| | |
|---|---|
| ***Title*** | Google shared. A case study in social search |
| ***Authors(s)*** | Briggs, Peter; Coyle, Maurice; O'Mahony, Michael P. |
| ***Publication date*** | 2009-07-10T09:25:48Z |
| ***Publication information*** | Houben, G.-J. ..[et al.] (eds.). User modeling, adaptation and personalization : 17th International Conference, UMAP 2009 : Proceedings |
| ***Conference details*** | UMAP - 1st and 17th International Conference on User Modeling, Adaptation, and Personalization, Trento, 22-26 June 2009 |
| ***Publisher*** | Springer |
| ***Link to online version*** | http://dx.doi.org/10.1007/978-3-642-02247-0_27 |
| ***Item record/more information*** | http://hdl.handle.net/10197/1244 |
| ***Publisher's version (DOI)*** | 10.1007/978-3-642-02247-0_27 |

# Google Shared. A Case-Study in Social Search*

Barry Smyth, Peter Briggs, Maurice Coyle, and Michael O'Mahony

CLARITY: Centre for Sensor Web Technologies
School of Computer Science and Informatics
University College Dublin, Ireland.
{firstname.lastname}@ucd.ie

**Abstract.** Web search is the dominant form of information access and everyday millions of searches are handled by mainstream search engines, but users still struggle to find what they are looking for, and there is much room for improvement. In this paper we describe a novel and practical approach to Web search that combines ideas from personalization and social networking to provide a more collaborative search experience. We described how this has been delivered by complementing, rather than competing with, mainstream search engines, which offers considerable business potential in a Google-dominated search marketplace.

## 1 Introduction

For all the success of mainstream Web search engines, users still struggle to find the right information quickly. Poor search productivity is largely a result of vague or ambiguous queries [6, 8, 20], and there is considerable research on different ways to improve result selection and ranking. For example, researchers have looked at ways to bias search towards special types of information (e.g., people, research papers, etc.); see for e.g. [9]. Others have attempted to profile the preferences of searchers in order to deliver more personalized result-rankings [10, 11, 21]. Recently, other researchers have explored how to take advantage of the collaborative nature of search [1, 12–14, 17]. In our own research we have explored a collaborative approach to personalized Web search [4,18,19], profiling the preferences of communities of users, rather than individuals, and generating recommendations inline with community preferences; see also [7].

While results have been promising, little attention has been paid to the issue of deployment and it is difficult to see how these technologies can be successfully brought to mainstream search. We have previously explored different deployment options [2, 5] as a way to loosely integrate community-based search with mainstream search engines. However it has been clear for some time that neither approach is likely to work for consumer Web search: users want to search as normal using their favourite search engine. However, the recent arrival of

* This work is supported by Science Foundation Ireland under grant 07/CE/I1147.

browser plugins has presented a new opportunity to deliver third-party search technology, via the browser, on top of some underlying service like Google.

This paper describes how this has been achieved through a new commercial venture called HeyStaks (*www.heystaks.com*). HeyStaks places an emphasis on the potential for collaboration within Web search as a route to a better search experience; see also [1, 12–14, 17]. The key motivating insight is that there are important features missing from mainstream search engines. For example, recent studies highlight that for 30% of searches the searcher is looking for something that they have previously found, yet search engines like Google offer no practical support to help users re-find information. Similarly, for up to 70% of searches the searcher is looking for something that has recently been found by a friend or colleague [19]. And, once again, search engines like Google offer no support for the sharing of search results. Helping searchers to organise and share their search experiences could deliver significant improvements in overall search productivity. We describe how HeyStaks adds these missing collaboration features to mainstream search engines and present results from a recent usage analysis based on the initial beta deployment of the system.

## 2 HeyStaks

HeyStaks adds two basic features to any mainstream search engine. First, it allows users to create *search staks*, as a type of folder for their search experiences at search time. Staks can be shared with others so that their searches will also be added to the stak. Second, HeyStaks uses staks to generate recommendations that are added to the underlying search results that come from the mainstream search engine. These recommendations are results that stak members have previously found to be relevant for similar queries and help the searcher to discover results that friends or colleagues have found interesting, results that may otherwise be buried deep within Google's default result-list.

