of PageRank. People that add a lot of backlinks to their websites can influence PageRank (M. P. Evans,2007). There are companies that are specialised in creating so called link farms. This, in combination with smart choices of HTML content is the main weakness of Bing and Google regarding spam and malicious use. Facebook Graph Search is less prone to spam and malicious use because it provides highly personalised results in an environment controlled by Facebook itself. However, people can still tag and like resources with wrong intentions. They could also create heaps of fake Facebook accounts, but this is not directly a threat, because people will have to become friends with those fake users before it will heavily influence their search results. However, when spammers are making those accounts publicly available, Facebook Graph Search might take them up in their search results to any user. Basically the major threat here is the creation of fake users that purposely act in a certain way to generate certain search results. For Delicious and FAROO this is also the major threat. Google Knowledge graph is also not that vulnerable to spam as Google search and Bing. The major threat here is that people will inject incorrect, malicious content into their webpages that will be crawled by the search method. However, data will be checked before it will be made available via Google Knowledge Graph, so also in this case, the environment is more controlled. For Sindice, the same holds,

but control is not in place, resulting in higher vulnerability to spam. Wolfram Alpha and DuckDuckGo are the least vulnerable to spam because they discover and retrieve their data mainly through Structured data transferring methods that are preselected and verified. The major threat here is that an organization convinces the search method that they are trustworthy, but will then provide spam or biased information later on.

User dependency

Another property of search methods is their dependency on its users. Here, the question is whether the quality of the search method relies on the amount of users that the search method has. Google search, Bing, Google Knowledge Graph, Wolfram Alpha, DuckDuckGo and Sindice do not rely on user generated content nor behaviour and are therefore independent of the amount of users. Delicious however, does rely on users. Discovery of new web resources does not work properly when there are only a few users. Users can still use the system independent of other users by storing their own web resources and making use of highly personalised search. For FAROO, this is not the case. Because FAROO does not personalise results, the search engine would come up with low quality results when the number of users is low. On the other hand, FAROO has the potential to provide high quality results when many users use it because of its scalability. Unlike any other search engine, FAROO also depends on its users with respect to computing power since it makes use of a fully decentralised architecture.

Topic scope

Not all search engines handle all topics. In general, the bigger the topic scope, the higher the diversity of types of queries it can handle. Wolfram Alpha has a rather limited topic scope, mainly focused on answering queries about mathematical and numerical facts. Facebook Graph Search also has a limited topic scope. It is only able to provide results to queries about information available in the social network, in particular your social environment. Google Knowledge Graph has a broader scope. It copes with all factual data. At the moment however, it does not support subjective data, such as opinions about Amsterdam. Google search, Bing, DuckDuckGo and Sindice support a significantly broader scope, namely all the content that is publicly available on World Wide Web. However, personal content cannot be discovered by any of those search methods. Delicious on the other hand does support retrieval of personal content and thereby has the broadest possible topic scope. This means that only the search method used by Delicious has the potential to provide proper results to all possible queries on all possible topics whereas all the other search methods do not have this potential in their current state. This is quite ironic because Delicious does not even call itself a search engine.

Features

Here, a few relevant features of search methods are compared, starting with personalisation of search results. The ranking methods that a search method uses determine the degree into which results can be personalised or non-personalised. Furthermore, it also influences how prone the search method is to spam and other malicious use. Pagerank, HTML analysis and TF-IDF are all methods that enable non-personalised search, but do not directly support personalised search. Tagging, liking, click through data tracking and browse behaviour tracking enable both personalised and non-personalised search. Both are supported because you can look at individual actions, actions of friends, but also at the aggregated amount of actions of all users. When looking at individual or friends level, results will be personalised. When looking at the aggregated level of all users, non-personalised, democratic results can be provided. Semantic ranking can also support both personalised and non-personalised search because it could optionally take into account a user profile and a social graph. Table 4 gives an overview of the extent into which search methods support personalised and non-personalised search. Personalised search has both advantages and disadvantages. People do not like to be tracked and profiled, but generally, it appears that they do like the results more if they are personalised. Google, Bing, Facebook and Delicious all support personalised search. Google Knowledge Graph, Wolfram Alpha, DuckDuckGo and Sindice all support non-personalised search. It is interesting to note that none of the search methods gives you the option to choose between personalised search and non-personalised search.

A handy feature that some search methods provide is suggestions for queries during the typing of a query. For instance, if you type in "New York", the search method could provide you with suggestions for queries such as "New York Pizza" and "New York Times". Google search, Bing, Facebook Graph Search, Google Knowledge Graph, Wolfram Alpha and FAROO provide this feature whereas Delicious, DuckDuckGo and Sindice do not provide this feature. Providing suggestions for a query to a user to be able to further disambiguate what the user is looking for is also done in a more explicit way by some search engines. With this feature, when a user has typed in a query, results are shown but also a visualisation of the results is shown in the form of a list of words that are somehow related to the query. The user can select words from this list to focus his search. An example of a search engine that provides this feature is Clusty, created by Vivisimo. Determining which words should be shown to the user is a task called clustering that has been studied extensively in the field of Data Mining. Notice that clustering is not necessarily about grouping documents. It can be applied in a broader scope, grouping a set of objects in such a way that objects in the same group are more similar to each other than to objects in other groups (Rajaraman & Ullman,2012). Although the user experience is different with suggestions during query typing compared to showing related words to a user, the concept and goals behind the two features are similar.

The last feature discussed here is instant search results. If this feature is supported by a search method, only providing a very short query already returns

results to the user instantly. There is no need to click the search button or press enter because the search method acts like you submit your query after every character you type. You could even imagine that the search results that are ought to be the most relevant to the user would already be provided before the user starts typing his query. Google and FAROO support this feature, although both do not provide instant results in all cases.

Summary

In the end a search engine should, given a query, provide the most relevant resource as fast as possible. Therefore, search engine quality can be determined based on the following three factors: speed, discovery capabilities and quality of the ranking function. Notice that speed does not only refer to the time it takes to process the query. Instead speed refers to the time it takes the user to identify the resources that are the most relevant to him. Therefore, speed is also influenced by the format in which the results are presented and features like instant search and suggestions. Table 6 provides an overview of how the search methods score on those three factors. The provided scores are based on Table 1 to 5.

	Speed	Discovery capabilities	Ranking quality
Google	3	1	3
Bing	3	1	3
Facebook	3	1	2
Knowledge Graph	3	1	2
Wolfram Alpha	2	1	1
Delicious	1	2	2
DuckDuckGo	2	1	2
Sindice	1	1	1
FAROO	1	2	2
New	3	3	3

Table 6: Overall search engine quality

5 Model development process

In the following Subsections the advantages of asynchronous Social Search over traditional search methods are described, multiple factors are explained that can support user involvement, Critical Success Factors are identified and requirements are specified for a prototype.

5.1 Competitive advantages of a new search method

Next, we propose a search method that has the potential to provide better search results than all discussed search methods so far. This search method

- Has significantly better Discovery capabilities based on the combination of Discovery methods that are used,
- Combines a unique set of ranking methods to enable democratic ranking of results and improved ranking quality in general,
- Gives users the explicit choice between personalised and non-personalised search, and
- Should provide a user interface for content that is presented in unreadable format for humans such as JSON. This will make content in the deep web accessible to users.

Better Discovery Capabilities

To be able to discover all sorts of documents, deep web and any other resources, the proposed search method makes use of tagging and browser tracking. Tagging will be used to enable users to tag every web resource they want. Because tagging is labor intensive it should be made as easy as possible for users to tag resources. Advanced users should be able to write macros such that they can tag multiple web resources with only one tag assignment. An example is that a user tags resources with the tag "map of X" where the user provides a list of possible values for "X" and a corresponding list of URLs. All users should be provided with suggestions for tags as well and should be able to tag resources with only one or two clicks while being at a web resource they like. This can be achieved using so-called bookmarklets or browser-specific plugins. Such advanced tagging methods have never been used before in search engines. Browser tracking is also a way to decrease tagging effort and increase the speed of discovery of new web resources. Browser tracking will be available to users that are willing to be anonymously tracked while browsing the web.

Personalisation and improved ranking

None of the search methods that were compared in Section 4.2 gave the user the choice between personalised and non-personalised search. The search method

proposed here should support both in a transparent way, taking away the suspiciousness regarding user profiling. The proposed search method can also actually support both personalised and non-personalised search because of the ranking methods that it makes use of. Likes, tags and click through data can be looked at in aggregate form and on a personal level. Ranking of results will be based mainly on likes, tags and click-through data. This way, democratic ranking is enabled, which should result in better results than existing ranking methods.

A user interface for the deep web

Advanced users should not only be able to tag resources in the deep web using macros, but must also be able to provide information on how the data should be styled when it is presented to the user. One example would be that a user knows about a list of soccer teams in Amsterdam available on the deep web in JSON format. In this case, the user should be able to indicate how the JSON should be converted to understandable HTML that is user friendly. Such markup indications should, of course, be as straightforward and easy as possible to create. Furthermore, the user should not be forced to provide markup. When a user does not provide markup information, markup should be automatically generated. Notice that such tools are not exotic ideas of the future but already exist and are available to the public for free. An example of such a tool is called json2html and can be found on <http://json2html.com/>.

5.2 Getting people involved

Opposite to most search methods, the proposed search method is depending on the people that use it. Theoretically, when the search engine has never been used, it would not be able to produce any result. Therefore, it is important to involve people in using the search engine.

As mentioned before, examples of systems that successfully involved users in generating content are Wikipedia and Stack Overflow. Much can be learned from the setup they used to get people involved. Forte and Bruckman conducted a study on why people write for Wikipedia and, more general, why people contribute to Open-Content Publishing (Forte & Bruckman,2005). In their study they conclude that there are similarities between the incentive system used in the scientific community and Wikipedia. Particularly, the cycle of credit, as described by Latour and Woolgar appears to be similar to the incentive system used by Wikipedia. Latour and Woolgar found that in the cycle of credit incentives are allocated in the form of power, resources and the ability to reinvest resources to be able to acquire more and more credibility (Latour & Woolgar,1979). Forte and Bruckman claim that an incentive economy works well to get and keep people involved in a community. They suggest that hard coded stratification of privileges should be kept to a minimum. Hard coded, stratified privileges are the rights of users that are explicitly stated and cannot be changed easily within a system. An example is that only an administrator is allowed to delete a post from a forum. According to them, keeping hard

coded stratification of privileges to a minimum is necessary in order to allow active participants to achieve higher levels of responsibility and efficacy within the community. This way, leaders can emerge in a democratic way. Forte and Bruckman also suggest that technology should be used to measure involvement of participants. For example, keeping track of the amount of comments someone posted.

Although incentives are an important factor in getting people involved in a community, it is definitely not the only way. Asynchronous Social Search makes use of tagging to relate web resources to keywords. There are many reasons for people to tag resources. Gupta, Li, Yin and Han provide a list of reasons why people would be willing to tag resources in their paper about social tagging techniques (Gupta et al.,2010):

- Future retrieval: this is the most prominent reason that people will have to tag resources when using a search engine. By providing metadata about the resource a person is able to find the resource back later on.
- Contribution and sharing: by tagging a resource you can make it available to not only yourself, but to the entire community or a part of the community.
- Attract attention: when there is a resource that a person wants to show to the rest of the community, he or she could tag it with popular tags that other people often look for. This way, popular tags are exploited. Although this does not always have to be a bad thing, tagging resources to attract attention comes close to spam and malicious use of the search method.
- Play and competition: tagging can become a game such as "tag the resource with what you think others would tag it with".
- Self presentation (self referential tags): indicates the type of relation between the resource and the person who created the tag. Examples are "my house", "my e-mail" and "been there".
- Opinion expression: tags can be used to indicate what you think about a resource. Do you find it good, bad, boring, interesting, etcetera.
- Task organization: people can also use tags to order their planning. Such tags could be "todo" or "done".
- Social signaling: tags can be used to inform others about the context of a web resource.
- Money: it is possible to pay users to tag resources.
- Technological ease: getting users involved can be achieved by making the effort to participate as low as possible and making participation fun. So when tagging becomes easier more people will tend to do it. Sometimes

it even becomes easier to tag than not to tag. Think about Facebook that proactively asks you to tag people in pictures with the correct person names assigned to the right person in the picture already.

By supporting as many reasons as possible to tag resources you will give people ever more reasons to become involved. This, in combination with minimization of hard coded stratified privileges could provide a decent basis to get people involved in contributing to the Asynchronous Social Search method.

5.3 Critical success factors

The Critical Success Factor (CSF) that holds specifically for a asynchronous Social Search method is user involvement. Without involvement of users asynchronous Social Search methods will never work because the method relies on input from users. Therefore, involvement should be easy, fun and make the user more productive. Examples of successful systems that rely on user involvement are Wikipedia and Stack Overflow. Although Wikipedia is considered to be one of the biggest websites in the world it had only 40 000 active contributors in 2006 (Ortega, Gonzalez-Barahona & Robles,2008). This indicates that only a fraction of people in the world have to become active contributors to serve the entire population with decent quality. One way of getting users involved is by keeping the effort required as low as possible (Golder & Huberman,2006). Therefore, efforts to tag and like web resources should be minimal. Also, the system should heavily rely on implicit feedback instead of explicit feedback to reduce efforts of users. To be able to rank resources based on both implicit and explicit user feedback, the Hit economy and the Like economy as described by Gerlitz and Helmond need to be supported by the system to determine the value of webpages (Gerlitz & Helmond,2013). A good option to support the Hit economy is that users should be able to enable automatic browser tracking behaviour so that every website that is visited by the user automatically gets indexed. Benefits of active participation should be maximised by providing active participants with search results they indicated were the most relevant to them regarding certain keywords. Results should be provided on an instant basis, changing with every key pressed on the keyboard by the user. Ninety per cent of the queries should return the most relevant results for that user within the first three key-strokes.

A CSF that holds for virtually all search methods is spam and malicious use prevention. People want their website to come up at the highest rank in the search engine and try to achieve this in all sorts of ways. In the case of PageRank for example, one might try to create heaps of webpages referring to one page that you want to come up on top in search engines. Another CSF is performance. Queries of users must be handled in milliseconds to be able to provide search results to users instantly. Because the web is very large and still expanding every moment, scalability is also very important. Although the asynchronous Social Search method can be applied in small communities, its real strength lies in massive collaboration. The amount of resources to index

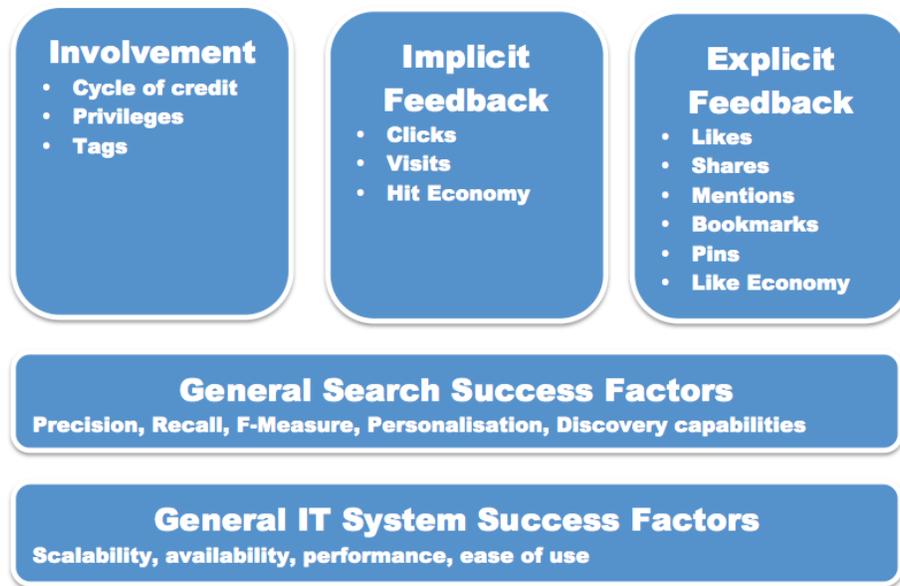


Figure 5: The key success factors of asynchronous Social Search.

will grow with the number of active participators, but also over time. Just like other search methods, it is also critical to have your search method always available to everyone on the internet. Therefore, 100 per cent uptime should be strived for. Just like any other system, the user interface of the search method has a critical impact on whether people will use the search method or not. The focus should be on ease of use and simplicity. Although people do not seem to like the idea most of the time, one CSF of a search method appears to be personalised search results. The search method should enable personalised search in a transparent way such that users can understand and manipulate their personal results to queries. Furthermore, users should be given the explicit choice to enable or disable personalised search. People should also be able to create and join groups that maintain their own corpus of resources. Joining such groups should influence the personalised ranking results. It should also be possible to search in the corpus that a group maintains. Table 7 provides a summary of the critical success factors of asynchronous Social Search. Figure 5 gives an overview of the main CSFs of asynchronous Social Search in the form of a model.

Factor	Description
Involvement	Without involvement of users asynchronous Social Search methods will never work because the method relies on input from users (Robey & Farrow,1982). A cycle of credit, much freedom for users and tagging support could support user involvement (Latour & Woolgar,1979;Ortega et al.,2008;Golder & Huberman,2006).
Implicit Feedback	Implicit feedback such as clicks and visits has been proven to be very useful for both discovery and ranking of search results (Matthijs & Radlinski,2011). This all comes down to supporting the Hit economy (Gerlitz & Helmond,2013).
Explicit Feedback	Explicit feedback such as likes, shares and pins gives people the possibility to directly influence the rankings according to their point of view. This enables true democratic ranking (Gerlitz & Helmond,2013).
General Search Success Factors	More general success factors such as precision, recall, personalisation and good discovery capabilities also hold for asynchronous Social Search. (Manning et al.,2008)
General IT System Success Factors	Even more general success factors that hold for most Information Technology systems that also hold for asynchronous Social Search are scalability, availability, performance and ease of use (Chung, Nixon, Yu & Mylopoulos,2000).

Table 7: The key success factors of asynchronous Social Search.

5.4 Requirements specification of a prototype

To be able to validate the hypothesis that asynchronous Social Search provides better quality in search results than computer based search a prototype was built. There are three ways in which results can be added to the prototype. The first is manually, by filling in a URL, title, description and keywords. Figure 6 provides a screenshot of what this way looks like in practice. The second way is by adding a bookmarklet to your favourites in your web browser. When a user has the bookmarklet in his favourites list in his web browser and he visits a website, he can click on the bookmarklet. This results in a popup of the search engine with a form shown to add a result to the search engine. In this form, the URL, title, description and keywords are already filled in based on the page that the user is currently visiting. This second way of adding webpages to the search engine is less time consuming than the first. An example is shown in Figure 7. The third way to add search results to the search engine is by installing an extension for the Chrome web browser. By installing this extension, all the websites that are visited by the user are added to the search engine automatically. When a result is added using the extension, all content of the page that is being added is indexed in case that the page makes use of the HTTP protocol and not of the HTTPS protocol. To guarantee a decent corpus size, the API of bookmarking website Delicious was used to enrich the corpus with resources tagged publicly on Delicious.

