Title	Collaboration and reputation in social web search
Authors(s)	McNally, Kevin; O'Mahony, Michael P.; Smyth, Barry; Coyle, Maurice; Briggs, Peter
Publication date	2010-09
Publication information	Geyer, W. et al. (eds.). Proceedings of the 2nd ACM RecSys'10 workshop on Recommender Systems and the Social Web
Conference details	2nd Workshop on Recommender Systems and the Social Web, in association with The 4th ACM Conference on Recommender Systems (RecSys 2010), Barcelona, Spain, September 26-30, 2010 2010
Link to online version	http://www.dcs.warwick.ac.uk/~ssanand/RSWeb_files/Proceedings_RSWEB-10.pdf
Item record/more information	http://hdl.handle.net/10197/2522

Collaboration and Reputation in Social Web Search

Kevin McNally, Michael P. O'Mahony, Barry Smyth, Maurice Coyle, Peter Briggs
CLARITY Centre for Sensor Web Technologies
School Of Computer Science & Informatics
University College Dublin
{firstname.lastname}@ucd.ie

ABSTRACT

Recent research has highlighted the inherently collaborative nature of many Web search tasks, even though collaborative searching is not supported by mainstream search engines. In this paper, we examine the activity of early adopters of HeyStaks, a collaborative Web search framework that is designed to complement mainstream search engines such as Google, Bing, and Yahoo. The utility allows users to search as normal, using their favourite search engine, while benefiting from a more collaborative and social search experience. HeyStaks supports searchers by harnessing the experiences of others, in order to enhance organic mainstream result-lists. We review some early evaluation results that speak to the practical benefits of search collaboration in the context of the recently proposed *Reader-to-Leader* social media analysis framework [11]. In addition, we explore the idea of utilising the reputation model introduced by McNally et al. [6] in order to identify the *search leaders* in HeyStaks, i.e. those users who are responsible for driving collaboration in the HeyStaks application.

Categories and Subject Descriptors

H.4.0 [**Information Systems Applications**]: General

General Terms

Algorithms, Experimentation, Security

Keywords

Collaborative Web Search, Social Recommender System, Reputation Model, HeyStaks

1. INTRODUCTION

The so-called *Social Web* paradigm serves to emphasise how the culture of the Internet has evolved far beyond simple information transfer, and how it is becoming an increasingly social and collaborative environment. The Internet today is a place where individuals learn about the views and opinions of others, a place where they can express their own opinions or rate content and services. They can even comment on the opinions of others, and otherwise share their views and observations about the world in which we live. And they can do this all within an eco-system of implicit and explicit communities, which harness groups, large and small, of like-minded users. Today billions of people participate in these types of social activities whether through large destination sites such as Facebook, MySpace, Twitter, Wikipedia, Flickr, or Amazon, or via the millions of blogs and discussion boards that cover every conceivable topic.

This level of information sharing and social activity has provided the raw material for a new form of online collaboration helping millions of users to make better decisions during their everyday lives. Amazon's user reviews have proved to be a vital source of trusted product information, for example, to help millions of users to make better purchases, whether through Amazon itself or elsewhere. Similarly, Twitter is now an important way for people to discover interesting Web pages with their friends and followers. Indeed the potential of the Internet as an open collaboration platform is exemplified by a new generation of so-called *social tools* that allow groups of users to collaborate on a wide range of common tasks, from document creation and editing (e.g. Google Docs, Write.ly, etc.), messaging (e.g. Twitter, Yammer etc.), data modeling and visualization (e.g. ManyEyes), knowledge sharing and conversation (e.g. Wikis and blogs) to the grand vision of Google Wave as a new platform for collaborative communication.

In all of this there is one aspect of Internet life that has yet to get the truly *collaborative* treatment, and that is Web search. After email, Web search is perhaps the most used Internet service as the leading search engines deal with literally billions of queries every day. However, until relatively recently Web search has largely been viewed as a solitary service in which individual users interact, in isolation, with their search engine of choice. All this is set to change, however, as researchers have begun to question the solitary nature of Web search, proposing a more collaborative search model in which groups of users can cooperate to search more effectively [3, 12, 13, 14, 15, 18]. Moreover recent work by [7] highlights the inherently collaborative nature of more general purpose Web search. Indeed, despite the absence of explicit collaboration features from mainstream search engines, there is clear evidence that users implicitly engage in many different forms of collaboration as they search, although, as reported by [7], these collabora-

ACM Conference on Recommender Systems 2010 Barcelona, Spain

tion "work-arounds" (email, instant messaging etc.) are often frustrating and inefficient. Naturally, this has motivated researchers to consider how different types of collaboration might be supported by future editions of search engines.