As per Fig. 1, HeyStaks takes the form of two basic components: a client-side *browser toolbar* and a back-end *server*. The toolbar allows users to create and share staks and provides a range of ancillary services, such as the ability to tag or vote for pages. The toolbar also captures search result click-thrus and manages the integration of HeyStaks recommendations with the default result-list. The back-end server manages the individual stak indexes (indexing individual pages against query/tag terms and positive/negative votes), the stak database (stak titles, members, descriptions, status, etc.), the HeyStaks social networking service and, of course, the recommendation engine. In the following sections we will briefly outline the basic operation of HeyStaks and then focus on some of the detail behind the recommendation engine.

### 2.1 System Overview

Consider the following example. Steve, Bill and some friends were planning a European vacation and they knew that during the course of their research they

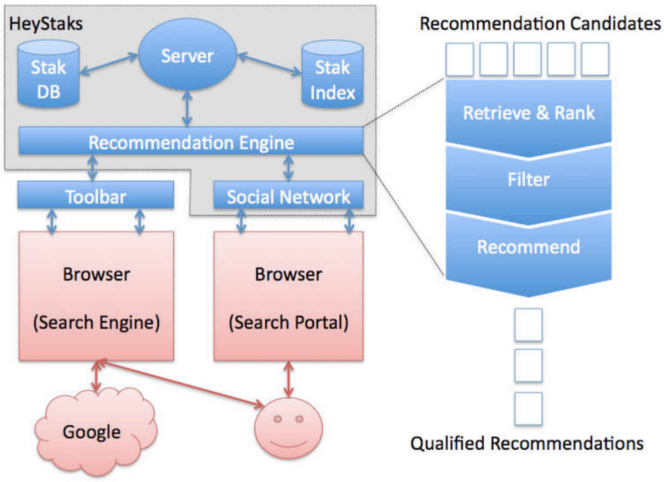


**Fig. 1.** The HeyStaks system architecture and outline recommendation model.

would use Web search as their primary source of information about what to do and where to visit. Steve created a (private) search stak called "European Vacation 2008" and shared this with Bill and friends, encouraging them to use this stak for their vacation-related searches.


**Fig. 2.** Selecting a new active stak.

Fig. 2 shows Steve selecting this stak as he embarks on a new search for "Dublin hotels", and Fig. 3 shows the results of this search. The usual Google results are shown, but in addition HeyStaks has made two promotions. These were promoted because other members of the "European Vacation 2008" stak had recently found these results to be relevant; perhaps they selected them for *similar* queries, or voted for them, or tagged them with related terms. These recommendations may have been promoted from much deeper within the Google result-list, or they may not even be present in Google's default results. Other relevant results may also be highlighted by HeyStaks, but left in their default Google position. In this way Steve and Bill benefit from promotions that are based on their previous similar searches. In addition, HeyStaks can recommend results from other related public staks as appropriate, helping searchers to benefit from the search knowledge that other groups and communities have created.