A ranking of search results given a query Q is based on credits. Credits are assigned to search results regarding queries in two ways. First, a user can like and dislike a search result given a query. Second, when a user clicks on a search result, credits are assigned to the result provided the query. A click and like both add one credit to the current credit score of a search result and corresponding query. A dislike subtracts one credit of the current credit score. Newly added search results always start with a credit score of zero. A query Q consists of characters $\{c_1, c_2, \dots, c_n\}$. To provide people with instant search results during query typing, credits are not only assigned to a result list R given Q , but also given the queries $\{c_1, c_2, \dots, c_{n-1}\}, \{c_1, c_2, \dots, c_{n-2}\}, \dots, \emptyset$. This way, by only typing a few characters, a user is already provided with the most popular search results. Therefore, the search engine supports instant search results during query typing. We refer to this proposed search method as the social instant method SI .

When less than ten results are returned using this search method, a fallback method is used. This fallback method makes use of full-text search F , which is a search method based on TF-IDF principles. The thousand best results based on full-text search F were then reranked based on a formula which involves a social score, the length of the query and the full-text score. The social score S , is an indicator for relevance of a resource based on the number of likes, shares and other interaction forms with the page via multiple Social Media platforms. The social score S is calculated as follows:

$$S = \frac{\sum_{i=1}^N \log_{10}(1 + M_i)}{N}$$

Search Sign up Become efficient Chrome Extension Sign In

Keywords e.g., fruit, meat, dinner, best vegetarian recipes, top 10 herbs

URL e.g., food.com

Title

Description

Mini Image URL

Figure 6: Screenshot of how a link can be added manually.

www.food.com

Search 475,000 Recipes

Sign In or Register

what's new top recipes holidays meal planning talk & tips TRENDING: \$3,000 Recipe Contest, 30 Mind-Blowing Kitchen Tips

Popular Categories

- Slow Cooker Recipes
- Chicken Recipes
- Most Popular Recipes
- Main Dish Recipes
- Easy Appetizers
- Salmon Recipes
- Beef Chili Recipes
- Popular Cod Recipes
- Copycat Recipes
- Ground Beef Recipes

Top Trending Searches

1. Weight Watchers Recipes

“ We live on top of a hill with

Just Added

Recipe of the Day

Recipe: Banana Split Sakes

by HistoryShopper

like dislike

Food.com - Thousands Of Free Recipes From Home Chefs With Recipe Ratings, Reviews, and More

http://www.food.com/

Food.com - Thousands Of Free Recipes From Home Chefs With Recipe Ratings, Reviews, and More

Food.com has a massive collection of recipes that are submitted, rated and reviewed by people who are passionate about food. From international cuisines to

http://www.food.com/favicon.ico

Figure 7: Example of how a link can be added using the bookmarklet.

In this equation, M is a list of interaction scores from N social media platforms. In the prototype, N equalled seven. The social media platforms used were Facebook, Twitter, Pinterest, Google+, StumbleUpon, Delicious and Linked In. An example of calculating the Social Score S for a document d only taking into account two Social Media platforms is as follows: let's say that the website "example.com" has 99 likes on Facebook and has been mentioned in 9 tweets. Then, $M_1 = 99$, $M_2 = 9$ and $N = 2$. We calculate the Social Scores per media and divide by N . The Social Score for Facebook is $\log_{10}(1 + 99) = 2$ and the Social Score for Twitter is $\log_{10}(1 + 9) = 1$. To calculate Social Score S for document d we take the average resulting in $S = 1.5$. As you can see in this example, the Social Score increases with the number of likes and shares. Only a Social Score per page does not make a good ranking algorithm because it does not take into account the query Q posed by user U . Full-text search results are therefore reranked according to the following formula:

$$SF = F \times (S + (Q_w * 5))$$

In this formula, Q_w refers to the number of white spaces in Query Q . From this formula, you can deduct that the more words are used in a query Q , the less Social Score S is taken into account. When more words are present in a query, it is probably less ambiguous and it is more important to look at the full-text score F , the match between the query Q and the content of a document d . When a query contains only one or a few words, it is hard to determine the meaning only based on the content. Then we rely on the popularity of pages more heavily. We refer to this ranking method as the Social Full-text method SF . The final score in the fallback search method SF is calculated for the thousand web resources returned by the full-text search method F . The ten resources with the highest SF scores are appended to the results from the base search method in descending order of SF score.

Users can sign up for an account in the prototype. Because of privacy issues, the Chrome extension differentiates between the HTTP and HTTPS protocols. When the HTTP protocol is used when a user accesses a web resource, the resource is added to the search engine publicly, meaning that everyone can retrieve it. When the HTTPS protocol is used however, the web resource is only stored when the user is signed in on the search engine and will only be accessible to the user himself.

Basically, all signed in users of the prototype should have been able to perform any Create, Read, Update and Delete operations to all search results that are visible to them. There are a several reasons to provide users with such broad privileges:

- First of all, it is consistent with the research performed by Forte and Bruckman in which they claim that hard coded stratification of privileges should be kept to a minimum to keep people involved in a community.
- Second, it enables everyone to modify all publicly viewed search results. This also means that everyone is able to add additional search terms to a search result.

- Third, partially prohibiting modifications of all visible content of the user leads to a more sophisticated, non straight forward rule system that is harder to understand for users.
- Fourth, because the amount of users of the prototype is rather limited, malicious use of the system will not be very rewarding and is therefore not expected.

As mentioned before, during the development of the prototype, the Scrum development method was used. Work was planned in one week sprints, a backlog was maintained and the most important items from the backlog were taken up in the next sprint. Unfortunately, due to prioritisation and time restrictions, Update and Delete operations were not implemented in the prototype. Furthermore, no interface for the deep web was implemented and no sophisticated tagging methods were designed, also due to time restrictions.

To provide an overview of the system in a systematic and generally accepted format, a Use Case Diagram and a Components Diagram were created. Figure 8 shows a Use Case Diagram stating all the actions that a user should be able to do with respect to the prototype. Figure 9 gives an overview of the software architecture using a Components Diagram. Both diagrams make use of the Unified Modeling Language notation.

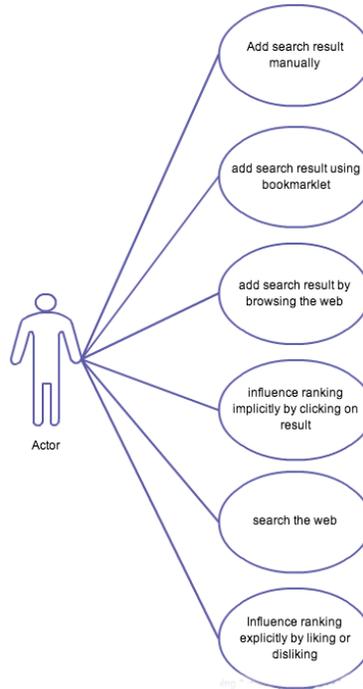


Figure 8: Use Case Diagram of the asynchronous Social Search Engine prototype.

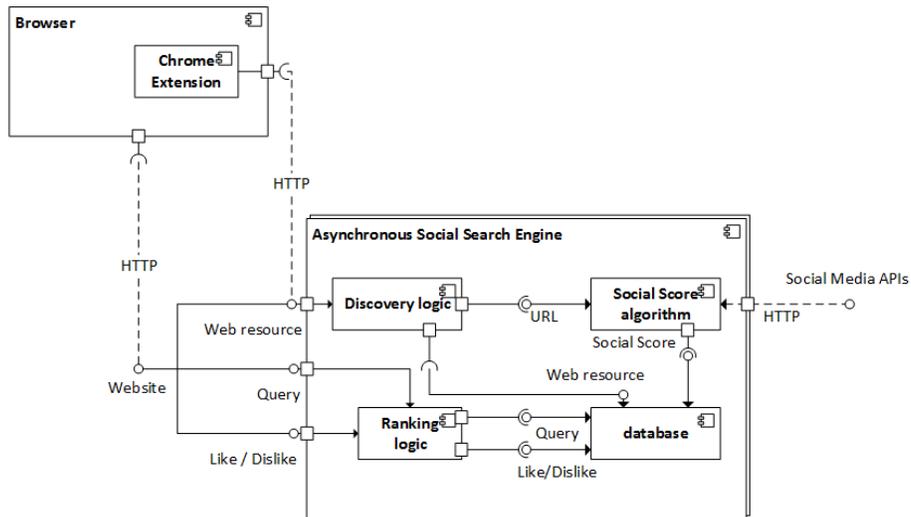


Figure 9: Components Diagram of the asynchronous Social Search Engine prototype.

5.5 Hard and Software requirements

Computer systems in general consist of both hard and software. In this Section we elaborate on the requirements regarding the hard and software for building an Asynchronous Social Search engine. Requirements of a Asynchronous Social Search engine heavily depend on the amount of users. First, we describe the requirements for the system as if it would have billions of users. Second, the requirements are described for the prototype that was built as part of this research. The prototype supports several hundreds of users.

Requirements for billions of users

In many ways the proposed system has similarities with other search engines such as Google and Bing. Therefore, hard and software requirements can be derived from those already existing search engines. Furthermore, an Asynchronous Social Search engine also has similarities with Social media websites such as Facebook and Twitter. One advantage is that the proposed system will only store text whereas Social media websites often also have to store pictures and videos which require significantly more storage space.

Brin and Page give rather detailed information about the initial system they setup (Brin & Page,1998). They indexed 24 million webpages and claimed that the amount of storage required to run their initial setup of the search engine is approximately 55 GB (Giga Bytes). As mentioned before, in 2012 Google claimed to have indexed 30 trillion webpages. Brin and Page also claim that their approach was very well scalable. Assuming linear growth of the amount of required storage space based on the amount of webpages, the storage requirements for 30 trillion webpages would be 69 PB (Peta Bytes). This is quite consistent with the claim made by Google in 2013 that their index size is over 100 PB. Assuming a price of \$100 for 2 TB (Tera Bytes), such a storage capacity would cost a total of \$5 million dollars. Of course a search engine does not only require storage but also other hardware. To be able to respond to queries and make calculations in general, a search engine would also need working memory. Google claims to answer approximately 100 billion queries a month which comes down to 38 thousand queries a second. We do not go into further detail of requirement estimations because it is beyond the scope of this research and the prototype that we built only requires a very small fraction of the computational power necessary to support the entire world with Asynchronous Social Search.

The given numbers are stunning high. However, there are arguments that support the claim that hardware requirements should be less of an issue nowadays than it was in 1998. Following Moore's law, processing power doubles every 18 months, which would mean that now, 15 year later, processing power should be approximately a 1000 times faster than it was in 1998 (G. E. Moore et al.,1965). The same also holds for storage space and many other hardware products. Consistently, the price per GB storage in 1998 was around \$85 whereas in 2013 one GB of storage would cost \$0.07 which is approximately a factor 1000 cheaper.

On the other hand, the web also became much and much bigger. Accordingly, the size of the index Google maintains has grown drastically as well. Measured in pages, the index size has grown by a factor 1.25 million. This suggests that indexing the web has become three orders of magnitude "harder" with respect to 15 years ago. We believe that the amount of pages in the index of a search engine is not the most important factor for quality anymore.

One similarity between an Asynchronous Social Search engine and social media websites such as Facebook and Twitter is that hardware requirements scale linearly with the amount of users of the system. This leads to a major advantage of an Asynchronous Social Search engine over a computer based search engine: it doesn't require the same hardware when there are only a few people using the system as when the entire world is using the system. Therefore investment costs are very low and the hardware can be scaled up at the moment that more people start using the system. This way, from a business point of view, risky investment costs can be avoided.

According to a study about search engine performance conducted by Michael, Moreira, Shiloach and Wisniewski it is better to increase performance by scaling out than to scaling up (Michael, Moreira, Shiloach & Wisniewski,2007). Scaling up means that a hardware unit is upgraded so that it gains additional computational power or storage capacity. Scaling out means that instead of upgrading hardware units, more hardware units are used and work is distributed among multiple, replaceable, hardware units that work together as one computer.

Regarding software requirements, when talking about huge quantities to process and store, it is always more efficient to use a compiler instead of an interpreter. Furthermore, algorithms should be written in such a way that they can run in $O(1)$ time as much as possible. When using a distributed hardware architecture it is important that algorithms can run in parallel using a programming model such as MapReduce (Dean & Ghemawat,2008).

Requirements for a prototype

For the prototype that we created, we used the LAMP software bundle. In this case, LAMP stands for Linux, Apache, MySQL and PHP. Using PHP is inconsistent with the statement made earlier about preferring compiled code over interpreted code. However, because this is a prototype that doesn't need to support millions of people it does not harm performance to use PHP. The main reason to choose for PHP is that it supports solutions that are fast and easy to put in place. The prototype runs with a storage capacity of 50GB and 1GB working memory. These specifications are rather similar to those of the initial setup of Google's first search engine.

6 Experiment: quality of the proposed search method

An experiment was performed to determine the quality of the proposed search method. Furthermore, the capabilities to discover resources on the web were analysed. Last, a case study was performed to determine the feasibility of the use of the search method in a business environment. The following Subsections elaborate on the approaches taken and the results found.

6.1 Ranking quality

An experiment was conducted to measure the quality of the proposed search method based on the method described in Section 2.4. An example of the interface is shown in Figure 3. The experiment ran from December 25th 2013 till January 7th 2014, a period of 14 days. To acquire a sufficient number of clicks, an online advertisement was placed for the search method's website. The advertisement was in English. When people clicked on the advertisement they were redirected to the Search Engine's homepage. In total, 4 180 unique visitors to the website were recorded. The advertisement was not targeted on a specific country, so it was shown all over the world. Most visits came from India, China and Pakistan. The three countries together accounted for just over 50 per cent of all visits. In total, 4 030 queries were posed to the search method. Notice that with instant search, every extra character typed into the search box is counted as a new query. 735 clicks were recorded. 560 clicks came from the initial page. The initial page shows the results for an empty query. In that case, the user did not actually pose a query to the search engine yet. However, instant search enables search methods to come up with results during query time, but also to already show the most relevant results before a query was even posed. As you can infer from Table 9, the social search method performed way better for the initial page than the full-text search, which makes perfect sense because full-text search is solely based on match between the text of the resources and the query. An empty string as a query fits equally well to every document and therefore the full-text search method can not make any proper estimation of the most relevant resources before a query is posed. Because this is an exceptional case and it makes up for a large amount of the clicks, we also show the results as they would be if you exclude the initial page.

To be able to analyse the results of the experiment, the sign test was used. The number of clicks on the social search method were compared to the number of clicks of a baseline method, namely the full-text search method F in combination with a naive instant search method I . Naive instant search method I worked by looking at substrings in all keywords related to resources. When multiple matches are found based on query Q , the search method took arbitrary choices in ranking them. So two search methods were compared, that both consisted of two search methods themselves. In both search methods, instant results took precedence over full text results. Table 8 gives an overview of the

	baseline method	social method
instant	I	SI
full text	F	SF

Table 8: Search methods overview. In the experiment, the baseline method is compared to the social method.

	baseline	social	p-value
initial page included	162	573	<0.0001
initial page excluded	73	102	0.0340

Table 9: Distribution of clicks and corresponding p-values in a two-tailed sign test

four search methods used. As a null hypothesis, it is assumed that results from the social search method and the baseline method are equally often clicked on. Conversely, the H1 hypothesis states that the number of clicks on the baseline method and the social method are not equally distributed. A two tailed sign test with a confidence level of 95% was used to determine significance of results. The distribution of clicks over the search methods is shown in Table 9 and also the corresponding p-values are indicated. In both cases, including and excluding the initial page the p-value is significant. Therefore, we can reject the H0 hypothesis and conclude that the social search method works better than the baseline search method. To get a better understanding of where clicks came from, Table 10 provides an overview of percentages of clicks generated from each search method. We can not simply conclude that the social instant search method is the best because it accounted for 42% of all clicks because it was only guaranteed for the baseline method and the social method to be shown to the user the same number times. It is very likely that social instant search results were shown to the user more often than social full text search results because the latter are only shown when there are not sufficient social instant search results given a query Q .

The Social Fulltext ranking method SF is influenced by the Social Score S as defined in Section 5.4. From the corpus a top 50 was assembled based on Social Score S . Figure 10 gives an overview of the most important websites worldwide according to the Social Score S . Notice that there were ten duplicates in the list, such as "https://twitter.com" and "http://twitter.com" which both refer to the same content. Such duplicates were removed from the list. In large extent the list feels intuitively right. The most disturbing about the list is that Wikipedia has been ranked only 31st. In a PageRank algorithm Wikipedia would probably score top five, but apparently Wikipedia is not a source that many people frequently share or like via Social Media compared to the number of back links created to Wikipedia by authors on the web. Notice that theoretically there could be sources missing that have never been indexed by the search engine. That would be rather unlikely though, because indexing is based on

	baseline	social
instant	13%	42%
full text	29%	16%
total	42%	58%

Table 10: Distribution of clicks over search methods in percentages. The initial page clicks are excluded.

visits and you would expect that the most popular webpages on the web would have been visited at least once during this study by one or more users. It could be the case that there is a website in a large country or continent of which no users had installed the Chrome extension. That would most likely be South America according demographic usage statistics of the performed experiment. Obviously, the ranking presented in Figure 10 changes over time. With every like, share or other form of Social Media interaction with respect to a web resource, the ranking of the resource changes. It was outside the scope of this research to determine how often the Social Score should be updated.

#	URL	Social Score
1	http://www.google.com/	5,90
2	http://www.facebook.com/	5,73
3	https://twitter.com/	5,50
4	http://www.youtube.com/	5,33
5	http://www.flickr.com/	5,14
6	http://www.amazon.com/	5,10
7	http://espn.go.com/	5,07
8	http://www.ted.com/	4,97
9	http://grooveshark.com/	4,94
10	http://www.pandora.com/	4,88
11	http://www.nytimes.com/	4,85
12	http://www.yahoo.com/	4,85
13	http://9gag.com/	4,79
14	http://www.ebay.com/	4,75
15	http://www.etsy.com/	4,74
16	http://www.apple.com/	4,70
17	http://www.imdb.com/	4,68
18	http://www.youtube.com/watch?v=9bZkp7q19f0	4,58
19	http://maps.google.com/	4,57
20	http://www.pinterest.com/	4,50
21	http://mashable.com/	4,49
22	http://www.nationalgeographic.com/	4,49
23	http://www.time.com/time/	4,48
24	http://www.linkedin.com/	4,40
25	http://www.rollingstone.com/	4,40
26	http://www.speedtest.net/	4,40
27	http://www.mtv.com/	4,39
28	http://www.codecademy.com/	4,35
29	http://www.kickstarter.com/	4,29
30	http://www.wix.com/	4,28
31	http://www.wikipedia.org/	4,26
32	http://www.fcbarcelona.com/	4,26
33	http://www.youtube.com/watch?v=jofNR_WkoCE	4,25
34	http://dictionary.reference.com/	4,25
35	http://translate.google.com/	4,25
36	http://www.indeed.com/	4,25
37	http://www.ign.com/	4,25
38	http://instagram.com/	4,21
39	http://www.asos.com/	4,21
40	http://digg.com/	4,19
41	http://www.last.fm/	4,18
42	http://www.stereomood.com/	4,17
43	http://imgur.com/	4,17
44	http://thenextweb.com/	4,16
45	http://www.picmonkey.com/	4,15
46	http://edition.cnn.com/	4,15
47	http://www.apple.com/iphone/	4,14
48	https://mail.google.com/mail/	4,14
49	http://weavesilk.com/	4,13
50	http://www.weather.com/	4,12

Figure 10: Top 50 URLs worldwide according to Social Score S on march 27th 2014.