In this paper we consider the need for, and potential of, a more collaborative vision of Web search, one in which the appropriate experiences of relevant users can be harnessed in pursuit of a better, more productive, search experience. We consider the behaviour of users of *HeyStaks* (www.heystaks.com) which has been designed to provide just such a collaboration facility as an additional layer on top of mainstream search engines. Specifically, we examine the behaviour of a group of early adopters of the system, focusing on any collaboration they have taken part in. We consider user behaviour in terms of the Reader-to-Leader framework proposed in [11], which distinguishes between the users of a social system as a function of their level of interaction within that system. In addition, we show how a model of user reputation can be applied to discover search leaders, i.e. those users who drive search collaboration by contributing high quality search knowledge to their community.

2. A REVIEW OF COLLABORATIVE INFORMATION RETRIEVAL

Collaborative information retrieval research takes a fresh look at information retrieval and Web search, which highlights the potential for collaboration between searchers during extended search tasks. Recent work by [7] highlights the inherently collaborative nature of general purpose Web search. For example, during a survey of just over 200 respondents, clear evidence for collaborative search behaviour emerged. More than 90% of respondents indicated that they frequently engaged in collaboration at the level of the *search process*. For example, 87% of respondents exhibited "back-seat searching" behaviours, where they watched over the shoulder of the searcher to suggest alternative queries. A further 30% of respondents engaged in search coordination activities, by using instant messaging to coordinate searches. Furthermore, 96% of users exhibited collaboration at the level of *search products*, that is, the results of searches. For example, 86% of respondents shared the results they had found during searches with others by email. Almost 50% of respondents telephoned colleagues directly to share Web search results, while others prepared summary documents and/or Web pages in order to share results with others.

Thus, despite the absence of explicit collaboration features from mainstream search engines there is clear evidence that users implicitly engage in many different forms of collaboration as they search, although, as reported by [7], these collaboration "work-arounds" are often frustrating and inefficient. Naturally, this has motivated researchers to consider how different types of collaboration might be supported by future editions of search engines. The resulting approaches to *collaborative information retrieval* can be usefully distinguished along two important dimensions, namely *time* and *place*. In terms of the former, collaborative search systems can be designed to support *synchronous* or *asynchronous* collaborative search. And in terms of the latter, systems can be designed to support either *co-located* or *remote* forms of collaborative search.

Co-located systems offer a collaborative search experience for multiple searchers at a single location, often a single PC (e.g. [1]) or, more recently, by taking advantage of computing devices that are more naturally collaborative, such as table-top computing environments (e.g. [17]). In contrast, remote approaches allow searchers to perform their searches at different locations across multiple devices; see e.g. [8, 9, 21]. While co-located systems enjoy the obvious benefit of an increased faculty for direct collaboration that is enabled by the face-to-face nature of co-located search, remote services offer a greater opportunity for collaborative search.

Synchronous approaches are often characterised by systems that broadcast a "call to search" in which specific participants are requested to engage in a well-defined search task for a well defined period of time; see e.g. [16]. In contrast, asynchronous approaches are characterised by less well-defined, ad-hoc search tasks and provide for a more open-ended approach to collaboration in which different searchers contribute to an evolving search session over an extended period of time; see e.g. [8, 19]. In this paper we will focus on a community-based approach to collaborative Web search in which the *asynchronous* search experiences of communities of like-minded *remote* searchers are harnessed to provide an improved search experience that is more responsive to the learned preferences of a community of searchers.

In designing HeyStaks our primary goal is to provide social Web search enhancements, while at the same time allowing searchers to continue to use their favourite search engine. As such, a key component of the HeyStaks architecture is a browser toolbar that permits tight integration with search engines such as Google, allowing searchers to search as normal while providing a more collaborative search experience via targeted recommendations. We now briefly look at the HeyStaks architecture and recommendation engine, as well as how users interact with the system. A more detailed description of the HeyStaks architecture and recommendation engine is given by Smyth et al. [20].