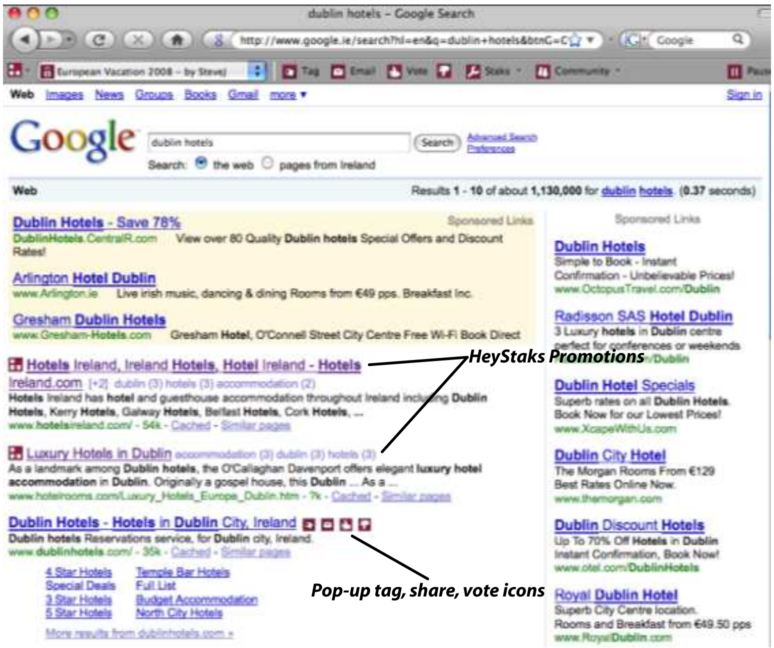


**Fig. 3.** Google search results with HeyStaks promotions.

Separately from the toolbar, HeyStaks users also benefit from the HeyStaks *search portal*, which provides a social networking service built around people's search histories. For example, Fig. 4 shows the portal page for the "European Vacation 2008" stak, which is available to all stak members. It presents an activity feed of recent search history and a query cloud that makes it easy for the user to find out about what others have been searching for. The search portal also provides users with a wide range of features such as stak maintenance (e.g., editing, moving, copying results in staks and between staks), various search and filtering tools, and a variety of features to manage their own search profiles and find new search partners.

### 2.2 The HeyStaks Recomendation Engine

In HeyStaks each search stak ($S$) serves as a profile of the search activities of the stak members and HeyStaks combines a number of implicit and explicit

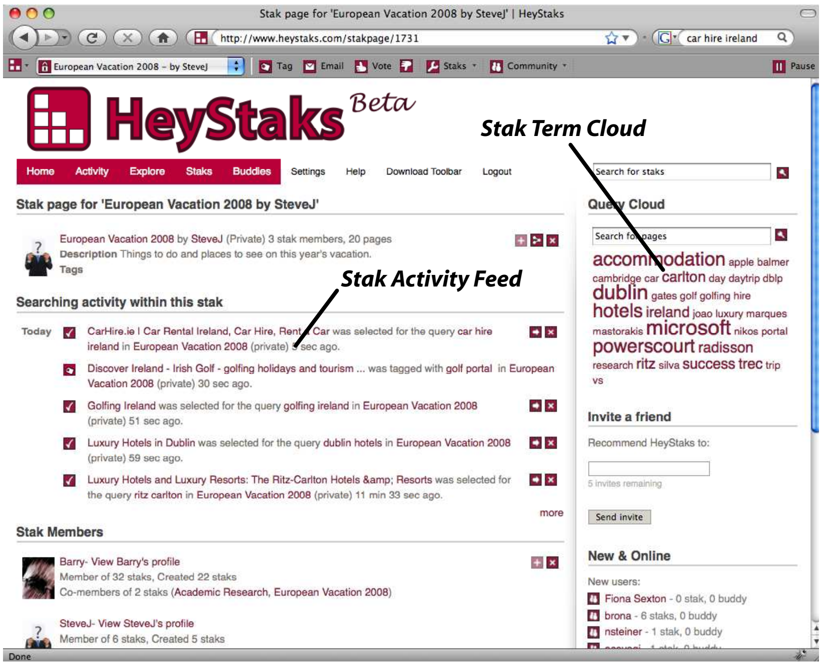


**Fig. 4.** The HeyStaks search portal provide direct access to staks and past searches.

profiling techniques to capture a rich history of search experiences. Each stak is made up of a set of result pages ($S = \{p_1, ..., p_k\}$) and each page is anonymously associated with a number of implicit and explicit interest indicators, including the total number of times a result has been selected ($sel$), the query terms ($q_1, ..., q_n$) that led to its selection, the number of times a result has been tagged ($tag$), the terms used to tag it ($t_1, ..., t_m$), the votes it has received ($v^+, v^-$), and the number of people it has been shared with ($share$) (all explicit indicators of interest) as indicated by Eq. 1.

$$p_i^S = \{q_1, ..., q_n, t_1, ..., t_m, v^+, v^-, sel, tag, share\} \tag{1}$$

In this way, each page is associated with a set of *term data* (query terms and/or tag terms) and a set of *usage data* (the selection, tag, share, and voting count). The term data is represented as a Lucene (*lucene.apache.org*) index table, with each page indexed under its associated query and tag terms, and provides the basis for retrieving and ranking *promotion candidates*. The usage data provides an additional source of evidence that can be used to filter results and to generate a final set of recommendations. At search time, a set of recommendations is produced in a number of stages: relevant results are retrieved and ranked from the Lucene stak index table; these promotion candidates are filtered based on an *evidence model* to eliminate noisy recommendations; and the remaining results are added to the Google result-list according to a set of *recommendation rules*.