6.2 Discovery capabilities

To be able to measure the discovery capabilities of asynchronous Social Search methods, the prototype its corpus was compared to the corpus of Google. The prototype was named Kiwi, so the term prototype and the name Kiwi are used interchangeable from now on. In absolute size, the acquired corpus consisted of over a 120 000 resources. That is approximately seven orders of magnitude smaller than the absolute corpus size of Google. However, based on a random sample of 400 resources taken from our corpus, it was found that 41 per cent of the resources indexed by Kiwi, were not indexed by Google. This is consistent with research performed by Heymann, Koutrika, Garcia-Molina, who found that 25 per cent of bookmarked resources on Delicious were not indexed by search engines (Heymann et al., 2008). That we found an even higher percentage could be explained by the fact that intranet websites were also automatically indexed by the search engine. One thing that also has to be taken into account is the fact that we did not perform any checks regarding whether authors of pages explicitly indicated that pages should not be indexed by search engines. Although the corpus is relatively small, we do know that all pages in the index have been visited at least once. Therefore, we can argue that it is a limitation to a search engine if those pages are not indexed. 29 215 resources were acquired via the Delicious API. We also know that the rest of the index was generated mainly by the browse behaviour of just over 20 participants that installed the Chrome Extension. That means that every user of the extension roughly attributed 4 500 resources to the index during the experiment. An increase in the number of participants would definitely result in an increase of the corpus. The corpus will also grow over time, even when the number of users does not increase. The search method seems to be able to discover unique content that has never been indexed before, but with a limited number of users the corpus is small. An interesting question is how many web resources are actually useful to people and we believe that quality of resources is more important nowadays than quantity in digital search.

7 Case study: feasibility in a business environment

Sharing information and knowledge in a business environment has become an important topic for organisations. From a business point of view, many of their information resources should only be available internally. Reasons to keep access restricted to people from the organisation are often related to privacy and competitive advantages. By using intranets, the barriers to share knowledge have been lowered. However, direct access to the resources is only available to a small part of the organisation. In many cases, only specific business units or workgroups have access to information that would also be valuable to many others in the organisation. In such an environment information is stored in so called data silos. One major problem that leads to inaccessibility of resources is information overload. The information infrastructure is often complicated to understand and the amount of knowledge available is way too much. The right information at the right time becomes as good as invisible as a needle in a haystack (Offsey,1997). There is a need for a solution which doesn't require extensive change to the information systems of the organisation. According to Accenture, one of the first steps that should be taken is to provide the people of the organisation with a Single Point of Access (Nanterme & Daugherty,2014).

An organisation was selected that is active in the Human Resource Management (e-HRM) and automation of wage and salary administration business with approximately one thousand employees. The organisation is mainly based in the Netherlands and possesses one of the largest development teams in the Netherlands. The schedule shown in Table 11 was followed during the case study.

The proposed search method was slightly modified for the case study. First of all, the search engine was hosted internally at the organisation. Second, only a preselected list of domains was indexed by the Chrome extension that were deemed to be relevant for the daily work. This list of domains that were automatically indexed using the Chrome extension was composed based on input from participants that actually installed the Chrome extension. Furthermore, the bookmarklet for adding results and the personal account feature were not part of this case study. No resources were acquired from Delicious, as was the case in the online public experiment. In total, 22 domains were indexed by the Chrome extension, of which some cannot be shown for reasons of confidentiality. A few that are not confidential are

- linkedin.com
- microsoft.com
- stackoverflow.com

The case study ran for 25 workdays from the 10th of february onwards. In total, 16 employees participated in the case study. All 16 installed the Chrome Extension and used the search method for a period of four to five weeks. Participation

Activity	Time period
Prepare and deploy prototype	Feb 3 - Feb 7
Convince people to join the project	Feb 10 - Feb 14
Quantitative Data collection	Feb 10 - March 14
Interviews with participants	March 3 - March 14
Report on results	March 17 - March 21

Table 11: Time schedule for the case study.

was voluntary, which is a relevant factor according to the TAM2 model. According to TAM2, subjective norm should not have any influence on Intention to Use if usage of a system is on voluntary basis (Venkatesh & Davis,2000). In the interviews conducted in the fifth week of the case study, all other factors that can influence the Perceived Usefulness of a system according to TAM2 were asked about implicitly and sometimes explicitly. From the 16 participants, 15 were interviewed. The last one was on holidays. Table 12 gives an overview of the number of people per function in the participants group. In total, 22 unique visitors were identified and a total of 232 visits to the website were recorded with a total of 641 page views. Figure 11 gives an overview of the direct influencing factors from the TAM2 model on Perceived Usefulness and the actual influence the factors had on the Perceived usefulness based on the interviews held with participants. Appendix A gives an overview of the questions that were asked during the semi structured interviews with participants. Transcriptions of parts of the interviews in which the answers were given to those questions can be found in Appendix B. Notice that interviews were held in Dutch and the Appendices are translated to English.

Perceived Ease of Use

Most people found the interface clear, simplistic, easy to understand and all except for one participant was happy with the performance of the system. For example, the Product owner said about the interface:

”It is clear, light and simple. It doesn’t take any effort to add results”.

People were able to access to prototype rapidly because they had stored the link in their bookmarks, used it as their default search method from the omnibox or set it up as their default page for opening new browser pages and page tabs. There were however, people that did not use Chrome as their default browser and this group of people had to take more effort to access and use the prototype because it was only properly functioning in the Chrome browser. There were also some people who would have liked some more explaining text in the interface to get them started. The majority however, had a clear understanding of the way the interface was meant to be used. There were also two persons who argued that it is not user friendly that you first have to visit a page before it

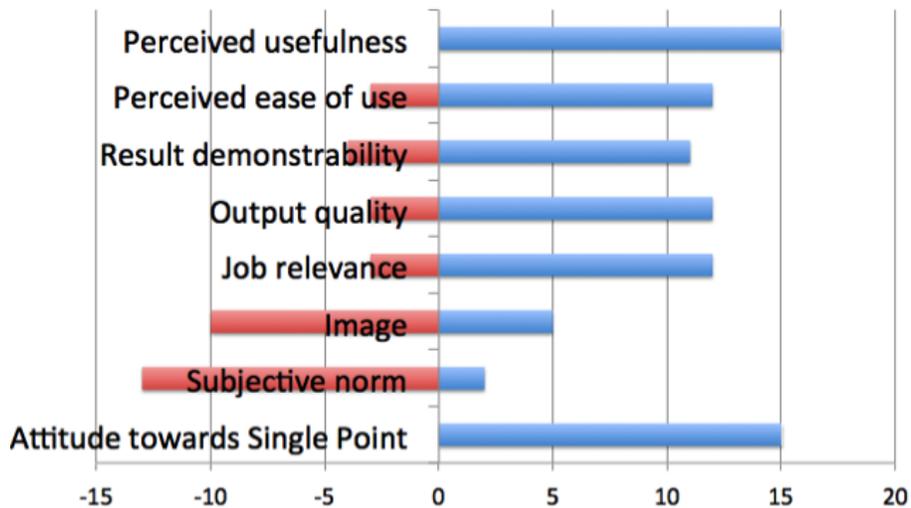


Figure 11: Direct influencing factors on Perceived Usefulness in the TAM2 model. The red horizontal bars indicate the number of people that experienced the factor as having a negative influence on Perceived Usefulness whereas the blue bars indicate the number of people that experienced the factor as having a positive influence on Perceived Usefulness.

is indexed by the search engine. They, and a few others, would have liked to have an additional sort of crawler in place that would index pages connected to visited pages automatically. There were also a few participants who said that adding results manually should have been easier. Most of the people were able to retrieve the documents they added either manually or implicitly by visiting pages using the Chrome extension without any trouble. People also indicated that it was rather easy to modify the ranking of results. There were a few people who found that it was too easy for people to manipulate the rankings on their own. They didn't like the fact that likes could be assigned an indefinite number of times and not only once for one person. People also indicated that there should be a quick way to pose the same query to other search engines such as Google. A button should be provided which would instantly pose the same query to Google. Or, even better, Google results should be integrated into the interface in such a way that they come below the results from the prototype as it works right now. Several also indicated that it could be useful to show how a score for a resource is composed. Did it get 65 likes because one person liked it up on its own? Or did 40 different people click on the link and 25 people explicitly assigned a like to the resource? One of the participants that came up with this suggestion also noted that not too many changes should be made to system as is, because things would only become more complex.

Function	Number of participants
Software Engineer	5
Support Consultant	2
Development Manager	2
User Experience Designer	2
Product Owner	1
Test Engineer	1
Application Specialist	1
Business Intelligence Specialist	1
Chief Information Officer	1

Table 12: Number of people per function in the participants group of the case study

Result Demonstrability

Most people knew both how results could be added and how they were then ranked during querying. They correctly indicated that ranking is based mainly on likes and results can be added manually and using the Chrome extension. As Software Engineer 5 phrased it:

”What I see is that the number of likes is key in ranking results”

. One participant also found out about the bookmarklet, although this wasn’t the intention during the case study. It turned out however, that this participant was very enthusiastic about the bookmarklet. It was not always clear to people how likes could be assigned to search results. Most participants knew you could add them manually. Some indicated that likes were also generated implicitly based on page visits or clicks. Notice that the number of visits did not actually play a role in ranking although this was presumed by some participants. An interesting note is that only one person mentioned the fact that keywords are also used when ranking results. The others probably took this for granted, because they all knew they could pose queries to the system.

Output Quality

Participants were asked whether they were satisfied with the results they got from the system, what percentage of queries lead to success and whether they could find back recourses they added to they system both implicitly and explicitly. On average, approximately one in three queries returned good results. Nearly all people were satisfied with their success ratio on queries. The participating Application Specialist stated:

”One in four, and this ratio increased. I was satisfied with one in four. You know that it is something that is being built up and more information is becoming available all the time”.

Some people indicated that there was a big difference in success ratio between intranet and internet search. Many indicated that the ratio differed with what you were looking for. The majority claimed that the tool was much more suitable for intranet search than for internet search. As Software Engineer 4 indicated:

”(The success ratio) depends on what you are looking for. Documentation like sources and intranet sources (high success ratio), but no technical sources like Stackoverflow (low success ratio)”.

Quite some people indicated that where search tools of internal systems failed, the prototype was a good alternative. Such internal systems included a Wiki, Sharepoint and Team Foundation Server. Not everyone tried to retrieve pages they added, but most of the people who added pages could find them back easily within the prototype. In general, we can conclude that people were rather satisfied with the output quality of the system.

Job Relevance

Virtually all employees indicated that the information available through the prototype was relevant for their daily work. As the Test Engineer indicated:

”You get ideas by using the system about resources you might not have known about before”.

Mainly the internally hosted resources on the intranet were found useful by most participants. Search is a task that all participants had to do regularly and mainly for the intranet they found that the support of the prototype was good. For external sources such as Stackoverflow however, the support was often not good because not all resources were indexed by the system. Several people also indicated that they should have asked to index a few more domains with the Chrome extension automatically that were relevant to their work. People were also asked whether the tool made them do their work more efficient and effective. Most people thought it would, although there were also quite a few who said that some improvements needed to be made before this would actually be the case. Suggestions came in to support search on group level and individual level, taking only a small group of people into account, e.g., your own department. A side note to this suggestions was that tunnel vision should be prevented and you should not lose all access to information outside your own group. That would ruin the inspiring aspect about the prototype. People also asked whether sources hosted on disk could also be indexed and other document types such as PDF format. Another person indicated that it would be useful to work with tags to be able to distinguish information topics. One participant stated that the current prototype could only search in unstructured data and that it would be very useful to him if also structured data sources could be queried in natural language. He compared this feature to Microsoft Enquiry.

Image

Most participants indicated that they didn't know what their colleagues were thinking about the system or even knew that their colleagues did not know about the system. Software Engineer 3 for example, said:

"Didn't hear my colleagues about it and my manager doesn't know about it".

There were some participants that talked about the prototype with each other, all in a positive way. As the CIO stated:

"Didn't have the time to share with my colleagues, one colleague found it a nice idea".

In general, we conclude that Image had a negative influence on Perceived Usefulness, mainly because not much was known about it in the organisation outside the participants group. Using the system would therefore not enhance one's status within the organisation.

Subjective Norm

Participants were using the system for at least four weeks, but many of them did not use the system more than a few times a week. In general, we can say that people did not have much experience with the system and therefore, Subjective Norm does have an influence on Perceived Usefulness in our case study. The vast majority of participants did not have the feeling that their supervisors and colleagues were expecting them to make use of the system. Therefore, Subjective Norm had a direct negative influence on Perceived Usefulness. Furthermore, according to the TAM2 model, Subjective Norm has a positive correlation with Image. Therefore, in our case study, Subjective Norm has a negative influence on Image and thereby also an additional indirect negative influence on Perceived Usefulness.

Quantitative data

As mentioned before, quantitative data was used to validate the qualitative data. The number of clicks, queries and the size of the index was recorded over time. Figure 12 gives an overview of the number of clicks recorded per week. It also shows the number of unique clicks per week. The number of unique clicks is defined as the number of queries in which one or more clicks were registered during a visit. The total number of clicks shows a bit of a random pattern, although you see a clear decline in the last two weeks. The number of unique clicks shows a more stable pattern, also decreasing slightly over time. Figure 13 gives an overview of the number of recorded queries per week, where each additional keystroke is recorded as a different query because of the instant search feature. Also the number of unique queries per week is shown. Here, the number of unique queries is defined as the number of unique queries posed to

the system in one visit. In this case the number of unique queries also shows a more stable, slightly decreasing pattern. Last, Figure 14 gives an overview of the total index size at the start of every week. Notice that the index size at the start of the experiment was 40, because there was already one participant who started using the system one day before the start of the measuring period for the case study. During the third week of the experiment there was a decline in index growth. This can be explained by the fact that at the end of the second week, an update of the Chrome Extension was released. This had to do with the fact that additional domains were requested to be indexed. Unexpectedly, this led to Chrome automatically disabling the extension until the user explicitly gave permission that this additional domain could also be indexed. Therefore, many people had the extension disabled during the third week. The growth function of the index seems to be rather linear, but there is noticeable decrease in growth already indicating logarithmic growth. This can be explained as follows. The majority of relevant resources has not yet been indexed by the system. Because slightly fewer resources are being indexed towards the end of the case study we can infer that people are also revisiting already indexed resources resulting in a decrease of index growth. This holds only by the assumption that participants did not change their frequency in browse behaviour using Chrome. In total, 3034 attempts were made to add a resource to the prototype, either using the Chrome extension or manually. Based on the interviews, we assume that over 99 per cent of attempts to add resources were made without any user effort, so using the Chrome extension. This means that approximately 3000 pages were visited by the 16 participants within the 22 domains that were allowed to be automatically indexed. Because the final index contained 1238 unique URLs, on average, every page in the index was visited over 2.4 times by the participants. Although we did not measure the number of attempted additions over time, we expect that this factor increased over time since, at the start of the case study, the index was very small and the chance to visit new resources using the Chrome was bigger than near the end of the experiment. The quantitative data indicates a slight decrease of use of the system over time. That could be explained as that people did not like the system. However, this would be inconsistent with qualitative data acquired via the interviews in which most people stated that the system could be of great help. Another explanation would be that people forgot about the system because they were too busy and the prototype was not an integral part of their daily work. This is also what we found during the interviews with participants.

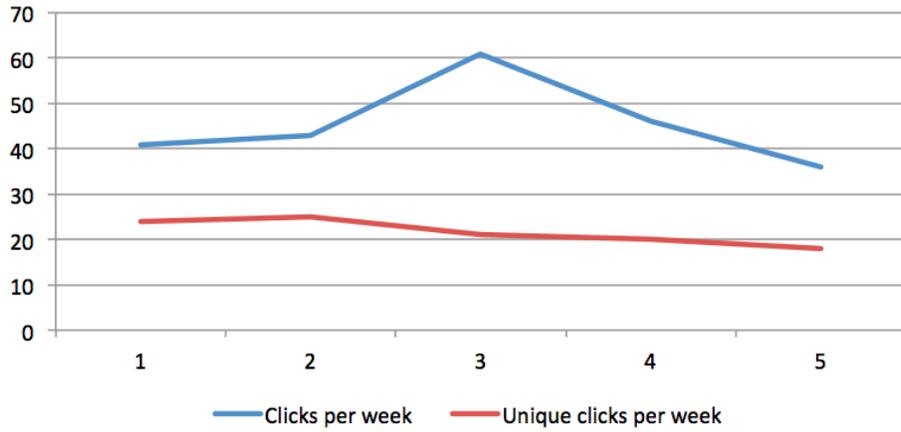


Figure 12: Number of recorded clicks per week

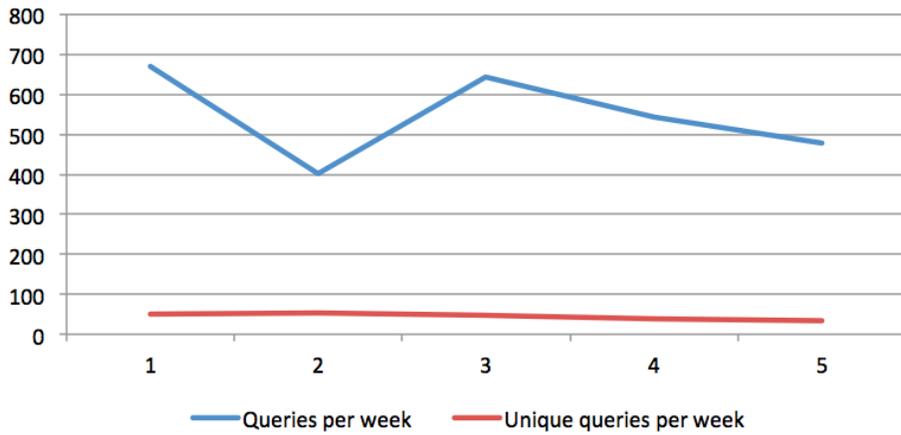


Figure 13: Number of recorded queries per week, where every additional keystroke during query typing is considered to be an additional query due to the instant search feature.