2.1 The HeyStaks System Architecture

HeyStaks adds two important collaboration 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 own 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 either be buried deep within Google's default result-list or not present at all.

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.

HeyStaks users can create staks to search within, automatically storing results as they search. If others join the stak these results can be shared, and the new stak members can input their own shareable search knowledge. This can be particularly useful when a user wishes to harness the knowledge of a community bound together by a particular topic. Members of a community searching within a stak can input search results simply by selecting them, or by performing HeyStaks specific actions: Adding keywords to a page viewable by their community (*tagging*), voting on the page (*vote-up* if they like the page, or *vote-down* if they don't), or sharing pages directly with other HeyStaks users. These results are subsequently recommended to any member of the stak, if deemed relevant by the HeyStaks recommendation engine, appearing as an augmentation to a search engine's query result list,. This recommendation engine is discussed in the following section.

2.2 The HeyStaks Recomendation Engine

In HeyStaks each search stak (S) serves as a profile of the search activities of the stak members. 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 ($share$) as indicated by Eq. 1.

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

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, 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, recommendations are produced in a number of stages: first, relevant results are retrieved and ranked from the stak index; next, these promotion candidates are filtered based on the usage evidence to eliminate noisy recommendations; and, finally, 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 a *TF*IDF*-based retrieval function as per Equation 2, which serves as the basis for an initial recommendation ranking.

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

Evidence-Based Filtering. Staks are inevitably noisy, in the sense that they will frequently contain pages that are not on topic. 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. Further, pages that have received a high proportion of negative votes will be eliminated.

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 HeyStaks relevance values. These recommendations are then added to the Google result-list as an optional, expandable list of recommendations.

In summary, HeyStaks is designed to help users to collaborate during Web search tasks and, importantly, it succeeds in integrating collaborative recommendation techniques with mainstream search engines. In the next section, we turn our attention to an examination of the usage of the system and consider, for example, the types of users that are active within the system, what kind of search staks are created and the nature and extent of the collaboration that exists in the search activities that are performed by users in these staks.

3. EVALUATING SEARCH COLLABORATION IN HEYSTAKS

So far we have introduced HeyStaks as a social search utility that introduces a collaboration layer on top of mainstream search, one in which users are able to organise and share their search experiences in order to help each other to search more efficiently. Here, we describe a recent evaluation based on the current HeyStaks Beta system with a view to answering some important questions about how users are actually using the service:

- Do HeyStaks users actually create search staks and, if so, how many on average?
- When users create staks, are they *private* or *public* staks? The answer to this question speaks to the openness (or otherwise) of HeyStaks users and their willingness to share their search experiences with others.
- Do users actively share the staks that they create? And do users tend to join staks created by others?
- Are HeyStaks' users benefiting from search collaboration? Is there evidence that users are selecting promoted results that come from their staks?
- Is there any evidence that some users are better searchers than others? Do some users tend to be more successful

when it comes to contributing valuable search knowledge to staks, search knowledge that others tend to benefit from during subsequent searches?

3.1 Modelling User Behaviour

For the purpose of this evaluation it is useful to consider HeyStaks users and their activities in the light of recent studies of user behaviour and online services, including social media [2, 10, 4, 5, 11]. The literature contains a number of useful frameworks, for example, that describe how the behaviour of online users changes as users become more or less engaged in a particular online or social media service. For example, Porter's *Funnel Model* distinguishes between four different types of user behaviour including, *interested*, *first-time use*, *regular use*, *passionate use*, and the rapid fall-off in participation that tends to occur at each stage; see [10] and [4] for a related model. Most recently, Preece and Shneiderman [11] have proposed their *Reader-to-Leader* framework that distinguishes between casual social media participation and more active engagement by proposing four classes of users:

1. *Readers* are users who consume the social media created by others, by and large without actively contributing themselves. For example, large numbers of users visit discussion boards, read blogs, and refer to Wikipedia without posting content or commenting themselves.

2. *Contributors* are users who do make a meaningful contribution to the evolving social media usually by rating or commenting on content that others have created. Contributors are defined by a tendency to follow the lead of other more active users. For example, in Wikipedia a contributor is someone who tends to edit existing pages rather than create new pages of content.