**Retrieval & Ranking.** Briefly, there are two types of promotion candidates: *primary promotions* are results that come from the active stak $S_t$; whereas *secondary promotions* come from other staks in the searcher's stak-list. To generate these promotion candidates, the HeyStaks server uses the current query $q_t$ as a probe into each stak index, $S_i$, to identify a set of relevant stak pages $P(S_i, q_t)$. Each candidate page, $p$, is scored using Lucene's *TF*IDF* retrieval function as per Equation 2, which serves as the basis for an initial recommendation ranking.

$$score(q_t, p) = \sum_{t \epsilon q_t} tf(t \epsilon p) \bullet idf(t)^2 \tag{2}$$

**Evidence-Based Filtering.** Staks are inevitably noisy, in the sense that they will frequently contain pages that are not on topic. For example, searchers will often forget to set an appropriate stak at the start of a new search session, and although HeyStaks includes a number of automatic stak-selection techniques to ensure that the right stak is active for a given search, these techniques are not perfect, and misclassifications do inevitably occur. As a result, the retrieval and ranking stage may select pages that are not strictly relevant to the current query context. To avoid making spurious recommendations HeyStaks employs an *evidence filter*, which uses a variety of threshold models to evaluate the relevance of a particular result, in terms of its usage evidence; tagging evidence is considered more important than voting, which in turn is more important than implicit selection evidence. For example, pages that have only been selected once, by a single stak member, are not automatically considered for recommendation and, all other things being equal, will be filtered out at this stage. In turn, pages that have received a high proportion of negative votes will also be eliminated. The precise details of this model are beyond the scope of this paper but suffice it to say that any results which do not meet the necessary evidence thresholds are eliminated from further consideration.

**Recommendation Rules.** After evidence pruning we are left with revised primary and secondary promotions and the final task is to add these *qualified recommendations* to the Google result-list. HeyStaks uses a number of different recommendation rules to determine how and where a promotion should be added. Once again, space restrictions prevent a detailed account of this component but, for example, the top 3 primary promotions are always added to the top of the Google result-list and labelled using the HeyStaks promotion icons. If a remaining primary promotion is also in the default Google result-list then this is labeled in place. If there are still remaining primary promotions then these are added to the secondary promotion list, which is sorted according to TF*IDF scores. These recommendations are then added to the Google result-list as an optional, expandable list of recommendations.

## 3 Empirical User Studies

In this section we examine a subset of 95 HeyStaks users who have remained active during the course of the early beta release of the toolbar and service. These users registered with HeyStaks during the period October-December 2008 and the results below represent a summary of their usage during the period October 2008 - January 2009. Our aim is to gain an understanding of both how users are using HeyStaks, and whether they seem to be benefiting from its search promotions. Because this is a study of live-users *in the wild* there are certain limitations about what we have been able to measure. There is no control group, for example, and it was not feasible, mainly for data privacy reasons, to analyse the relative click-through behaviour of users, by comparing their selections of default Google results to their selections of HeyStaks promotions. However, for the interested reader, our earlier work does report on this type of analysis in more conventional control-group laboratory studies [3, 4, 19].

### 3.1 On the Creation and Sharing of Search Staks

Key to the HeyStaks proposition is that searchers need a better way to organise and share their search experiences. HeyStaks provides these features but do users actually take the time to create staks? Do they share them with others or join those created by others?

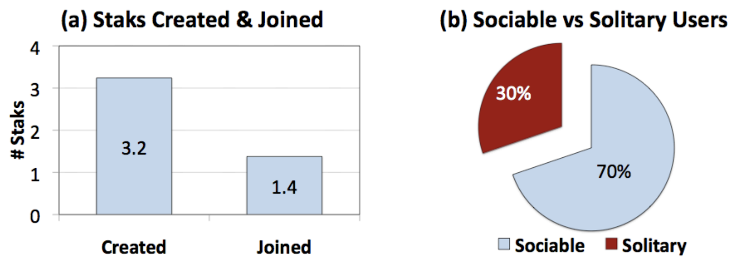


**Fig. 5.** (a) Average staks created and joined per user. (b) The percentage of *sociable* and *solitary* users.

During the course of the initial deployment of HeyStaks users did engage in a reasonable degree of stak creation and sharing activity. For example, as per Fig. 5, on average, beta users created just over 3.2 new staks and joined a further 1.4. Perhaps this is not surprising: most users create a few staks and share them with a small network of colleagues or friends, at least initially.