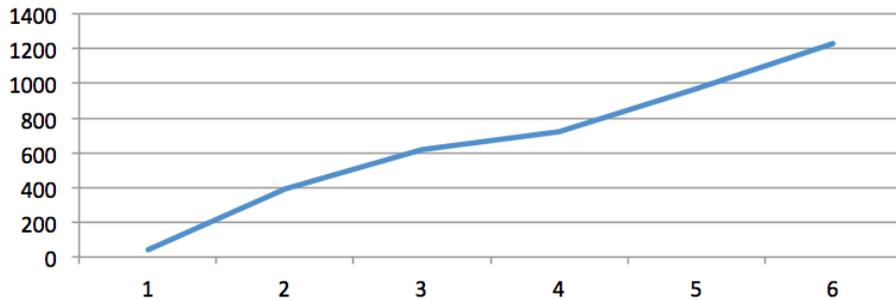


Figure 14: Size of the index measured at the start of every week.

What we learned

We believe that some essential changes need to be made to the prototype to make it really useful to people in supporting their daily work. All participants we interviewed indicated that there was definitely a need for a Single Point of Access through which all information related to the organisation could be accessed. All of them saw the prototype as a very good starting point for such a Single Point of Access. Many of the participants had additional feature requests for and remarks about the system, the most important ones being:

- **Browser independency.** Quite some participants made use of multiple browsers. The system should be made available in all browsers to make it easier to access the system.
- **The like system should be improved to prevent malicious use.** The ranking system with the likes should be made less prone to malicious use. Now people can assign an indefinite number of likes to a resource for a certain query. There should be a limit in the influence that one person can have on the ranking of a document. It could also be useful for people to see what a score of a page is composed of.
- **Trending topics.** It would be useful to know what colleagues are looking at a lot, also in the short term. Therefore, a list of trending topics and recently added pages that get many views could be very useful to employees. So next to the ranking based on all likes, there should also be a ranking based on recent likes.
- **Manually adding results should be made easier.** It takes too much effort to add results manually and the process is not always straightforward. Introducing the bookmarklet would be a great solution to this problem. Looking back at the case study, it was a mistake not to provide the participants with the bookmarklet.
- **Dynamically identify relevant domains.** The Chrome extension made use of a predefined set of domain names that were indexed such that

irrelevant domain with respect to work the organisations were not indexed automatically. This list should be easier to modify or even be changed dynamically based on Artificial Intelligence. Now people had come to us to ask for additional domains to be added to the list, we had to perform an update of the Extension and then people had to accept the new rights required due to the update. In short, an inefficient process.

- **internet vs intranet search.** The prototype indexed both internal and external sources. Most participants stated that the prototype worked best for intranet search and worse for internet search. To keep the goal of the system clear, it could be smart to exclude external internet resources. This way, the goal and scope of the system would be clearer: enterprise search.
- **Personalisation and groups.** Some people described the system as a sort of shared bookmarks. They would like to also have an overview only their own bookmarks. They would also like to have sort of intermediate level between organisation wide ranking and personal ranking. Employees should be able to join groups, like their own department or their own function. Such groups of people should then be able to have their own set of resources. This could also involve tagging of resources with group names.
- **Fallback search methods.** It can be frustrating to search the prototype, mainly when searching on the web because not many resources are indexed compared to the size of the web. Therefore, there should be a fallback search method or several fallback search methods such that if the number of results provided by the system is limited, the user can, without much effort, pose the query to another search method or this should even be done automatically. This could be achieved with a button for each search method, or more sophisticated by appending for example Google Search results to the list of results. Another option would be to use an automatic redirect of the browser when there are no results found.
- **Additional information.** An info page explaining how the system works, what the goal of the system is and how it works would be useful according to several participants.

8 Conclusions and discussion

Critical Success Factors that hold specifically for asynchronous Social Search methods are user involvement, implicit user feedback and explicit user feedback. It is not necessary to get the entire world involved. Just like the case of Wikipedia, only a small number of contributors can make the search method work for the rest of the world. Getting and keeping users involved can be achieved by maintaining a cycle of credit in which incentives are allocated in the form of power, resources and the ability to reinvest resources to be able to acquire more and more credibility. Furthermore, hardcoded, stratified privileges should be kept to a minimum such that people are given as much freedom as possible within the system. Implicit feedback such as click-through data and systems like the proposed Chrome-extension are also essential to an asynchronous Social Search method to support the Hit and Like economy. Implicit feedback can perform the vast majority of the effort required by users to add, tag and rate resources.

Asynchronous Social Search is able to discover links that other search methods would never be capable of. In other words, the method is able to discover links from the deep web. Ranking of results works better than full-text search in combination with naive instant search, but there are Search Engines that also perform better than basic full-text search already. The prototype would not have been a match against existing Search Engines like Google and Bing because fewer than 150 thousand links were indexed and there were not enough people actively involved. There might however be great potential for asynchronous Social Search when more people are actively involved.

Consistently with research performed by Accenture, we found that there is a need for a Single Point of Access within organisations (Nanterme & Daugherty,2014). Based on the performed case study we conclude that the proposed search method is a very good starting point for such a Single Point of Access. The alternative way of discovering and indexing webpages has the advantage that integration with the existing software architecture is not necessary. Therefore, the search method can be implemented within an organisation faster and cheaper than existing Enterprise Search Engines.

The proposed concept of a Social Score for every web resource based on existing interaction on social media such as likes and shares is an interesting topic on its own. We conclude that the proposed Social Score S is a good alternative for the more computational expensive PageRank method. It could be a complementary property to take into account in existing search methods. The Social Score could also be used to compare the presence of organisations behind webpages on Social Media. One could for example, want to determine the presence on Social Media of Coca Cola versus Pepsi, or the presence of Apple versus Microsoft.

By making use of the Social Score and asynchronous Social Search in general, a shift can be made towards the Like Economy, away from the Link Economy. This way, all internet users will gain more direct influence on the ranking of results. It would indicate a shift from an Aristocracy in which the power is

in the hands of the technically skilled web authors and developers to a direct democracy for Web Search.

Future research could be conducted on asynchronous Social Search with more users involved. Then, the search method could be compared to major Search Engines like Google and Bing. It could also be investigated into what extent asynchronous Social Search is feasible for personalisation of search results. The prototype used in this study did not take into account a Social Graph, the social relationships between users. More research could be performed on how a Social Graph could contribute to asynchronous Social Search. It could also be very useful to take into account the actual number of hits a web resource receives. In the prototype, only the number of clicks to a page via the Search Engine was taken into account, but using an application such as the Chrome Extension, one could also count the number of visits to a resource and take that into account in the Hit economy.

More research should be performed to determine the quantity of likes compared to the quantity of links available on the web. Also the quality of Social Signals with respect to links could be compared in an experiment with a double blind test in which two identical Search Engines are used except one is using PageRank and the other is using the Social Score. When a user poses a query, both search methods are queried and the results are mixed as described by Joachim (Joachims,2002). Based on which Search Engine receives the most number of clicks it can be determined which search method is performing better. It would also be interesting to experiment with the effects of taking into account different numbers of Social Signals from different numbers of Social Media platforms. It could be investigated whether Facebook Signals are from higher quality than Twitter Signals and whether bias can actually be removed by taking into account more Social Signals and Social Media platforms.

Also, the proneness of the Social Score and asynchronous Social Search in general to malicious use and spam should be evaluated. Would it be easier or harder to influence the ranking of results when Likes are used instead of Links to determine the value of webpages? Research could be performed to investigate how such malicious use of the Social Score could be prevented effectively. One direction for a solution might be to use many Social Media Platforms to force spammers to spam many different systems. One advantage of the Social Score is that spam and malicious use is not only the problem of the Search Engine, but also a direct problem for the Social Media platforms at hand.

References

- Aizawa, A. (2003). An information-theoretic perspective of tf-idf measures. *Information Processing & Management*, 39(1), 45–65.
- Ajzen, I. (1991). The theory of planned behavior. *Organizational behavior and human decision processes*, 50(2), 179–211.
- Bagozzi, R. P. (2007). The legacy of the technology acceptance model and a

- proposal for a paradigm shift. *Journal of the association for information systems*, 8(4).
- Berners-Lee, T., Hendler, J., Lassila, O. et al. (2001). The semantic web. *Scientific american*, 284(5), 28–37.
- Bilenko, M. & White, R. W. (2008). Mining the search trails of surfing crowds: identifying relevant websites from user activity. In *Proceedings of the 17th international conference on world wide web* (pp. 51–60).
- Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1), 107–117.
- Chung, L., Nixon, B., Yu, E. & Mylopoulos, J. (2000). Non-functional requirements. *Software Engineering*.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319–340.
- Dean, J. & Ghemawat, S. (2008). Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1), 107–113.
- D’souza, D. & Wills, A. C. (1998). *Objects, components, and frameworks with uml: The catalysis(sm) approach*. Addison-Wesley Professional Reading.
- Evans, B. M. & Chi, E. H. (2008). Towards a model of understanding social search. In *Proceedings of the 2008 acm conference on computer supported cooperative work* (pp. 485–494).
- Evans, B. M. & Chi, E. H. (2010). An elaborated model of social search. *Information processing & management*, 46(6), 656–678.
- Evans, B. M., Kairam, S. & Pirolli, P. (2010). Do your friends make you smarter?: An analysis of social strategies in online information seeking. *Information Processing & Management*, 46(6), 679–692.
- Evans, M. P. (2007). Analysing google rankings through search engine optimization data. *Internet research*, 17(1), 21–37.
- FAROO (Tech. Rep.). (2007).
- Forte, A. & Bruckman, A. (2005). Why do people write for wikipedia? incentives to contribute to open-content publishing. *Proc. of GROUP*, 5, 6–9.
- Gerlitz, C. & Helmond, A. (2013). The like economy: Social buttons and the data-intensive web. *New Media & Society*.
- Golder, S. A. & Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of information science*, 32(2), 198–208.
- Golovchinsky, G., Pickens, J. & Back, M. (2009). A taxonomy of collaboration in online information seeking. *arXiv preprint arXiv:0908.0704*.
- Gordon, I. (1989). *Beat the competition: How to use competitive intelligence to develop winning business strategies*. Basil Blackwell.
- Gupta, M., Li, R., Yin, Z. & Han, J. (2010). Survey on social tagging techniques. *ACM SIGKDD Explorations Newsletter*, 12(1), 58–72.
- He, B., Patel, M., Zhang, Z. & Chang, K. C.-C. (2007). Accessing the deep web. *Communications of the ACM*, 50(5), 94–101.
- Hecht, B., Teevan, J., Morris, M. R. & Liebling, D. J. (2012). Searchbuddies: Bringing search engines into the conversation. *ICWSM*, 12, 138–145.
- Hendler, J. (2009). Web 3.0 emerging. *Computer*, 42(1), 111–113.

- Hevner, A. R., March, S. T., Park, J. & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 28(1), 75–105.
- Heymann, P., Koutrika, G. & Garcia-Molina, H. (2008). Can social bookmarking improve web search? In *Proceedings of the international conference on web search and web data mining* (pp. 195–206).
- Hinchcliffe, D. (2008). An executive guide to mashups in the enterprise. *JackBe Whitepaper*.
- Horowitz, D. & Kamvar, S. D. (2010). The anatomy of a large-scale social search engine. In *Proceedings of the 19th international conference on world wide web* (pp. 431–440).
- Joachims, T. (2002). Unbiased evaluation of retrieval quality using clickthrough data. In *Sigir workshop on mathematical/formal methods in information retrieval* (Vol. 354).
- Keith, C. (2010). *Agile game development with scrum*. Pearson Education.
- Latour, B. & Woolgar, S. (1979). *Laboratory life: The social construction of scientific facts*. Princeton University Press.
- Manning, C. D., Raghavan, P. & Schütze, H. (2008). *Introduction to information retrieval* (Vol. 1). Cambridge University Press Cambridge.
- Matthijs, N. & Radlinski, F. (2011). Personalizing web search using long term browsing history. In *Proceedings of the fourth acm international conference on web search and data mining* (pp. 25–34).
- Michael, M., Moreira, J. E., Shiloach, D. & Wisniewski, R. W. (2007). Scale-up x scale-out: A case study using nutch/lucene. In *Parallel and distributed processing symposium, 2007. ipdps 2007. ieee international* (pp. 1–8).
- Moore, G. C. & Benbasat, I. (1991). Development of an instrument to measure the perceptions of adopting an information technology innovation. *Information systems research*, 2(3), 192–222.
- Moore, G. E. et al. (1965). *Cramming more components onto integrated circuits*. McGraw-Hill.
- Morris, M. R. (2008). A survey of collaborative web search practices. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 1657–1660).
- Morris, M. R. (2013). Collaborative search revisited. In *Proceedings of the 2013 conference on computer supported cooperative work* (pp. 1181–1192).
- Morris, M. R. & Horvitz, E. (2007). Searchtogether: an interface for collaborative web search. In *Proceedings of the 20th annual acm symposium on user interface software and technology* (pp. 3–12).
- Nanterme, P. & Daugherty, P. (2014, January). *From digitally disrupted to digital disrupter* (Tech. Rep.). Technology Labs, Accenture.
- Noll, M. G. & Meinel, C. (2007). Web search personalization via social bookmarking and tagging. In *The semantic web* (pp. 367–380). Springer.
- Offsey, S. (1997). Knowledge management: linking people to knowledge for bottom line results. *Journal of Knowledge Management*, 1(2), 113–122.
- Ortega, F., Gonzalez-Barahona, J. M. & Robles, G. (2008). On the inequality of contributions to wikipedia. In *Hawaii international conference on system sciences, proceedings of the 41st annual* (pp. 304–304).

- Pariser, E. (2011). *The filter bubble: What the internet is hiding from you*. Penguin UK.
- Rajaraman, A. & Ullman, J. D. (2012). *Mining of massive datasets*. Cambridge University Press.
- Robey, D. & Farrow, D. (1982). User involvement in information system development: A conflict model and empirical test. *Management Science*, *28*(1), 73–85.
- Russell, S. (2003). *Artificial intelligence: A modern approach, 2/e*. Pearson Education India.
- Schwaber, K. & Beedle, M. (2002). *Agile software development with scrum* (Vol. 1). Prentice Hall Upper Saddle River.
- Sugiyama, K., Hatano, K. & Yoshikawa, M. (2004). Adaptive web search based on user profile constructed without any effort from users. In *Proceedings of the 13th international conference on world wide web* (pp. 675–684).
- Van Raaij, E. M. & Schepers, J. J. (2008). The acceptance and use of a virtual learning environment in china. *Computers & Education*, *50*(3), 838–852.
- Venkatesh, V. & Davis, F. D. (2000). A theoretical extension of the technology acceptance model: four longitudinal field studies. *Management science*, *46*(2), 186–204.
- Venkatesh, V., Morris, M. G., Davis, G. B. & Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS quarterly*, *27*(3).
- Wu, B. & Davison, B. D. (2005). Identifying link farm spam pages. In *Special interest tracks and posters of the 14th international conference on world wide web* (pp. 820–829).
- Yanbe, Y., Jatowt, A., Nakamura, S. & Tanaka, K. (2007). Can social bookmarking enhance search in the web? In *Proceedings of the 7th acm/ieee-cs joint conference on digital libraries* (pp. 107–116).
- Yin, R. K. (2009). *Case study research: Design and methods* (Vol. 5). Sage.

A Interview questions

0. Do you have a need for a Single Point of Access within the organisation?

Experience with the system

1. Do you know what Kiwi is? (Goal?)
2. Do you use Kiwi regularly? (Times per day/week)

Subjective norm

3. What is the opinion of your 1) colleagues and 2) managers about Kiwi?
4. Do you colleagues and managers expect you to use Kiwi?

Job relevance

5. Is the information available through Kiwi relevant to your work?
6. Do you think that the use of Kiwi makes you perform your work more efficient and effective? (If no, do you see possibilities for changes to the system to achieve this?)

Output quality

7. How often did your searches via Kiwi succeed? Were you satisfied about this?
8. If you added a result, either using the extension of manually, could you retrieve this result later on using Kiwi?

Result Demonstrability

9. Do you understand in which way documents are sorted on relevance if you pose a queries to the system?
10. Do you understand in which ways results can be added to the search engine?

Perceived ease of use

11. Did you know where to find and access Kiwi? Where and how (omnibox, bookmark, URL)? Did it take much effort to access the system?
12. Do you think Kiwi is user friendly?
 - a. *Was the interface clear and understandable?*
 - b. *Do you think Kiwi is fast enough?*
 - c. *How much effort does it require to add results to the search engine?*
 - d. *How much effort does it require to get the results high in the rankings that you think are the best results?*

Do you have any suggestions or feature requests with respect to Kiwi?

Perceived usefulness

13. Do you think that a system like Kiwi is valuable for you as an employee? And for the organisation? With and without changes to the current system?