3. *Collaborators* are users who engage in more deliberate activities that serve to elicit some form of communal response, often in the form of an identifiable episode of collaboration between at least two users. These collaborations can be light-weight and short-lived (e.g. two users engaging in conversation via a thread of comments on a particular blog-post) or they can be longer-lasting engagements with a wider community of users (e.g. connecting with large numbers of people on social network websites like Last.fm and Facebook).

4. *Leaders* are the synthesizers of the social media space. They tend to be those users who contribute the most to a particular service, provide the most comments, the most ratings or the most blog posts, but they typically also go further by creating the conditions for new discussions to develop. The popular bloggers are leaders as they catalyze a community response to their views and opinions on a particular topic or theme, for example.

Of course these user categories do not define crisp, mutually exclusive subsets of users and, as discussed in [11], in many cases users will switch between being contributors and collaborators or between being readers and contributors, for example. Nevertheless this framework is a useful starting point to understand the different types of user engagement that can exist in a social service such as HeyStaks.

3.2 From Readers to Leaders in HeyStaks

In our evaluation we consider the usage data associated with 299 active HeyStaks users who have joined the Beta service during 2009. While detailed demographic data is unavailable for these users, the Beta invitations were largely circulated among our local researchers and their friends and so it is likely that many of these users are college students or recent graduates. In total, the data set covers 99,097 *activity records* covering all aspects of HeyStaks activity (stak creation, sharing, joining, search, result selection, result tagging and voting etc.).

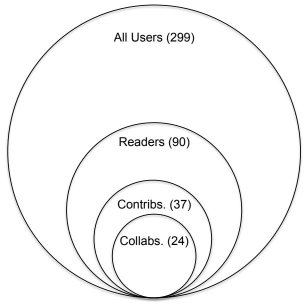


Figure 1: The HeyStaks user-base.

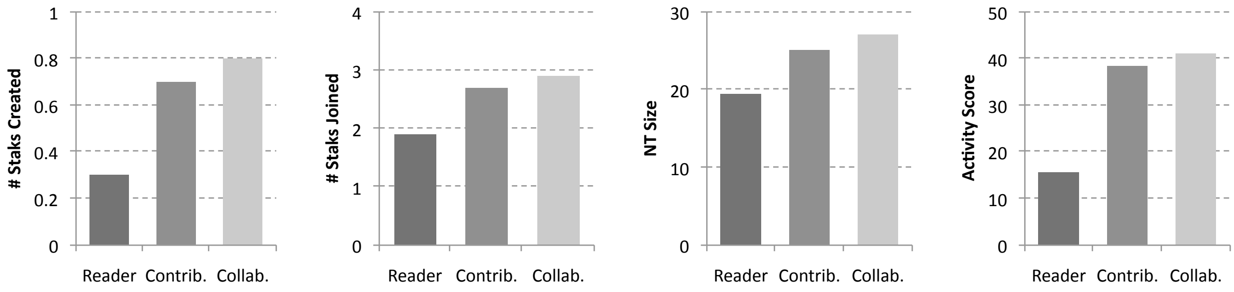
We apply the *Reader-to-Leader* framework as described above to analyse users and their activities in HeyStaks. Moreover, we introduce an additional superset class of users which we label *All Users*; this is necessary given the Beta nature of the HeyStaks deployment. A description of this user class and the other framework classes in the context of the HeyStaks domain is as follows:

- *All Users*: This is our group of 299 test users, each of whom have engaged in some minimal level of HeyStaks activity. Specifically, this is activity beyond creation of a *My Searches* stak and joining one other stak. Importantly, many of these users will not have engaged in any social activity, choosing not to share their staks or join staks created by others.

- *Readers*: An important criterion in HeyStaks is whether or not users actively engage in the sharing of search experience and in this context *readers* are defined to be those users that have displayed at least some form of sharing behaviour. Specifically we define an *active-shared stak* to be a stak that is shared with at least 2 users, containing at least 10 pages. Then a user is considered to be a *reader* if they are a member of at least one active-shared stak.

- *Contributors*: These are users who have added content to an active-shared stak. Typically, they will have selected or tagged or voted on a new result or page which then gets added to the stak and is available for future promotion/recommendation.