In total there were over 300 staks created on a wide range of topics, from broad topics such as travel, research, music and movies, to more niche interests including archaeology, black and white photography, and mountain biking. A few users were prolific stak creators and joiners: one user created 13 staks and joined another 11, to create a search network of 47 other searchers (users who co-shared the same staks). In fact on average, each user was connected to a search network of just over 5 other searchers by the staks that they shared.

The vast majority of staks were created as public staks, although most (52%) remained the domain of a single member, the stak creator. Thus 48% of staks were shared with at least one other user and, on average, these staks attracted 3.6 members. One way to look at this is as depicted in Fig. 5(b): 70% of users make the effort to share or join staks (*sociable* users); and only 30% of users created staks just for their own personal use and declined to join staks created by others (*solitary* users).

### 3.2 On the Social Life of Search

At its core HeyStaks is motivated by the idea that Web search is an inherently social or collaborative activity. And even though mainstream search engines do not support this, searchers do find alternative collaboration channels (e.g., email, IM, etc.) with which to partially, albeit inefficiently, share their search experiences. One of the most important early questions to ask about HeyStaks users concerns the extent to which their natural search activity serves to create a community of collaborating searchers. As users search, tag, and vote they are effectively producing and consuming community search knowledge. A user might be the first to select or tag a given result for a stak and, in this context, they have *produced* new search knowledge. Later, if this result is promoted to another user and then re-selected (or tagged or voted on), then this other user is said to have *consumed* that search knowledge; of course they have also produced search knowledge as their selection, tag, or vote is added to the stak.

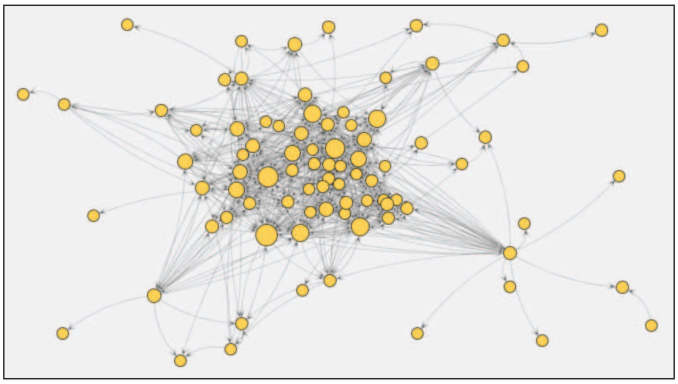

**Fig. 6.** A representation of the collaboration network among HeyStaks searchers.

These relationships between the producers and consumers of search knowledge within staks effectively creates an implicit social network of search collaboration. Fig. 6 presents a visualization of this network of the beta users. Each node is a unique user and edges between nodes correspond to evidence for search collaboration. These edges are directed: an edge from *user A* (the producer) to *user B* (the consumer) signifies that *user B* has selected at least one of the search results that *user A* has been responsible for adding (through his/her own selections, tagging or voting activity) to a search stak that is shared between both users. Of course a single edge can (and typically does) reflect many collaboration instances between two users. In this example the diameter of the nodes reflects the *reputation* of the user in terms of their relative ability to help other users to search; however a detailed discussion of this reputation mechanism is beyond the scope of this paper.

Perhaps the first thing to notice is the extent of the collaboration that is evident among these users. From Fig. 6 we can see that the sharing of search knowledge is not limited to to a small clique of especially social searchers. In fact, far from it, the graph includes 85% of beta users meaning that 85% of users have engaged in search collaborations. The majority have consumed results that were produced by at least one other user, and on average these users have consumed results from 7.45 other users. In contrast 50% of users have produced knowledge that has been consumed by at least one other user, and in this case each of these producers has created search knowledge that is consumed by more than 12 other users on average.

These production/consumption statistics can be contrasted with more conventional social media participation levels, where less than 10% of users actively engage in the production of information [15]. In HeyStaks, the implicit nature of search knowledge production means that 50% of users are effectively contributing to the search knowledge as a side effect of their normal search habits.

Moreover, these collaboration instances are far from being one-offs. As mentioned above each edge typically relates to multiple instances of collaboration. One particular user has been helped by 18 other users during 286 searches. Another user has produced search knowledge that 27 users have found to be useful during 499 different searches.