B Interview answers

Test Engineer	
Q0	Yes, sounds like it will provide us with a clear overview of resources with respect to our work
Q1	Yes
Q2	daily use, I also use Internet Explorer next to Chrome, 5-10 searches a day
Q3	Didn't hear not much about it from them, but they are honest. I think: no news is good news. Not much is said about it by colleagues and I didnt speak about it with my bosses
Q4	They didn't say I had to use it. I do think they prefer I do use it. It is fun and we expect from each other that we use it.
Q5	yes
Q6	Yes, You get ideas by using the system about resources you might not have known about before. You could also think of different groups with different links on top. So personalisation on group level. But not tunnel vision should occur thanks to that. That would affect the inspiring thing about the tool
Q7	9 out of 10
Q8	Yes
Q9	By likes, and maybe also number of visits, not sure. It should be logical that visits should also affect ranking positively
Q10	Yes
Q11	Via Google, usually I make use of the bookmark in Chrome "favorites",. Always have a tab open in second screen
Q12	Yes, it can be optimised though. I can easily work with it. Interface is clear, search is fast enough, effort to add results can be reduced. Information about how to use "howto". Easy to modify ranking with likes
Suggestions?	Shouldn't be possible that someone likes his own document up in the rankings with 100 likes or so. Search terms should also be modifiable. Use of the system will optimise the system. More results per page, you should be able to navigate to the next pages as well with results. Also, I want to be able to modify the title and description of resources. For me and for others to be able to see those changes. Give people freedom with the system, don't limit them in rights (consistent with literature about social systems ze i k)
Q13	Yes, for both me and *****
Software Engineer 1	
Q0	yes
Q1	yes
Q2	10-25 times a day
Q3	Don't know
Q4	No
Q5	Yes, but unique questions with very specific information that you only use once is not available. For example, information from Stackoverflow that you only lookup once. The first time it will not be available and you won't use it a second time, when it would be available through Kiwi
Q6	Yes, specifically for systems that are used by the same people within the organisation and less for pages on the web that you only access once
Q7	1 out of 3
Q8	yes
Q9	Yes, clicks and likes
Q10	Yes
Q11	Yes, "kiwi.comprise.com", usually I search via the omnibar
Q12	Yes, fast enough, clear simplistic interface, low effort to add results and influence ranking
Suggestions?	There needs to be a button: "search this query on Google"
Q13	Yes, both for ***** and myself as employee

Software Engineer 2	
Q0	Yes, would be nice, definitely
Q1	Yes, partially. Goal is to search multiple system sources. System is meant like shared bookmarks alike system.
Q2	No, only used it once
Q3	Don't know, managers dont know anything about it
Q4	No
Q5	It could, based on what you are looking for. Documentation like sources and intranet sources. No technical sources like Stackoverflow.
Q6	Yes, definitely. Mainly for the intranet. Everybody puts content in different places within the intranet. Example of information systems are Sharepoint and TFS. Improvement would be to index sources on disk, PDFs etcetra
Q7	Never, partially because I didn't know the system well. I assumed that the + would add the entire domain to the system instead of only that URL. Not sure whether this would be better than how it works now though.
Q8	Not always, keywords I added manually didn't work when querying. Wasn't as straightforward as I thought
Q9	Based on clicks
Q10	Manually
Q11	"kiwi.comprise.com", added it as a startpage. As my browser starts, multiple tabs are opened including one with Kiwi
Q12	No, there should be tooltips, more explaining tekst. Maybe an explanation page. The "plus" gave me the idea that the entire domain would be crawled. Kiwi is fast, Adding results doesn't take much effort. Getting results high in the rankings taked 62 clicks. The principle is strange... Imagine a new page is added after 4 years, it would get very low in the ranking. Maybe you also need a second pane with the most recent links. This years relevant links could be very different from last year
Suggestions?	Way the system was introduced at *****, it was not useful to me. The way I thought it would work is different than as I now know. I expected that the intranet would be indexed but also the internet is indexed. The system is way better for indexing the intranet. Watch out for tunnelvision. Don't call the term "like" likes. Use a "score" like Stackoverflow does. Just a number. Also show how a score is put together. I can imagine that if the system has indexed many pages, you can get the same problems, like Sharepoint. I like the concept very much of Kiwi. Maybe personalisation could be useful, but I am not sure about that. It would be useful if people could add domains to a list that are automatically indexed by the extension. Would be nice to have personal bookmarks as well and should be indexed on content of the page. On Stackoverflow, you can add questions to favorites but I don't like that. If you would use Kiwi and index those pages on content you could easily search through those / your "favorites"
Q13	Yes, both for me and ***** if we can find our stuff quickly. If employees have to search 60% less, that saves money for *****.
Software Engineer 3	
Q0	Yes
Q1	Yes, search engine, for ***** and other sites. No idea what the goal is. Maybe central point of access, which is useful, for internal use
Q2	50 times a month
Q3	Didnt hear them about it. Manager doesnt know about it
Q4	No, not yet
Q5	Stackoverflow dump, .net develop site. Most things I Google. For *****, then I would use, *****net cao cant find, With Kiwi I can
Q6	No
Q7	5 out of 10, was satisfied yes
Q8	Didnt add any and then tried to retrieve
Q9	Didnt check
Q10	No didnt look into that. Didnt have time because of performance problems. Idea is great though
Q11	Yes, via shortcuts and default search engine. "kiwi.comprise" and added to favourites
Q12	Yes, clear and simple, yes fast, didnt add any, didnt add any to top. For the rest it's a fine tool
Suggestions?	If you search, click through to Google should be good.
Q13	Yes, if you make it part of *****net for example

Software Engineer 4	
Q0	Not specifically, I use Google for my software engineering use. Most things are on Internet, for internal systems it could be of value. Every situation is different in my work
Q1	Not 100%. I think the goal is that you search together. Likes result in the correct information on top
Q2	Used as standard search in browser (omnibox) but removed it. The suggestion about a button to directly pose the query to Google would be a solution. He would take the effort to use for other to become efficient
Q3	We didn't talk about it much. I think that if we provide each other with good search results that would be useful
Q4	Don't know, I think that they expect i will use it when it is implemented ***** wide
Q5	No, didnt take the effort to ask for additional sites to index
Q6	If more programming results are added, yes
Q7	Depends on the subject. 0% for work related, but for ***** 100%. Depends on what you are looking for. The percentage of success is good
Q8	I didn't add any pages, didn't really understand the process and didn't check how this works
Q9	Yes, likes
Q10	No
Q11	Omnibox and "kiwi.comprise.com"
Q12	Yes, interface clear, fast enough (didn't see it was slow), no results added, easy to like results up in the rankings
Suggestions?	The suggested button "search on Google" is a good idea
Q13	Yes, for both me and ***** as a company. Prevent double search, together we could search better. Both in my function and in other functions this could be useful
Product Owner	
Q0	Yes, would be very nice
Q1	Yes, the goal of the system is approximately an internal search engine for organisational data. I don't know whether this was actually the goal of the system but this how I am using it right now
Q2	2-3 times a day
Q3	Don't know, never asked them
Q4	Yes, my colleagues
Q5	Partially, all information from internal websites is perfectly fitted, external sources are not in line. Use is for internal sources
Q6	Yes, very nice system
Q7	As Google engine, almost never, for internal sources 9 out of 10.
Q8	Yes
Q9	No not really, but i suppose that likes and views from colleagues influence the ranking
Q10	Yes, via plugin and with the "add" button
Q11	Via the URL
Q12	Yes, interface clear light and simple, superfast, no effort to add results, modifying the ranking he doesnt know. Adding results manually could be made easier. If you are on a page add him like to favorites (like the bookmarklet)
Suggestions?	He thinks it is a perfect solution for searching in organisational information . Better then existing search methods for specific systems like the search function from Sharepoint. Active documents show up high in the rankings. He didn't expect the tool to be so effective. The server has to stay up, also when you leave. He is really happy with it he emphaises. It suprised me how well it worked. He thinks Kiwi as a system is a good niche. Very feasible in a business environment. It should be used next to Google, for internal sources.He uses Kiwi as standard search method for. We have a lot of information within ***** , but all information is spread over different systems and the biggest problem is to find the correct information. ***** would be willing to pay for a system like this. Everybody in ***** is looking for knowledge and we had projects to make knowledge better accessible but it never worked. This has the potential to work, because it requires no effort and is simple to deploy
Q13	Yes, for me: i can find information quicker, get triggered from information found by colleagues. And as Reat it is good to share knowledge. I really think this could be a useful tool. Ambitious project, but I think that in the business world this is a very useful tool. There needs to be some work done, but it is really useful. You convinced me, a while ago I didn't really see how this tool would work. Let's make up the business case!

Software Engineer 5	
Q0	Yes, I think that would be really useful. Information is very widespread withing *****. Sometimes you don't even know whether it even exists what you are looking for. Single point of access would surely help.
Q1	Yes, Goal is to test a concept whether all information in an organisation can be indexed to get all information available from one location.
Q2	Not really, and didn't search that much either. Also depends on the type of info you are looking for. Before Googling, first I go Kiwiing. Got useful results. Used 1-10 times a week
Q3	Don't know, we didn't talk about that much. Interesting project with a lot of possibilities. Concept is clear. There are possible improvements before you would use it on a daily basis.
Q4	Boss didnt talk about it
Q5	No
Q6	I think so, if it is not, which domains are added to the plugin, so that is not the concept but the configuration of the extension.
Q7	In this stadium, not yet. But again, I think there is potention. From Kiwi, there should be a direct link to Google as an alternative. Or even using iFrames. Combine with Google. On top, the Kiwi results and beneath the results from Google. Mainly the link with a Search engine should be an important addition. The like system needs to be changed. Who has liked what could help, or make it more complex, one like per person, who are your friends, likes of friends count more for your results, or the people in your team or department. That would lead to a more complex system. More research could be done on this.
Q8	1 in 5. Satisfied, expected nothing and got something. Best results were on the Comprise Wiki. The Wiki search fails, but if I think about it now, Sharepoint could also be nice and other internal systems that are not publicly available so you cant use Google. That would be valuable to have a Kiwi for that.
Q9	Manually never added anything. The wiki pages that I already visited I could find those back at once. Felt euphoric.
Q10	I think so. Dont know whats all down there. What I see is that the number of likes is key. But I assume more complex structures under the hood. Pages with more likes always come higher than with fewer likes. Likes are created by clicking the like button or clicking on results.
Q11	In a certain domain from the list, visit pages, and manually is also possible.
Q12	omnibox: type in "k" and Kiwi is used. "kiwi.comprise.com" also sometimes
Q13	Yes, sufficient, interface clear, fast enough, didnt notice any delays, no effort to add results, costs a minute. Not relevant how much effort. Shouldn't be possible.
Suggestions?	Relevance should be directly linked to use of that page. Coupling with SE like Google and extend like system (relevance system) need additional research and has potential. Best would be to keep indexing URLs and that it keeps working like it is now. Multiple browser would be nice, not only Chrome, could be quite hard to do though.
Q13	For me as employee yes, and for ***** yes, sure

Development Manager 1	
Q0	Yes, for internal documents
Q1	Yes, search engine for ***** related documents. Goal is to find information in one place within *****.
Q2	No, once a week
Q3	Don't know
Q4	No
Q5	Yes, because many resources I visit are relevant to my work and are also indexed by the system if I access them
Q6	Not in its current form, with the provided suggestions I think it can
Q7	Not seriously tried any queries.
Q8	Yes
Q9	Likes, you can click the like buttons to modify the ranking of documents
Q10	Yes, with the Chrome extension
Q11	Yes, added to my Favourites list
Q12	No, I don't want to visit pages, let them automatically be indexed by the extension, before I can access them. There should also be a sort of crawler
Suggestions?	I'm allergic to configuration. You, Marco, had to configure the domains that are indexed. This should be determined intelligently by the system. It should determine based on visits which domains are relevant and also index those domains. Tags would also be nice to tuse "kernwoorden". System should also work on all browsers. External documents could be optional to search through.
Q13	Yes, for both ***** and me as an employee if the provided suggestions are implemented. I think that the prototype is a good basis for a very useful system. I would like to know the results from your study (quantitavely and qualitatively) and whether you are going to actually going to bring this product to the market

User Experience Designer 1	
Q0	Yes, would be useful
Q1	Yes, goal is to easily access and find information over all software systems of *****
Q2	No, 9 times a week, mainly used for the UX guide I was working on
Q3	Don't know
Q4	No, very useful tool though, the main thing to use if you want information quickly from internal systems
Q5	It gives satisfying results, fast information access
Q6	You find information faster, also depends on third parties, how they hide their information. Suggestions (instant) would be nice. I am satisfied with the system as is 100%, could be because of the things I was looking for. Specific pages in the domain were maybe not always retrieved over the main domain homepage, but I am not sure about that anymore.
Q7	
Q8	yes
Q9	Yes with likes
Q10	Yes, with the Chrome tool automatically
Q11	Yes, bookmarked the link
Q12	Yes, clear interface, could put the search bar in the middle of the page instead of top left, fast enough, effortless to get results on top, put my own UX guide on top of the rankings without much effort
Suggestions?	Make the tool available for other browsers than only the Chrome browser.
Q13	Yes, handy tool to find information with respect to the organisation quickly

Application Specialist	
Q0	Yes, would be very useful to have
Q1	Yes, the goal is one central point where you can search for information. Nice if you dont know where to search to have a central point of access. Not all data has to be in one place, that makes sense, but it would be useful to have a place where you can search in all different systems
Q2	Every day, 1 to 2 times a day
Q3	We didn't talk about it much, hard to estimate, I think they think it's good to have one central point
Q4	Don't know whether they expect that, but I think they find it useful
Q5	Yes, but that is what you can make it yourself. That's what I find useful about it, if you access a page once, you find it back in Kiwi
Q6	Not yet, but if you need information outside your standard environment it could be useful. More users could lead to irrelevant links because they are not working on the same thing. A product tag could work to distinguish information resources.
Q7	one in four, this ratio increased. I was satisfied with 1 in 4. You know that it is something that is being build op, more information is becoming available all the time
Q8	Didn't add anything manually, with the extension I could find them back in the system
Q9	I think it works with with keywords and the like system. Likes are based generated by clicks on results and manually liking pages
Q10	I know it can be done manually, didn't do this myself. The Chrome tool indexes automatically in the list of domains if I browse them
Q11	If I open Chrome it opens multiple tabs including a tab with Kiwi. Also access the URL
Q12	Yes, you type in your query and get results. Kiwi is fast enough, no effort to add results.
Suggestions?	Getting on top can be done by opening pages and assigning likes. A like fight could arise when the group of users becomes larger. This could be prevented by dividing departments. A department with 5 people would "lose" from a department with 20 persons. There needs to be seperation or a weighing factor for this to compensate.
Q13	Yes to both, same answer to both: useful to have one point of access to information instead of searching multiple platforms.

Development Manager 2	
Q0	Yes, of course. It should be one clear way to access information anytime anywhere.
Q1	Add-on for Chrome that builds up an index. Goal is productivity increase: speed, secure, information enrichment
Q2	No, I didn't exactly know anymore how it worked.
Q3	Don't know
Q4	They would say yes, but they don't do anything about it. He thinks it is somewhere in my functionomschrijving
Q5	Don't know
Q6	I think so, it could help. There are many systems, is this the system we should choose, compare it to Hyves and Facebook. Make the tool browser independent
Q7	Wasn't clear when I was searching with Google and when I was searching with Kiwi
Q8	Yes
Q9	No, don't know
Q10	I thought that everything I was browsing related to HR was indexed based on my browse behavior
Q11	"kiwi.org" or something. I think I also Googled Kiwi. I never turn my PC off, so the screen has been open for a few weeks as well
Q12	No, don't know how it can be improved, can not visualize it very well right now. A more graphical interface would have been nice. Fast enough, automatic adding of results so not much effort, easy. Getting results on top I haven't tried. Also because I don't know how that works
Suggestions?	Beforehand, I didn't make much use of the system. Mainly because I don't use Chrome much during my work. That is an issue, it should be working in all browsers. Second point: the information about the case study, I would have liked some additional information in textual format. An e-mail would have been nice with some information about the project. Maybe an AI agent would be useful that you can ask questions. This agent should also be able to deduct new knowledge from existing knowledge. I worked on such a product at another organisation for a service desk, new generated knowledge was deducted from existing knowledge. So maybe a chat robot could help who can answer questions, in our case the knowledge base became larger and the service desk had less work to do because the robot could answer ever more questions.
Q13	Yes I think so, if it's integrated. In a perfect situations in which the organisation has one database, ons central point of access. If it is a tool on its own that can be used to search in only a part of the information it would not work. A knowledge network (graph) with related entities could be indexed by tools like Kiwi where people will get lost on their own.

User Experience Designer 2	
Q0	Yes, should be part of *****net, intrasite
Q1	Functionally not really clear yet, doesn't seem to work like it should. Goal is a single point of access. Dont know if I have to look at it as a search engine.
Q2	Not that much, Mostly use Internet Explorer. between 1 and 10 times a month
Q3	Didnt talk about it much. I think one collegue was quite enthusiastic
Q4	No the word doesnt get spread
Q5	Depends on how you organise it yourself. Went into the system ad hoc and found a few things
Q6	I don't have to surf that much, usually I go via IE and use Google. With improvements it could work. Google is too elaborated, too many commercial items. If we can reduce that in Kiwi, and save used links in Kiwi instead of my favorites list that would be better and easier
Q7	Adding a results seems to be a bit buggy. Added an incorrect link and this resulted in a wrong link. Domain registration advertisement instead of results. Reat.com, you could check whether you could reproduce this. Adding results manually is not intuitive. Don't know how often queries had success
Q8	That didn't seem to work well. First I worked without logging in. In My links I didn't see the links that I added
Q9	Not sure, but it had to do with likes I think
Q10	Yes, manually. Don't know other ways of adding results
Q11	As favorites, bookmark. No effort, bookmark
Q12	Except for a few things yes, fast yes, adding results manually is easy, also looks for additional details itself. Don't know about the ranking that much
Suggestions?	Favorites function alike system. Don't know where to pose queries to the system.
Q13	Hard question, for myself, I would like a restriction on the results of Google and that could be possible with this tool. Favorites search alike is easy. For the organisation, if I have to search a lot yes this could improve efficiency. 500X5 seconds a person a year. In the long term this could lead to an improvement in efficiency. Would be nice to have a sort of favorites tool, that is how I see it. And if it's not available in the favorites, you go to Google again

BI specialist	
Q0	Yes, you could make difference between structured information and unstructured information. Would be nice if questions can be answered in one place in natural language for both structured and unstructured information
Q1	Search engine, I think the goal is a search engine for internal sites, where value of information is based on likes and comments
Q2	Yes
Q3	Well my colleagues like it, my manager as well, but he doesn't know that yet. Colleagues didn't participate in the case study so didn't use it, but if it would be implemented organisation wide, they would definitely use it. I have several examples of results that were very useful and I didnt even know these resources actually existed.
Q4	No
Q5	It could, the more of my colleagues would use it, the better the result would be related to my work
Q6	Yes, a Single point of access, in this case for unstructured data, would make my work much more efficient. In Microsoft Enquiry, you can ask questions in natural language, that would make the system even better. So quering structured data as well in natural language
Q7	I knew where I was looking for, but 6 of 10 was found. When not found, it was most of the time not indexed. I was very satisfied with the ranking and success ratio
Q8	Didn't try this, also used the bookmarklet to add results
Q9	Based on likes, maybe also on how much a document is search for
Q10	Click on the plus and add it (bookmarklet). I thought that the extension does automatic indexation based on my browse behavior, like the NSA
Q11	Yes, maybe my homepage but at least it is in my favourites list, link is "kiwi.comprise.com". Is there a good reason why that is the URL? ***** domain would be more logical, this could give the impression that you are on an external site. Also makes it more trustworthy
Q12	Yes, nice and clean interface, there are some things that can be improved. If you click on a result and it is not what you want, you should be able to click "back" and instantly be presented with the query again already filled in in the search engine. Now I had to reenter the query if I hit the back button. Effortless adding of results. You are not allowed to manipulate the ranking, I would not know how
Suggestions?	Bookmarklet should close if a result is added. Now it stays there. Structured data search would be very interesting, then it would be even more useful. In my field of work, I always get questions from management where structured data is necessary. If they would have a place to ask those questions in natural language, that would be very useful, also for me
Q13	Yes, clearly. With this I can add results quickly. Nice to see what colleagues like and that has given me some valueble information resources.

Support Consultant	
Q0	Yes
Q1	I've been ill for a week. I think it's a point to access information related to *****. Goal is to see what your colleagues search for and bring up documents higher in the rankings. And see which questions colleagues ask
Q2	Except for the week I was ill, 4/5 times a week
Q3	Dont know, manager doesnt know it exists
Q4	No
Q5	Not completely but partially, mainly development issue related. No service desk oriented and customer oriented information. That content is missing.
Q6	Don't know yet, I used it for a too short period of time to say something about those things.
Q7	More content is important though
Q8	1 in 5. I was satisfied with that
Q9	I didn't add any documents
Q10	Yes, I think the more often that document is found and the the more likes a document gets, the higher the result will be ranked
Q11	No, didn't play with that yet
Q12	Yes, as homepage. If I start the browser
Suggestions?	Yes, keep it like this, fast enough, dont know about adding results, not much effort to get results on top
Q13	More content Yes, I think so. Probably a few things should be change, From my "service desk" point of view, feedback to the rest of the organisation could be communicated via the system. Useful for intern use in *****. Communicating to the rest of the organisation what the customer wants? I miss internally, feedback from frontend (commercial) to backend of the organisation
C10	
Q0	Yes, definetely, one way to acces information would be great
Q1	Kiwi is an Enterprise Search engine where you can search for web sources related to the organisation. You find links to ***** related sites. Goal is to have a fast and easy way to search within the company to work related information
Q2	
Q3	Didnt have the time to share with colleagues, one colleague found it a nice idea. We are working on one single cloud system, where search could be a part of this large project
Q4	No
Q5	Yes, with the limited knowledge I have about the system, I think so
Q6	Yes, finding information faster is always good. Make it available in all relevant browsers. Mainly the browsers and versions selected in the A-matrix for ***** That would stimulate use, I am an Internet Explorer user and I have to think about Kiwi specifcly now before I use it.
Q7	Didn't use the system much, but I guess 50%
Q8	Not directly, could also be because I am unfamiliar with the system.
Q9	Documents that are viewed more often and liked more often
Q10	Not everything is added, but if you specify a certain area it should be taken into account/indexed
Q11	First go to Chrome, then favorites or url "kiwicomprise"
Q12	Yes, simple version of Google layout, looks familiar that is good for user acceptance, doesn't look bad, interface clear, there was a delay in showing character when typing in a query. What I did like is that searching occurred in the background and I didn't even had to press enter before results were returned. That costs extra computing power though, you should check whether that is feasible
Suggestions?	Hard to say, don't know enough about and don't have enough experience with the system.
Q13	Yes I think so, not sure whether this is the system we should use for enterprise search though.

C Paper submitted to KDIR conference 2014

The Social Score: a new method to determine the relative importance of webpages based on Online Social Signals

First Author Name¹, Second Author Name¹ and Third Author Name²

¹*Institute of Problem Solving, XYZ University, My Street, MyTown, MyCountry*

²*Department of Computing, Main University, MySecondTown, MyCountry*
{f_author; s_author}@ips.xyz.edu, t_author@dc.mu.edu

Keywords: Social Score, PageRank, Web Search, Top-K Ranking, Quality Assessment, Data Analytics, Information Extraction.

Abstract: There are many ways to determine the importance of webpages, the most successful one being the PageRank algorithm. In this paper we describe an alternative ranking method that we call the Social Score method. The Social Score of a webpage is based on the number of likes, tweets, bookmarks and other sorts of intensified information from Social Media platforms. By determining the importance of webpages based on this kind of information, ranking becomes based on a democratic system instead of a system in which only web authors influence the ranking of results. Based on an experiment we conclude that the Social Score is a great alternative to PageRank that could be used as an additional property to take into account in Web Search Engines.

1 INTRODUCTION

It has been proven that top-K ranking in Web Search cannot be based only on matching queries to documents (Manning et al., 2008). Although many techniques for ordering Webpages based on query-document similarity are present, it is not sufficient to get good results in Web Search. There are too many resources that look too much alike to determine which documents are more relevant than others. Search Engines turn to other factors to determine the general importance of webpages and take that into account especially when many documents contain approximately the same content. What appears to work best is to use query-document matching ranking techniques in combination with query-independent ranking techniques. This way, Search Engines are able to determine what search results should be shown on top, even when a query matches many documents more or less equally well. Important concepts in query-document matching are Term FrequencyInverse Document Frequency (TF-IDF) ranking and metadata extraction (Salton and McGill, 1983; Hu et al., 2005). Those are then combined with query-independent ranking techniques to gain better results. One such factor to determine the general importance of webpages in which the match between a document and the query is taken out of the equation is PageRank (Brin and Page, 1998). With PageRank, the value

of pages is determined by the Link structure on the Web. When a page is linked to from many important webpages, the page itself is also considered to be important. Using a recursive algorithm, the relative value of every webpage is calculated. This way is based on how the value of scientific papers can also be estimated: based on the number of times your paper is cited by other papers that all also have their own relative value in the community. PageRank has been proven very successful and it is often referred to as the algorithm that made Google gain its huge market share in Web Search. Such a system in which links determine the value of webpages is also referred to as the Link Economy (Gerlitz and Helmond, 2013). In the Link Economy, the power to influence the ranking of search results is with the authors on the web. Before the Link Economy, there was the Hit Economy, in which the value of webpages was mainly based on the visit counters that were available on webpages. In the Hit Economy, the number of visits determined the value of webpages and the power to influence the rankings was directly in the hands of all internet users. We propose a method to give back the power to influence Web Search rankings directly to all internet users instead of only web authors. The system is based on the Like Economy, in which the value of webpages is based on online Social Signals such as likes, tweets, mentions, shares, bookmarks and pins. For every webpage we calculate a Social Score, which

is based on Social Signals from multiple Social Media platforms. In 2007, Bao, Wu, Fei, Xue, Su and Yu also saw the potential of social annotations to determine the value of webpages (Bao et al., 2007). Although they took a different approach with their ranking method that they call SocialPageRank, the idea is similar to the Social Score method as proposed in this paper. The main differences between the approaches are that SocialPageRank makes use of more complex math calculations whereas the Social Score makes use of simpler math and is easier to understand. Furthermore, the computational complexity of the Social Score method to calculate the Social Score of one Webpage is $O(1)$ whereas in the SocialPageRank method it is not possible to calculate any individual Score for a webpage without calculating the other scores for the other webpages. This is because SocialPageRank makes use of recursive Matrix multiplications just like PageRank does to converge to a stable scoring model. In each iteration the computational complexity is $O(|U||W| + |s||W| + |U||s|)$ where $|U|$ is the number of users U of the Social Media platform, $|W|$ is the number of Webpages W in a Corpus C and $|s|$ is the number of social annotations or Social Signals. The number of iterations determines the accuracy of the resulting scores for the Webpages. Last, SocialPageRank only makes use of data from one Social Media platform what leaves more open space for bias. The Social Score method is more generic and can take into account as many Social Signals from as many Social Media Platforms as desired.

2 SOCIAL SCORE

The Social Score can be calculated for every webpage individually. An arbitrary number of Social Signals from different Social Media Platforms can be taken into account. To prevent bias towards a certain group of internet users or a certain domain, it is good practice to take as many Signals from as many Social Media Platforms as possible into account. To calculate the Social Score S for a Webpage W , we take into account n Social Signals related to Webpage W . The Social Score takes into account a list L of n Social Signal Scores s , where s is the number of Social Signals from one Social Media Platform. For example, s could be the number of shares of a Webpage W on Facebook or the number of tweets about W on Twitter. Now the Social Score S is calculated as defined in Equation 1.

$$S = \frac{\sum_{i=1}^n \log_{10}(1 + L_i)}{n} \quad (1)$$

An example of calculating the Social Score S for

a Webpage W only taking into account two Social Media platforms is as follows: let's say that the website "example.com" has 99 likes on Facebook and has been mentioned in 9 tweets on Twitter. Then, $L_1 = 99$, $L_2 = 9$ and $n = 2$. We calculate the sub scores per Social Signal and divide by n . The sub score for likes on Facebook is $\log_{10}(1 + 99) = 2$ and the sub score for tweets on Twitter is $\log_{10}(1 + 9) = 1$. To calculate Social Score S for Webpage W we take the average resulting in $S = 1.5$. As you can see in this example, the Social Score increases with the number of likes and shares. Table 1 gives some more examples in which three Social Signals are taken into account: the number of likes on Facebook, the number of tweets on Twitter and the number of bookmarks on Delicious. From this table we can infer that S is higher when Social Signal Scores are in balance. When they are out of balance, the same total number of likes, tweets and bookmarks result in a lower Social Score S . This happens because the \log_{10} is taken of every individual Social Signal score s and not after summing them all up first. For example, only having 999 likes on Facebook results in a Social Score of 1.00 whereas 99 likes, 99 tweets and 99 bookmarks result in a Social Score of 2.00. Intuitively this makes sense because it looks like in the first case there is a bias towards Facebook and likes whereas in the second case there seems to be a balance between the Social Media Platforms.

Although there were several reasons to choose particularly for the \log_{10} in Equation 1, we did not experiment with other \log s and it could be that another \log would perform better in practice. What we can say about the \log , is that if you lower it, there will be more room for bias and if you increase it there will be less room for bias. That is because every individual Social Signal that is taken into account gets more influence on S if a lower \log is taken and gets less influence on S when a higher \log is taken. The first reason we chose for \log_{10} is that it is relatively easy to interpret for people. For example, a Social Score S of 1.00 can be interpreted as 10 likes, 10 tweets and 10 bookmarks assuming a balanced distribution over the Social Media platforms. Assuming equally distributed Social Signals over the Social Media platforms, the Social Score can be explained as the order of magnitude for the underlying Social Signal scores. Furthermore, using the \log_{10} gives a quite good scale for the Social Score. When a Webpage would have one billion Social Signals per Social Media platform that is taken into account, that page would have a Social Score S of 9.00. Currently there are no such webpages present in the world that have that many Social Signals related to them on any Social Media platform. Therefore, we can safely assume that the So-

cial Score will always produce values between 0 and 9 disregarded which Social Media platforms are used to calculate the Social Score.

Table 1: Examples of calculating Social Score S given three Social Signal Scores s

Likes	Tweets	Bookmarks	Social Score S
0	0	0	0.00
999	0	0	1.00
99	99	99	2.00
333	333	333	2.52
10^8	10^7	10^6	7.00

The Social Score S can be calculated in $O(1)$ time by making use of the Application Programming Interfaces (APIs) of the Social Media Platforms. Now let’s consider a Corpus C consisting of indexed Webpages W . To determine all the Social Scores of all Webpages W in C , this would take $O(|C|)$ time. Therefore, we can say that computational complexity increases linearly with the size of Corpus C . Notice that the Social Score S of a Webpage W will generally change over time because people keep interacting with the Webpage W via Social Media Platforms.

3 EXPERIMENT

To be able to validate whether the Social Score S can accurately determine the importance of webpages, an experiment was performed in which a Corpus C of over 120 000 webpages was gathered. The experiment was part of a larger experiment about Social Search engine quality in which a prototype was built and tested. Based on work from Evans and Chi (Evans and Chi, 2008) and Golovchinsky, Pickens and Back (Golovchinsky et al., 2009), we define asynchronous Social Search as

”Information seeking supported by a network of people where collaboration takes place in a nonconcurrent way.”

Important concepts in asynchronous Social Search are user-generated content and user feedback.

There were three ways in which results could be added to the prototype. The first one was manually, by filling in a URL, title, description and keywords. Figure 1 provides a screenshot of what this way looked like in practice. The second was by adding a bookmarklet to your favourites in your web browser. When a user had the bookmarklet in his favourites list in his web browser and he visited a website, he could click on the bookmarklet. This resulted in a popup of the search engine with a form shown to add a result to the

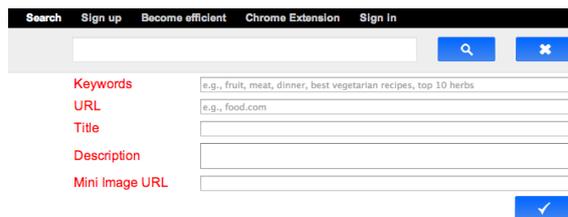


Figure 1: Screenshot of how a link could be added manually.

search engine. In this form, the URL, title, description and keywords are already filled in based on the page that the user is currently visiting. This second way of adding webpages to the search engine is less time consuming than the first. An example is shown in Figure 2. The third way to add search results to the search engine was by installing an extension for the Chrome web browser. By installing this extension, all the websites that were visited by the user were added to the search engine automatically. To guarantee a decent corpus size, the API of bookmarking website Delicious was also used to enrich the corpus with resources tagged publicly on Delicious. Delicious was launched in 2003 and enables people to tag webpages and discover them later on. In other words, Delicious is an online bookmarking service (Golder and Huberman, 2006). 29 215 resources were acquired via the Delicious API. We also know that the rest of the Corpus was mainly gathered by tracking the browse behaviour of just over 20 participants that installed the Chrome Extension. That means that every user of the extension roughly attributed 4 500 resources to the index during the experiment. Gathering resources started in June 2013. First, this happened only manually, then the bookmarklet was released and later on also the Chrome extension was released in September 2013. The end of the measurement period for the experiment was the seventh of January 2014.

For every page a Social Score S was calculated based on Signal Scores s from seven Social Media Platforms. The Social Media platforms used in the experiment were Facebook, Twitter, Pinterest, Google+, StumbleUpon, Delicious and Linked In. From Facebook and LinkedIn, the number of times the URL was shared was acquired. From Twitter the number of tweets in which the URL was mentioned was acquired. From Pinterest, the total number of times that items were pinned on the webpage was acquired. From Google+, the number of people that +1'd a URL was acquired. From StumbleUpon, the number of times a URL was stumbled upon was acquired. Last, from Delicious, the total number of times a URL has been bookmarked is retrieved. Two example calculations are shown in Table 2. Here, two

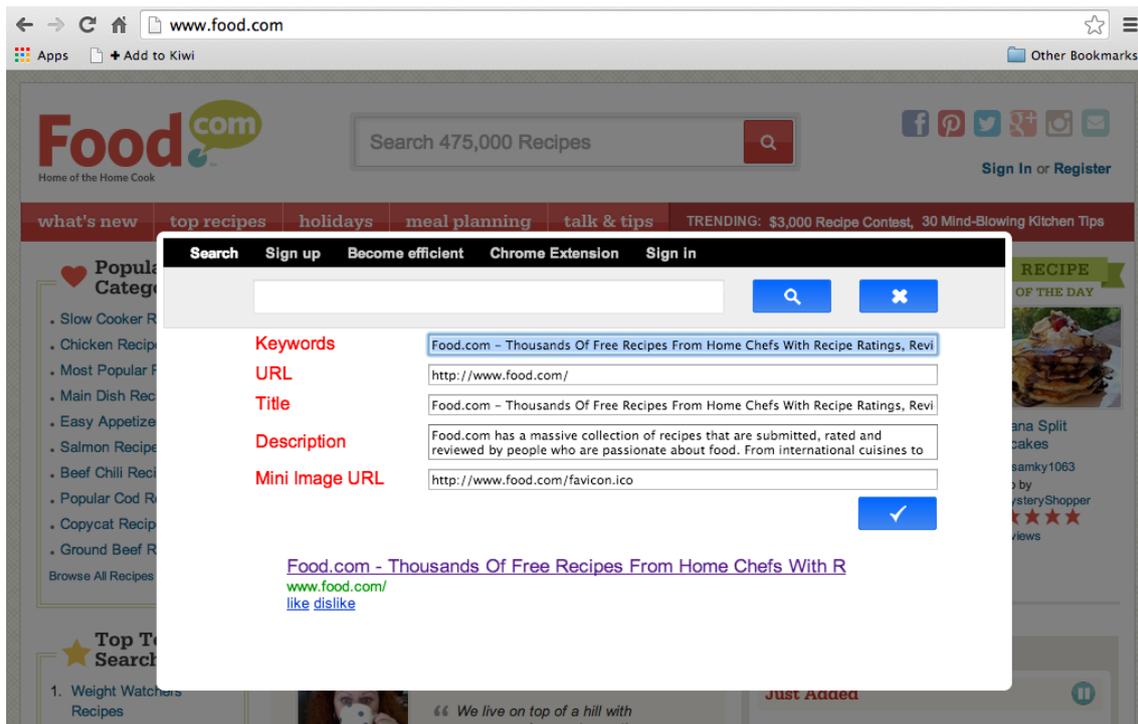


Figure 2: Example of how a link could be added using the bookmarklet.

webpages from different conferences are rated and it appears that <http://www.kdd.org/> is more important than <http://www.kdir.ic3k.org/> according to their Social Scores S . From the Corpus a top 50 was assembled based on Social Score S . Figure 4 provides an overview of the most important websites worldwide according to the Social Score S . Notice that there were ten duplicates in the list, such as <https://twitter.com> and <http://twitter.com> which both refer to the same content. Such duplicates were removed from the list. In large extent the list feels intuitively right. The most disturbing about the list is that Wikipedia has been ranked only 31st. In a PageRank algorithm Wikipedia would probably score top five, but apparently Wikipedia is not a source that many people frequently share or like via Social Media compared to the number of back links created to Wikipedia by authors on the web. Another interesting fact is that there are two Youtube videos in the top 50. Both are very popular songs that went viral via Social Media and therefore mainly scored high on Facebook and Twitter. Theoretically there could be sources missing that have never been indexed by the search engine. That would be rather unlikely though, because indexing is based mainly based on visits and you would expect that the most popular webpages on the web would have been visited at least once during this study by one or more users. It could be the case that there is a website in a large country or continent

of which no users participated in the experiment. Although we do not have exact data on where people came from that installed the extension that supported automatic browse behaviour tracking, we do have data about where the visitors of the search engine prototype came from. A map showing the distribution of sessions over the world is shown in Figure 3. The second biggest source of resources was Delicious. Delicious is used by a way broader audience which also decreases the chance that we are missing an important URL in our Top 50 according to the Social Score. Obviously, the ranking presented in Figure 4 changes over time. With every like, share or other form of Social Media interaction with respect to a Webpage, the ranking of the resource changes. It was outside the scope of this research to determine how often the Social Score should be updated.

4 CONCLUSIONS

The proposed concept of a Social Score for every web resource based on online Social Signals from Social Media platforms such as likes and shares is a promising alternative to existing methods to determine the query-independent importance of Webpages. We conclude that the proposed Social Score S is a good alternative for the more computational expensive PageRank and SocialPageRank methods in which iterative

Table 2: Social Signal Scores and Social Scores of <http://www.kdir.ic3k.org/> and <http://www.kdd.org/> based on data acquired on the 23rd of april 2014

	KDIR	KDD
Facebook	51	45
Twitter	1	11
Google+	1	12
Pinterest	0	0
StumbleUpon	2	1
Delicious	8	39
LinkedIn	0	1
Social Score	0.54	0.87

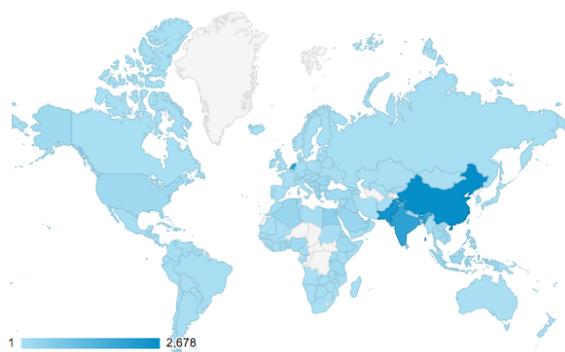


Figure 3: Distribution of sessions over the world measured from the first of June 2013 till the seventh of January 2014. Acquired using Google Analytics.

matrix multiplications are required. The Social Score of one Webpage W can be calculated in constant time and calculating the Social Scores for all Webpages W in a Corpus C would therefore be calculated in $O(|W|)$ time, so linear in the number of Webpages. Furthermore, contrary to PageRank and SocialPageRank, updating one score of one Webpage is actually possible using the Social Score method without any overhead. The Social Score could be a complementary property to take into account in existing search methods. By making use of the Social Score a shift can be made towards the Like Economy, away from the Link Economy. This way, all internet users will gain more direct influence on the ranking of results. It would indicate a shift from an Aristocracy in which the power is in the hands of the technically skilled web authors and developers to a direct democracy for Web Search.