### 3.3 Producers and Consumers

These data speak to the potential for HeyStaks as a collaboration platform for Web search. Clearly HeyStaks is capturing and harnessing a significant amount of natural search collaboration. In this section we dig a little deeper in to the nature of this collaboration from the perspective of an individual searcher.

One question we might ask is to what extent individual users tend to be producers or consumers of search knowledge. Are some searchers *net producers* of search knowledge, in the sense that they are more inclined to create search knowledge that is useful to others? Are other users *net consumers*, in the sense that they are more inclined to consume search knowledge that others have created? This data is presented in Fig. 7(a). To be clear a net producer is defined as a user who has helped more other users than they themselves have been helped by, where as a net consumer is defined as a user who has been helped by more users than they themselves have helped. The chart shows that 47% of users are

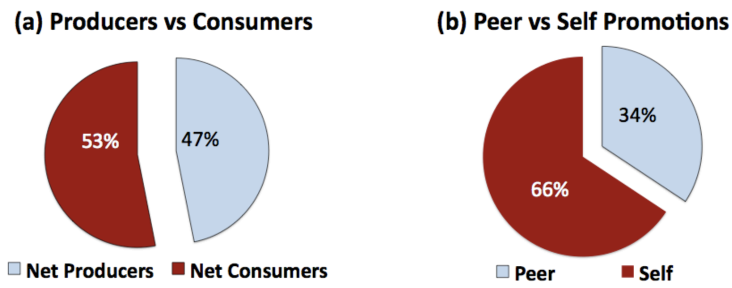


**Fig. 7.** (a) Average staks created and joined per user. (b) The percentage of *sociable* and *solitary* users.

net producers. Remember that, above, we noted how 50% of users have produced at least *some* search knowledge that has been consumed by some other user. It seems that the vast majority of *these* users, 94% of them in fact, are actually helping more people than they are helped by in return.

### 3.4 Peer vs Self Promotions

So, we have found that lots of users are helping other users, and lots of users are helped by other users. Perhaps this altruism is limited to a small number of searches? Perhaps, most of the time, at the level of individual searches, users are helping themselves? A variation on the above analysis can help shed light on this question by looking at the source of promotions that users judge to be relevant enough to select during their searches.

Overall, the beta users selected more than 11,000 promotions during their searches. Some of these promotions will have been derived from the searcher's own past history; we call these *self* promotions. Others will have been derived from the search activities of other users who co-share staks with the searcher; we call these *peer* promotions. The intuition here is that the selection of self promotions corresponds to examples of HeyStaks helping users to *recover* results they have previously found, whereas the selection of promotions from peers corresponds to *discovery* tasks, where the user is benefiting from focused new content that might otherwise have been missed, or have been difficult to find; see [16].

Fig. 7(b) compares the percentage of peer and self promotions and shows that two-thirds of selected promotions are generated from the searcher's own past search activities; most of the time HeyStaks is helping searchers to recover previously found results. However, 33% of the time peer promotions are selected (and we already know that these come from many different users), helping the searcher to discover new information that others have found.

The bias towards self promotions is perhaps not surprising, especially given the habits of searchers, and especially during the early stages of stak development. The growth of most staks is initially led by a single user, usually the creator, and so inevitably most of the promotions are generated in response to the creator's own search queries. And most of these promotions will be self promotions, derived from the leader's own search activities. Many staks are not shared and so are only capable of making self promotions. As staks are shared, however, and more users join, the pool of searchers becomes more diverse. More results are added by the actions of peers and more peer promotions are generated and selected. It is an interesting task for future work to explore the evolution of a search stak and to investigate how stak content and promotions are affected as more and more users participate. Are there well-defined stages in stak evolution, for example, as self promotions give way to peer promotions? For now it is satisfying to see that even in the early stages of stak evolution, where the average stak as between 3 and 4 members, that 34% of the time members are benefiting from promotions that are derived from the activities of their peers.

## 4 Conclusions

In the late 1990's the world of Web search was transformed by the idea of using connectivity information to rank search results, and within a few short years Google's PageRank had rendered purely term-based approaches obsolete. Today, Web search is the primary mode of information access but there is still considerable room for improvement. We believe that social (or collaborative) search techniques have the potential to have a similarly transformative impact on current Web search, and in this paper we have described the result of one research project in this area which has now matured in to a commercial venture.