More research should be performed to determine the quantity of likes compared to the quantity of links available on the web. Also the quality of Social Signals with respect to links could be compared in an experiment with a double blind test in which two identical search engines are used except one is using PageRank and the other is using the Social Score. When a user poses a query, both search methods are queried and the results are mixed as described by Joachim

#	URL	S
1	http://www.google.com/	5,90
2	http://www.facebook.com/	5,73
3	https://twitter.com/	5,50
4	http://www.youtube.com/	5,33
5	http://www.flickr.com/	5,14
6	http://www.amazon.com/	5,10
7	http://espn.go.com/	5,07
8	http://www.ted.com/	4,97
9	http://grooveshark.com/	4,94
10	http://www.pandora.com/	4,88
11	http://www.nytimes.com/	4,85
12	http://www.yahoo.com/	4,85
13	http://9gag.com/	4,79
14	http://www.ebay.com/	4,75
15	http://www.etsy.com/	4,74
16	http://www.apple.com/	4,70
17	http://www.imdb.com/	4,68
18	http://www.youtube.com/watch?v=9bZkp7q19f0	4,58
19	http://maps.google.com/	4,57
20	http://www.pinterest.com/	4,50
21	http://mashable.com/	4,49
22	http://www.nationalgeographic.com/	4,49
23	http://www.time.com/time/	4,48
24	http://www.linkedin.com/	4,40
25	http://www.rollingstone.com/	4,40
26	http://www.speedtest.net/	4,40
27	http://www.mtv.com/	4,39
28	http://www.codecademy.com/	4,35
29	http://www.kickstarter.com/	4,29
30	http://www.wix.com/	4,28
31	http://www.wikipedia.org/	4,26
32	http://www.fcbarcelona.com/	4,26
33	http://www.youtube.com/watch?v=jofNR_WkoCE	4,25
34	http://dictionary.reference.com/	4,25
35	http://translate.google.com/	4,25
36	http://www.indeed.com/	4,25
37	http://www.ign.com/	4,25
38	http://instagram.com/	4,21
39	http://www.asos.com/	4,21
40	http://digg.com/	4,19
41	http://www.last.fm/	4,18
42	http://www.stereomood.com/	4,17
43	http://imgur.com/	4,17
44	http://thenextweb.com/	4,16
45	http://www.picmonkey.com/	4,15
46	http://edition.cnn.com/	4,15
47	http://www.apple.com/iphone/	4,14
48	https://mail.google.com/mail/	4,14
49	http://weavesilk.com/	4,13
50	http://www.weather.com/	4,12

Figure 4: Top 50 URLs worldwide according to Social Score S on march 27th 2014.

(Joachims, 2002). Based on which search engine receives the most number of clicks it can be determined which search method is performing better. The same kind of experiment could be performed to compare the Social Search method with SocialPageRank. It would also be interesting to experiment with the effects of taking into account different numbers of Social Signals from different numbers of Social Media

platforms. It could be investigated whether Facebook Signals are from higher quality than Twitter Signals and whether bias can actually be removed by taking into account more Social Signals and Social Media platforms.

Also, the proneness of the Social Score to malicious use and spam should be evaluated. Would it be easier or harder to influence the ranking of results when Likes are used instead of Links to determine the value of webpages? Research could be performed to investigate how such malicious use of the Social Score could be prevented effectively. One direction for a solution might be to use many Social Media Platforms to force spammers to spam many different systems. One advantage of the Social Score is that spam and malicious use is not only the problem of the Search Engine, but also a direct problem for the Social Media platforms at hand.

REFERENCES

- Bao, S., Xue, G., Wu, X., Yu, Y., Fei, B., and Su, Z. (2007). Optimizing web search using social annotations. In *Proceedings of the 16th international conference on World Wide Web*, pages 501–510. ACM.
- Brin, S. and Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1):107–117.
- Evans, B. M. and Chi, E. H. (2008). Towards a model of understanding social search. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 485–494. ACM.
- Gerlitz, C. and Helmond, A. (2013). The like economy: Social buttons and the data-intensive web. *New Media & Society*, 15(8):1348–1365.
- Golder, S. A. and Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of information science*, 32(2):198–208.
- Golovchinsky, G., Pickens, J., and Back, M. (2009). A taxonomy of collaboration in online information seeking. *arXiv preprint arXiv:0908.0704*.
- Hu, Y., Xin, G., Song, R., Hu, G., Shi, S., Cao, Y., and Li, H. (2005). extraction from bodies of html documents and its application to web page retrieval. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 250–257. ACM.
- Joachims, T. (2002). Unbiased evaluation of retrieval quality using clickthrough data. In *SIGIR Workshop on Mathematical/Formal Methods in Information Retrieval*, volume 354. Citeseer.
- Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge.
- Salton, G. and McGill, M. J. (1983). *Introduction to modern information retrieval*.

D Paper submitted to Palgrave Macmillan KMRP

Asynchronous Social Search: a case study to investigate the added value of using asynchronous Social Search in Enterprise Search as a Single Point of Access to organisational information

First Author Name¹, Second Author Name¹ and Third Author Name²

¹*Institute of Problem Solving, XYZ University, My Street, MyTown, MyCountry*

²*Department of Computing, Main University, MySecondTown, MyCountry
{f_author, s_author}@ips.xyz.edu, t_author@dc.mu.edu*

Keywords: Enterprise Search, asynchronous Social Search, Case Study, Single Point of Access, Prototype.

Abstract: A fundamentally different approach to Enterprise Search is described in this paper. The approach is based on asynchronous Social Search, a search method in which people collaborate to find the information they are looking for. A prototype was built to test the feasibility in a business environment. A case study was performed at an organisation with over 1 000 employees to evaluate the quality of asynchronous Social Search as a Single Point of Access to information. Based on the results we conclude that asynchronous Social Search has great potential for Enterprise Search, as a Single Point of Access to organisational information. Both because the implementation requires no integration with the existing Information Technology infrastructure of organisations and participants were very satisfied with the results provided by the prototype.

1 INTRODUCTION

Sharing information and knowledge in a business environment has become an important topic for organisations. From a business point of view, many of their information resources should only be available internally. Reasons to keep access restricted to people from the organisation are often related to privacy and competitive advantages. By using intranets, the barriers to share knowledge have been lowered. However, direct access to the resources is only available to a small part of the organisation. In many cases, only specific business units or workgroups have access to information that would also be valuable to many others in the organisation. In such an environment information is stored in so called data silos. One major problem that leads to inaccessibility of resources is information overload. The information infrastructure is often complicated to understand and the amount of knowledge available is way too much. The right information at the right time becomes as good as invisible as a needle in a haystack (Offsey, 1997). There is a need for a solution which doesn't require extensive change to the information systems of the organisation. According to Accenture, one of the first steps that should be taken is to provide the people of the organisation with a Single Point of Access (Nanterme

and Daugherty, 2014).

Based on work from Evans and Chi (Evans and Chi, 2008) and Golovchinsky, Pickens and Back (Golovchinsky et al., 2009), we define asynchronous Social Search as

"Information seeking supported by a network of people where collaboration takes place in a nonconcurrent way".

Important concepts in asynchronous Social Search are user-generated content and user feedback. In this paper we describe a prototype that is based on asynchronous Social Search and we evaluate the feasibility of it for Enterprise Search.

2 REQUIREMENTS SPECIFICATION

To be able to validate the hypothesis that asynchronous Social Search provides better quality in search results than traditional search methods a prototype was built. There are three ways in which results can be added to the prototype. The first is manually, by filling in a URL, title, description and keywords. Figure 1 provides a screenshot of what this way looks like in practice. The second way is by adding a book-

Field	Value
Keywords	e.g., fruit, meat, dinner, best vegetarian recipes, top 10 herbs
URL	e.g., food.com
Title	
Description	
Mini Image URL	

Figure 1: Screenshot of how a link can be added manually.

market to your favourites in your web browser. When a user has the bookmarklet in his favourites list in his web browser and he visits a website, he can click on the bookmarklet. This results in a popup of the search engine with a form shown to add a result to the search engine. In this form, the URL, title, description and keywords are already filled in based on the page that the user is currently visiting. This second way of adding webpages to the search engine is less time consuming than the first. An example is shown in Figure 2. The third way to add search results to the search engine is by installing an extension for the Chrome web browser. By installing this extension, all the websites that are visited by the user are added to the search engine automatically. When a result is added using the extension, all content of the page that is being added is indexed in case that the page makes use of the HTTP protocol and not of the HTTPS protocol. To guarantee a decent corpus size, the API of bookmarking website Delicious was used to enrich the corpus with resources tagged publicly on Delicious.

A ranking of search results given a query Q is based on credits. Credits are assigned to search results regarding queries in two ways. First, a user can like and dislike a search result given a query. Second, when a user clicks on a search result, credits are assigned to the result provided the query.

Users can sign up for an account in the prototype. Because of privacy issues, the Chrome extension differentiates between the HTTP and HTTPS protocols. When the HTTP protocol is used when a user accesses a web resource, the resource is added to the search engine publicly, meaning that everyone can retrieve it. When the HTTPS protocol is used however, the web resource is only stored when the user is signed in on the search engine and will only be accessible to the user himself.

To provide an overview of the system in a systematic and generally accepted format, a Use Case Diagram and a Components Diagram were created. Use Case Diagrams are used to model the interaction that people should be able to have with a system, whereas Component Diagrams are used to show the software architecture and the flow of messages within that architecture, also showing the interfaces that are avail-

able in the architecture (D'souza and Wills, 1998). Figure 3 shows a Use Case Diagram stating all the actions that a user should be able to do with respect to the prototype. Figure 4 gives an overview of the software architecture using a Components Diagram. Both diagrams make use of the Unified Modeling Language notation.

3 CASE STUDY DESIGN

An embedded, closed, single-case holistic study as described by Yin, was performed with the proposed search method (Yin, 2009). With the case study we evaluated the capabilities of asynchronous Social Search methods to function as a Single Point of Access within an organisation. The search engine was hosted internally on the intranet such that knowledge became only available within the organisation. Employees used the Search Engine for five weeks and in that period more and more resources became available through the search engine. At the end of the five weeks semi-structured interviews were conducted to get information about how they experienced using the Search Engine.

The object of study was the proposed social search method. The main research question during this case study was:

Can asynchronous Social Search function as a proper Single Point of Access to information within an organisation?

A second question we wanted to answer was whether there is actually a need within the organisation for a Single Point of Access to information. To be able to answer the main research question in this case study, people from the organisation were interviewed that used the search method. Furthermore, quantitative data was assessed.

Another important question during case study design was whether we were measuring what we wanted to measure. We guaranteed construct validity by making use of both qualitative and quantitative data. In the case study, the cause was the introduction of the new search method in the organisation. We measured the effects of this introduction using both qualitative and quantitative data. This way we tried to guarantee internal validity. External validity based on a single case study is not really possible. However, if the results of the case study would indicate that the social search method is seen as added value to the organisation, this would be consistent with literature found that states that there is a need for a Single Point of Access to organisational information within organisations. Furthermore, there is consistent literature that

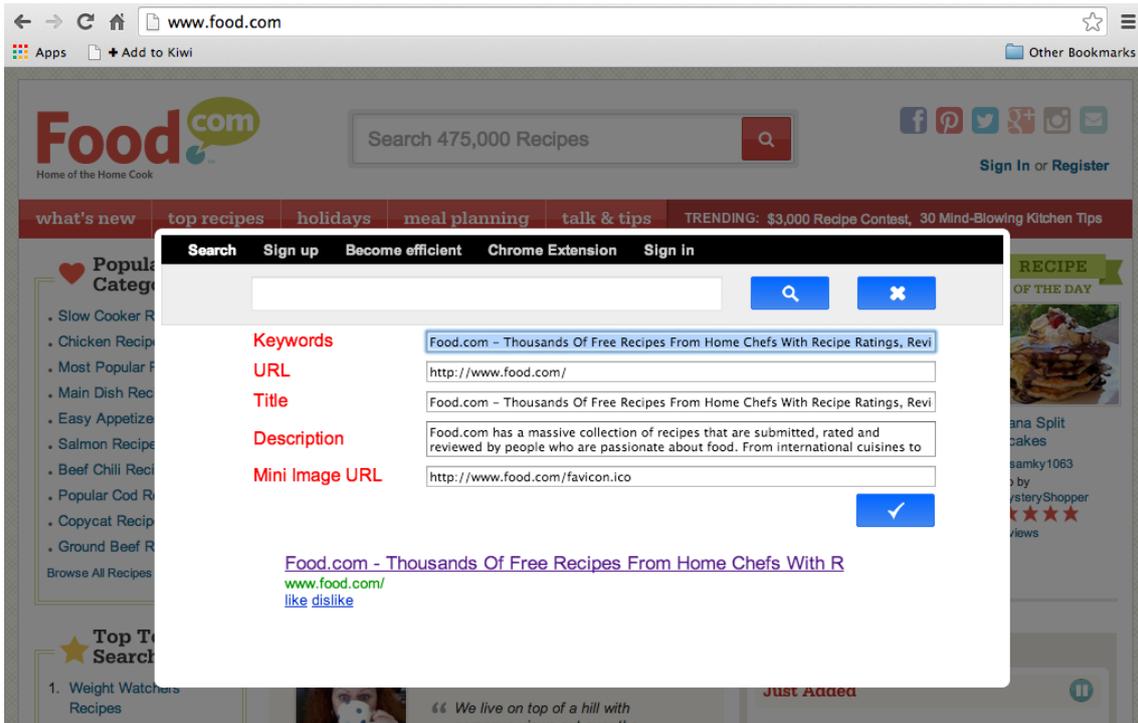


Figure 2: Example of how a link can be added using the bookmarklet.

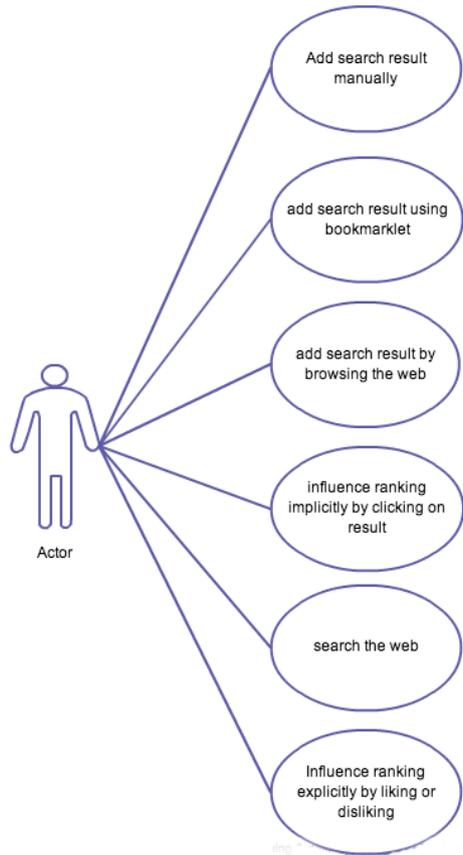


Figure 3: Use Case Diagram of the asynchronous Social Search Engine prototype.

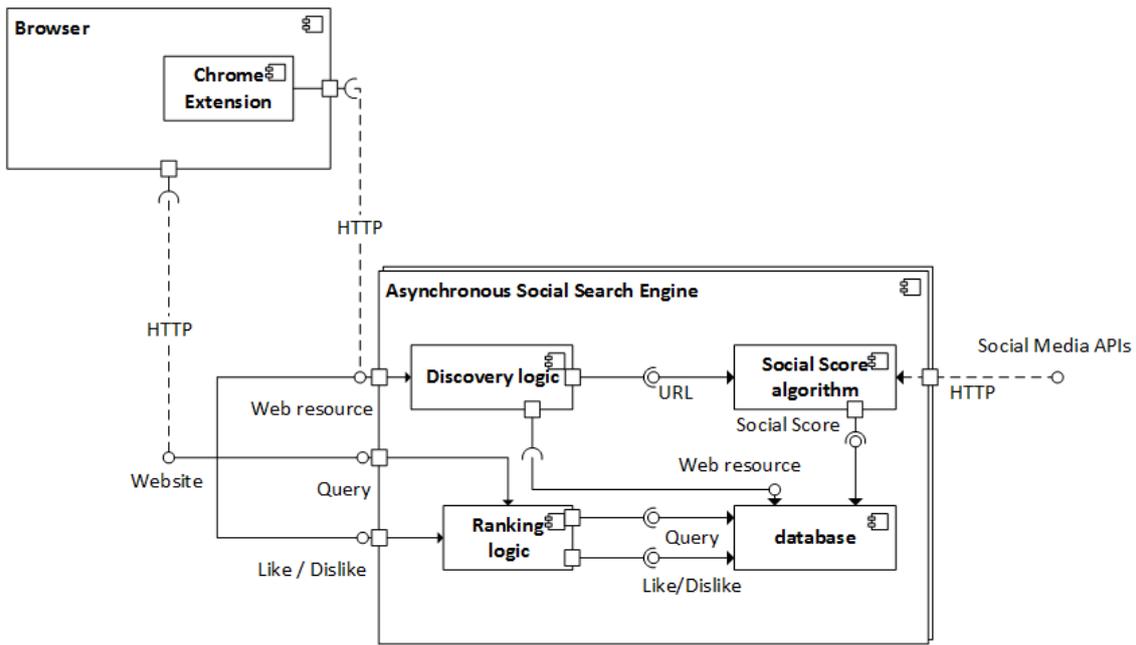


Figure 4: Components Diagram of the asynchronous Social Search Engine prototype.

emphasises the need for data lakes instead of data silos (Nanterme and Daugherty, 2014). Three criteria were used for the selection of an organisation to perform the case study at:

- the employees make use of web resources in their daily work, both internally hosted as externally hosted,
- the organisation should have an information infrastructure with multiple data sources, and
- the organisation has more than 100 employees, full filling over 100 Full-time equivalents (FTEs),
- the organisation must be, at least partially, physically located in the Netherlands since it would be impractical to perform the research further away from the researchers and university given the time and budget limitations of this research.

During this case study the Technology Acceptance Model 2 (TAM2) was used to determine the usefulness of the prototype and the potential of Social Search to function as a single point of access within organisations (Venkatesh and Davis, 2000). TAM2 is mainly based on TAM (Davis, 1989). TAM2 was chosen over TAM because it explicitly defines external factors that influence the perceived usefulness. Perceived usefulness was used as a surrogate for usefulness. Another option would have been to use the Unified Theory of Acceptance and Use of Technology (UTAUT) (Venkatesh et al., 2003). We chose to use TAM2 over UTAUT because UTAUT is more complex with 41 independent variables and 8 depen-

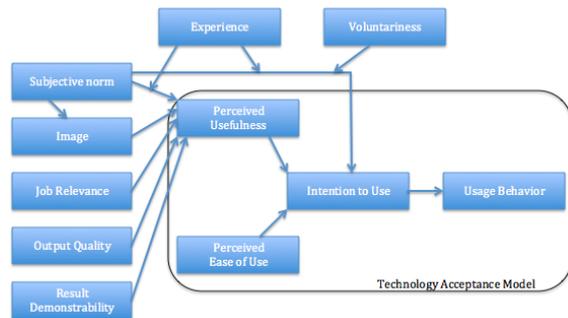


Figure 5: Technology Acceptance Model 2 as proposed by Venkatesh and Davis (Venkatesh and Davis, 2000)

dent variables (Bagozzi, 2007). UTAUT has also been criticised for being less parsimonious than TAM2 (Van Raaij and Schepers, 2008). Therefore, TAM2 is more practical to use for this case study in which only a small number of people participated and no conclusions could be drawn solely based on quantitative data. TAM2 is shown in Figure 5.

4 CASE STUDY RESULTS

An organisation was selected that is active in the Human Resource Management (e-HRM) and automation of wage and salary administration business with approximately one thousand employees. The organisation is mainly based in the Netherlands and possesses one of the largest development teams in the Nether-

Activity	Time period
Prepare and deploy prototype	Feb 3 - Feb 7
Convince people to join the project	Feb 10 - Feb 14
Quantitative Data collection	Feb 10 - March 14
Interviews with participants	March 3 - March 14
Report on results	March 17 - March 21

Table 1: Time schedule for the case study.

lands. The schedule shown in Table 1 was followed during the case study.

Only a preselected list of domains was indexed by the Chrome extension that were deemed to be relevant for the daily work. This list of domains that were automatically indexed using the Chrome extension was composed based on input from participants that actually installed the Chrome extension. The bookmarklet for adding results and the personal account feature were, unfortunately, not properly introduced to the participants and therefore not all participants were aware of the fact that these features existed. In total, 22 domains were indexed by the Chrome extension, of which some cannot be shown for reasons of confidentiality. A few that are not confidential are

- linkedin.com
- microsoft.com
- stackoverflow.com

The case study ran for 25 workdays from the 10th of february onwards. In total, 16 employees participated in the case study. All 16 installed the Chrome Extension and used the search method for a period of four to five weeks. Participation was voluntary, which is a relevant factor according to the TAM2 model. According to TAM2, subjective norm should not have any influence on Intention to Use if usage of a system is on voluntary basis (Venkatesh and Davis, 2000). In the interviews conducted in the fifth week of the case study, all other factors that can influence the Perceived Usefulness of a system according to TAM2 were asked about implicitly and sometimes explicitly. From the 16 participants, 15 were interviewed. The last one was on holidays. Table 2 gives an overview of the number of people per function in the participants group. In total, 22 unique visitors were identified and a total of 232 visits to the website were recorded with a total of 641 page views. Figure 6 gives an overview of the direct influencing factors from the TAM2 model on Perceived Usefulness and the actual influence the factors had on the Perceived

usefulness based on the interviews held with participants.

4.1 Perceived Ease of Use

Most people found the interface clear, simplistic, easy to understand and all except for one participant was happy with the performance of the system. For example, the Product owner said about the interface:

”It is clear, light and simple. It doesn’t take any effort to add results”.

People were able to access to prototype rapidly because they had stored the link in their bookmarks, used it as their default search method from the omnibox or set it up as their default page for opening new browser pages and page tabs. There were however, people that did not use Chrome as their default browser and this group of people had to take more effort to access and use the prototype because it was only properly functioning in the Chrome browser. There were also some people who would have liked some more explaining text in the interface to get them started. The majority however, had a clear understanding of the way the interface was meant to be used. There were also two persons who argued that it is not user friendly that you first have to visit a page before it is indexed by the search engine. They, and a few others, would have liked to have an additional sort of crawler in place that would index pages connected to visited pages automatically. There were also a few participants who said that adding results manually should have been easier. Most of the people were able to retrieve the documents they added either manually or implicitly by visiting pages using the Chrome extension without any trouble. People also indicated that it was rather easy to modify the ranking of results. There were a few people who found that it was too easy for people to manipulate the rankings on their own. They didn’t like the fact that likes could be assigned an indefinite number of times and not only once for one person. People also indicated that there should be a quick way to pose the same query to other search engines such as Google. A button should be provided which would instantly pose the same query to Google. Or, even better, Google results should be integrated into the interface in such a way that they come below the results from the prototype as it works right now. Several also indicated that it could be useful to show how a score for a resource is composed. Did it get 65 likes because one person liked it up on its own? Or did 40 different people click on the link and 25 people explicitly assigned a like to the resource? One of the participants that came up with this suggestion also noted that not too many changes should

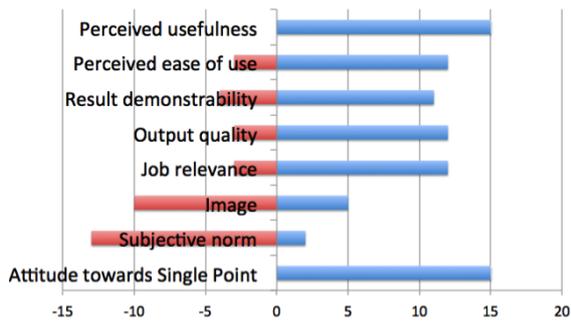


Figure 6: Direct influencing factors on Perceived Usefulness in the TAM2 model. The red horizontal bars indicate the number of people that experienced the factor as having a negative influence on Perceived Usefulness whereas the blue bars indicate the number of people that experienced the factor as having a positive influence on Perceived Usefulness.

Function	# of participants
Software Engineer	5
Support Consultant	2
Development Manager	2
User Experience Designer	2
Product Owner	1
Test Engineer	1
Application Specialist	1
Business Intelligence Specialist	1
Chief Information Officer	1

Table 2: Number of people per function in the participants group of the case study

be made to system as is, because things would only become more complex.

4.2 Result Demonstrability

Most people knew both how results could be added and how they were then ranked during querying. They correctly indicated that ranking is based mainly on likes and results can be added manually and using the Chrome extension. As Software Engineer 5 phrased it:

”What I see is that the number of likes is key in ranking results”

. One participant also found out about the bookmarklet, although this wasn’t the intention during the case study. It turned out however, that this participant was very enthusiastic about the bookmarklet. It was not always clear to people how likes could be assigned to search results. Most participants knew you could add them manually. Some indicated that likes were also generated implicitly based on page visits or clicks. Notice that the number of visits did not actually play a role in ranking although this was presumed by some participants. An interesting note is

that only one person mentioned the fact that keywords are also used when ranking results. The others probably took this for granted, because they all knew they could pose queries to the system.

4.3 Output Quality

Participants were asked whether they were satisfied with the results they got from the system, what percentage of queries lead to success and whether they could find back recourses they added to they system both implicitly and explicitly. On average, approximately one in three queries returned good results. Nearly all people were satisfied with their success ratio on queries. The participating Application Specialist stated:

”One in four, and this ratio increased. I was satisfied with one in four. You know that it is something that is being built up and more information is becoming available all the time”.

Some people indicated that there was a big difference in success ratio between intranet and internet search. Many indicated that the ratio differed with what you were looking for. The majority claimed that the tool was much more suitable for intranet search than for internet search. As Software Engineer 4 indicated:

”(The success ratio) depends on what you are looking for. Documentation like sources and intranet sources (high success ratio), but no technical sources like Stackoverflow (low success ratio)”.

Quite some people indicated that where search tools of internal systems failed, the prototype was a good alternative. Such internal systems included a Wiki, Sharepoint and Team Foundation Server. Not everyone tried to retrieve pages they added, but most of the people who added pages could find them back easily within the prototype. In general, we can conclude that people were rather satisfied with the output quality of the system.

4.4 Job Relevance

Virtually all employees indicated that the information available through the prototype was relevant for their daily work. As the Test Engineer indicated:

”You get ideas by using the system about resources you might not have known about before”.

Mainly the internally hosted resources on the intranet were found useful by most participants. Search is a task that all participants had to do regularly and

mainly for the intranet they found that the support of the prototype was good. For external sources such as Stackoverflow however, the support was often not good because not all resources were indexed by the system. Several people also indicated that they should have asked to index a few more domains with the Chrome extension automatically that were relevant to their work. People were also asked whether the tool made them do their work more efficient and effective. Most people thought it would, although there were also quite a few who said that some improvements needed to be made before this would actually be the case. Suggestions came in to support search on group level and individual level, taking only a small group of people into account, e.g., your own department. A side note to this suggestions was that tunnel vision should be prevented and you should not lose all access to information outside your own group. That would ruin the inspiring aspect about the prototype. People also asked whether sources hosted on disk could also be indexed and other document types such as PDF format. Another person indicated that it would be useful to work with tags to be able to distinguish information topics. One participant stated that the current prototype could only search in unstructured data and that it would be very useful to him if also structured data sources could be queried in natural language. He compared this feature to Microsoft Enquiry.

4.5 Image

Most participants indicated that they didn't know what their colleagues were thinking about the system or even knew that their colleagues did not know about the system. Software Engineer 3 for example, said:

"Didn't hear my colleagues about it and my manager doesn't know about it".

There were some participants that talked about the prototype with each other, all in a positive way. As the CIO stated:

"Didn't have the time to share with my colleagues, one colleague found it a nice idea".

In general, we conclude that Image had a negative influence on Perceived Usefulness, mainly because not much was known about it in the organisation outside the participants group. Using the system would therefore not enhance one's status within the organisation.

4.6 Subjective Norm

Participants were using the system for at least four weeks, but many of them did not use the system more

than a few times a week. In general, we can say that people did not have much experience with the system and therefore, Subjective Norm does have an influence on Perceived Usefulness in our case study. The vast majority of participants did not have the feeling that their supervisors and colleagues were expecting them to make use of the system. Therefore, Subjective Norm had a direct negative influence on Perceived Usefulness. Furthermore, according to the TAM2 model, Subjective Norm has a positive correlation with Image. Therefore, in our case study, Subjective Norm has a negative influence on Image and thereby also an additional indirect negative influence on Perceived Usefulness.

4.7 Quantitative data

Quantitative data was used to validate the qualitative data. The number of clicks, queries and the size of the index was recorded over time. Figure 7 gives an overview of the number of clicks recorded per week. It also shows the number of unique clicks per week. The number of unique clicks is defined as the number of queries in which one or more clicks were registered during a visit. The total number of clicks shows a bit of a random pattern, although you see a clear decline in the last two weeks. The number of unique clicks shows a more stable pattern, also decreasing slightly over time. Figure 8 gives an overview of the number of recorded queries per week, where each additional keystroke is recorded as a different query because of the instant search feature. Also the number of unique queries per week is shown. Here, the number of unique queries is defined as the number of unique queries posed to the system in one visit. In this case the number of unique queries also shows a more stable, slightly decreasing pattern. Last, Figure 9 gives an overview of the total index size at the start of every week. Notice that the index size at the start of the experiment was 40, because there was already one participant who started using the system one day before the start of the measuring period for the case study. During the third week of the experiment there was a decline in index growth. This can be explained by the fact that at the end of the second week, an update of the Chrome Extension was released. This had to do with the fact that additional domains were requested to be indexed. Unexpectedly, this led to Chrome automatically disabling the extension until the user explicitly gave permission that this additional domain could also be indexed. Therefore, many people had the extension disabled during the third week. The growth function of the index seems to be rather linear, but there is noticeable decrease in growth already

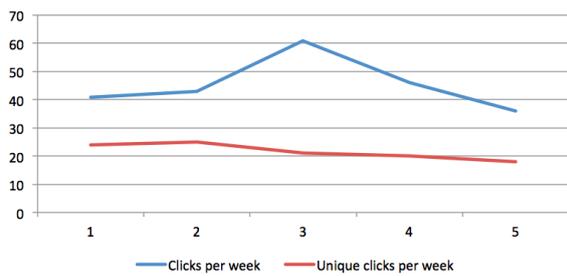


Figure 7: Number of recorded clicks per week

indicating logarithmic growth. This can be explained as follows. The majority of relevant resources has not yet been indexed by the system. Because slightly fewer resources are being indexed towards the end of the case study we can infer that people are also revisiting already indexed resources resulting in a decrease of index growth. This holds only by the assumption that participants did not change their frequency in browse behaviour using Chrome. In total, 3034 attempts were made to add a resource to the prototype, either using the Chrome extension or manually. Based on the interviews, we assume that over 99 per cent of attempts to add resources were made without any user effort, so using the Chrome extension. This means that approximately 3000 pages were visited by the 16 participants within the 22 domains that were allowed to be automatically indexed. Because the final index contained 1238 unique URLs, on average, every page in the index was visited over 2.4 times by the participants. Although we did not measure the number of attempted additions over time, we expect that this factor increased over time since, at the start of the case study, the index was very small and the chance to visit new resources using the Chrome was bigger than near the end of the experiment. The quantitative data indicates a slight decrease of use of the system over time. That could be explained as that people did not like the system. However, this would be inconsistent with qualitative data acquired via the interviews in which most people stated that the system could be of great help. Another explanation would be that people forgot about the system because they were too busy and the prototype was not an integral part of their daily work. This is also what we found during the interviews with participants.

4.8 What we learned

We believe that some essential changes need to be made to the prototype to make it really useful to people in supporting their daily work. All participants we interviewed indicated that there was definitely a need for a Single Point of Access through which all information related to the organisation could be accessed.

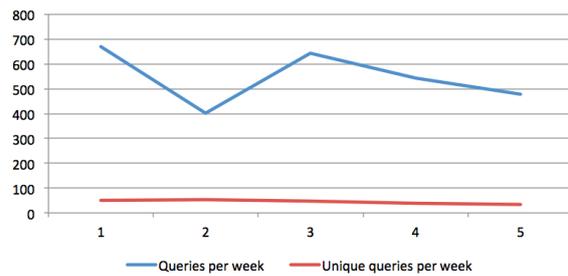


Figure 8: Number of recorded queries per week, where every additional keystroke during query typing is considered to be an additional query due to the instant search feature.

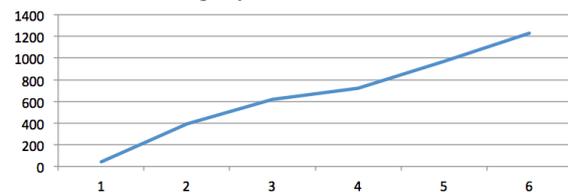


Figure 9: Size of the index measured at the start of every week.

All of them saw the prototype as a very good starting point for such a Single Point of Access. Many of the participants had additional feature requests for and remarks about the system, the most important ones being:

- **Browser independency.** Quite some participants made use of multiple browsers. The system should be made available in all browsers to make it easier to access the system.
- **The like system should be improved to prevent malicious use.** The ranking system with the likes should be made less prone to malicious use. Now people can assign an indefinite number of likes to a resource for a certain query. There should be a limit in the influence that one person can have on the ranking of a document. It could also be useful for people to see what a score of a page is composed of.
- **Trending topics.** It would be useful to know what colleagues are looking at a lot, also in the short term. Therefore, a list of trending topics and recently added pages that get many views could be very useful to employees. So next to the ranking based on all likes, there should also be a ranking based on recent likes.
- **Manually adding results should be made easier.** It takes too much effort to add results manually and the process is not always straightforward. Introducing the bookmarklet would be a great solution to this problem. Looking back at the case study, it was a mistake not to provide the participants with the bookmarklet.

- **Dynamically identify relevant domains.** The Chrome extension made use of a predefined set of domain names that were indexed such that irrelevant domain with respect to work the organisations were not indexed automatically. This list should be easier to modify or even be changed dynamically based on Artificial Intelligence. Now people had come to us to ask for additional domains to be added to the list, we had to perform an update of the Extension and then people had to accept the new rights required due to the update. In short, an inefficient process.
- **internet vs intranet search.** The prototype indexed both internal and external sources. Most participants stated that the prototype worked best for intranet search and worse for internet search. To keep the goal of the system clear, it could be smart to exclude external internet resources. This way, the goal and scope of the system would be clearer: enterprise search.
- **Personalisation and groups.** Some people described the system as a sort of shared bookmarks. They would like to also have an overview only their own bookmarks. They would also like to have sort of intermediate level between organisation wide ranking and personal ranking. Employees should be able to join groups, like their own department or their own function. Such groups of people should then be able to have their own set of resources. This could also involve tagging of resources with group names.
- **Fallback search methods.** It can be frustrating to search the prototype, mainly when searching on the web because not many resources are indexed compared to the size of the web. Therefore, there should be a fallback search method or several fallback search methods such that if the number of results provided by the system is limited, the user can, without much effort, pose the query to another search method or this should even be done automatically. This could be achieved with a button for each search method, or more sophisticated by appending for example Google Search results to the list of results. Another option would be to use an automatic redirect of the browser when there are no results found.
- **Additional information.** An info page explaining how the system works, what the goal of the system is and how it works would be useful according to several participants.

5 CONCLUSIONS

Consistently with research performed by Accenture, we found that there is a need for a Single Point of Access within organisations (Nanterme and Daugherty, 2014). Based on the performed case study we conclude that the proposed search method is a very good starting point for such a Single Point of Access. The alternative way of discovering and indexing web-pages has the advantage that integration with the existing software architecture is not necessary. Therefore, the search method can be implemented within an organisation faster and cheaper than existing Enterprise Search Engines.

REFERENCES

- Bagozzi, R. P. (2007). The legacy of the technology acceptance model and a proposal for a paradigm shift. *Journal of the association for information systems*, 8(4).
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, pages 319–340.
- D’souza, D. and Wills, A. C. (1998). *Objects, Components, and Frameworks with UML: The Catalysis(sm) Approach*. Addison-Wesley Professional Reading.
- Evans, B. M. and Chi, E. H. (2008). Towards a model of understanding social search. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 485–494. ACM.
- Golovchinsky, G., Pickens, J., and Back, M. (2009). A taxonomy of collaboration in online information seeking. *arXiv preprint arXiv:0908.0704*.
- Nanterme, P. and Daugherty, P. (2014). From digitally disrupted to digital disrupter. Technical report, Technology Labs, Accenture.
- Offsey, S. (1997). Knowledge management: linking people to knowledge for bottom line results. *Journal of Knowledge Management*, 1(2):113–122.
- Van Raaij, E. M. and Schepers, J. J. (2008). The acceptance and use of a virtual learning environment in china. *Computers & Education*, 50(3):838–852.
- Venkatesh, V. and Davis, F. D. (2000). A theoretical extension of the technology acceptance model: four longitudinal field studies. *Management science*, 46(2):186–204.
- Venkatesh, V., Morris, M. G., Davis, G. B., and Davis, F. D. (2003). User acceptance of information technology: Toward a unified view. *MIS quarterly*, 27(3).
- Yin, R. K. (2009). *Case study research: Design and methods*, volume 5. Sage.