HeyStaks is designed to work with mainstream search engines. Users search as normal but benefit from new collaboration features, allowing searchers to better organise and share their search experiences. Moreover, HeyStaks harnesses the product of search collaboration to generate result recommendations that offer more focused results than the underlying search engine. We have presented the results of a recent deployment that highlight how many early users have adapted well to the collaboration features offered by HeyStaks: most users create multiple search staks to store their search experiences and 70% of users share staks with others. In turn, collaboration has begun to pay dividends for early HeyStaks users: 85% of users have benefitted from the search experiences of others and, on average, 34% of the time users are seen to select promotions that have originated from their peers. Perhaps most surprising is the degree to which users are actively engaged in the production of useful search knowledge, which forms the basis of collaboration. Unlike other forms of social media, where a minority of users ($<$ 10%) participate in production, we have found that more than half of HeyStaks users are involved in the creation of useful search knowledge.

## References

1. Saleema Amershi and Meredith Ringel Morris. Cosearch: a system for co-located collaborative web search. In *CHI*, pages 1647–1656, 2008.

2. Evelyn Balfe and Barry Smyth. Case-based collaborative web search. In *ECCBR*, pages 489–503, 2004.
3. Oisín Boydell and Barry Smyth. Enhancing case-based, collaborative web search. In *ICCBR*, pages 329–343, 2007.
4. Maurice Coyle and Barry Smyth. Supporting intelligent web search. *ACM Trans. Internet Techn.*, 7(4), 2007.
5. Maurice Coyle and Barry Smyth. (web search)shared: Social aspects of a collaborative, community-based search network. In *AH*, pages 103–112, 2008.
6. George W. Furnas, Thomas K. Landauer, Louis M. Gomez, and Susan T. Dumais. The vocabulary problem in human-system communication. *Communications of the ACM*, 30(11):964–971, 1987.
7. Natalie S. Glance. Community Search Assistant. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 91–96. ACM Press, 2001.
8. S. Lawrence and C. Lee Giles. Accessibility of Information on the Web. *Nature*, 400(6740):107–109, 1999.
9. Steve Lawrence. Context in Web Search. *IEEE Data Engineering Bulletin*, 23(3):25–32, 2000.
10. Fang Liu, Clement Yu, and Weiyi Meng. Personalized Web Search for Improving Retrieval Effectiveness. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):28–40, 2004.
11. Alessandro Micarelli, Fabio Gasparetti, Filippo Sciarrone, and Susan Gauch. Personalized search on the world wide web. In Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors, *The Adaptive Web*, pages 195–230. Springer-Verlag, 2007.
12. Meredith Ringel Morris. A survey of collaborative web search practices. In *CHI*, pages 1657–1660, 2008.
13. Meredith Ringel Morris and Eric Horvitz. $S^3$: Storable, shareable search. In *INTERACT (1)*, pages 120–123, 2007.
14. Meredith Ringel Morris and Eric Horvitz. Searchtogether: an interface for collaborative web search. In *UIST*, pages 3–12, 2007.
15. Jakob Nielsen. Participation inequality: Lurkers vs. contributors in internet communities, 2006. http://www.useit.com/alertbox/participation_inequality.html.
16. Vicki L. O'Day and Robin Jeffries. Orienteering in an information landscape: how information seekers get from here to there. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '93)*, pages 438–445. ACM Press, 1993.
17. Jeremy Pickens, Gene Golovchinsky, Chirag Shah, Pernilla Qvarfordt, and Maribeth Back. Algorithmic mediation for collaborative exploratory search. In *SIGIR*, pages 315–322, 2008.
18. Barry Smyth. A community-based approach to personalizing web search. *IEEE Computer*, 40(8):42–50, 2007.
19. Barry Smyth, Evelyn Balfe, Jill Freyne, Peter Briggs, Maurice Coyle, and Oisín Boydell. Exploiting query repetition and regularity in an adaptive community-based web search engine. *User Model. User-Adapt. Interact.*, 14(5):383–423, 2004.
20. Amanda Spink, Dietmar Wolfram, Major B. J. Jansen, and Tefko Saracevic. Searching the web: the public and their queries. *Journal of the American Society for Information Science and Technology*, 52(3):226–234, 2001.
21. Jaime Teevan, Susan T. Dumais, and Eric Horvitz. Personalizing search via automated analysis of interests and activities. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 449–456. ACM Press, 2005.