- *Collaborators*: In HeyStaks, the selection of a *recommended result* is the basic unit of search collaboration. Users who contribute new content to a stak, which eventually gets recommended to another stak member, and which is ultimately selected (or tagged or shared)



(a) Staks created. (b) Staks joined. (c) Network size. (d) Activity scores.

Figure 2: Average number of staks created and joined, network size and activity score across readers, collaborators and contributors.

by this other stak member, are considered to be *collaborators.*

- *Leaders*: In general, leaders are those users who are responsible for driving search collaboration within the HeyStaks application. Thus, leaders will be drawn from the set of collaborators and we examine to what extent each of these users contribute to the collaborations that take place in the system. We explore leadership in detail in Section 5 where, for example, we introduce a model of *user reputation* which we believe to be a useful indicator of search leadership.

In the next section, we analyse activity in HeyStaks by examining user behaviour as a function of class membership and the various types of staks that users have created and joined. We examine the level of activity that users from different classes are engaged in and consider the benefit accruing to users as a result of stak membership through the introduction of *stak reuse coefficients*, which we discuss in Section 4. Finally, in Section 5 we discuss leadership in the context of HeyStaks and propose a reputation model to identify search leaders in the user community.

4. EVALUATION AND ANALYSIS

The primary goal of this paper is to analyse the behaviour of those users who are the chief collaborators within HeyStaks - our *search leaders*. We begin by analysing the activity of our 299 early adopters of the HeyStaks system, at both the user and stak-level, and categorise these users according to the Reader-to-Leader model.

4.1 Analysis of User Behaviour

Figure 1 presents the breakdown of the 299 users in terms of the main reader-to-leader user categories. The various classes of users (readers, leaders etc.) are not disjoint, i.e. leaders are a subset of collaborators, collaborators are a subset of contributors etc. Currently about 70% of the 299 users are yet to engage in the social-side of HeyStaks search: they have not shared or joined staks and so have not yet enjoyed any type of collaboration benefit. In contrast, 90 users are classified as readers. In turn, about 40% of readers (37 users) are contributors: as a result of their search actions, new content has been added to shared staks as a precursor to collaboration. And about 70% of these contributors (24 users) are classified as collaborators; that is, they have played a role in the recommendation of content that was subsequently selected by some other user.

Users have partaken in a total of 99,097 activities across all staks. The vast majority of these actions, over 95%, are result selections, a figure reflecting the fact that selection is the most natural type of search activity. An important issue to consider is the extent to which users engage in the sort of activities that ultimately facilitate search collaboration. Do they create and join staks, for example? How many other users are they connected with (their *network size*)? All other things being equal, if a user creates and joins many staks then they are more likely to be connected with other users and will thus benefit from a larger collaboration network to act as a source of recommendations.

Figures 2(a) to 2(d) show the mean user values of key engagement metrics across *active-shared* staks, in terms of staks created and shared, network size and activity levels for the main user groups (readers, contributors and collaborators). Clearly as users transition from reader to contributor to collaborator we see a consistent increase in engagement level across all indicators. For example, collaborators on average create (0.8) and join (2.9) more staks than either contributors (0.7 and 2.7 respectively) and readers (0.3 and 1.9 respectively). Collaborators tend to have the largest number of users with which they can share search results: for example, network size is large across this user group, 27.1 on average, compared to an average network size of 19.4 for readers. Increasing engagement across user class is reflected in Figure 2(d), showing mean total number of selects, tags, votes and shares across each user category, with collaborators registering the highest average score.

4.2 Analysis of Active Staks

We now turn our attention to analysing stak membership across the various classes of users. In HeyStaks, there are two types of stak, i.e. either private or public, and each of these stak types can be shared or unshared. We look at various activity metrics within these different stak types to gain greater insight into user behaviour.

Figure 3 shows mean scores with respect to number of members, distinct results and activities within private and public instances of unshared and shared staks. Users tend to input a greater amount of search results into private rather than public staks. This may seem to indicate that people have a greater wish to retain their own results rather than share with others. However, it is the *private-shared* staks which show the greatest amount of activity, with more distinct pages (71.2) and activities (85.6) on average than the other stak types. This, coupled with the fact that public-shared staks are the more active public staks, indicates the desire of users' to share their search results with members of the HeyStaks community.

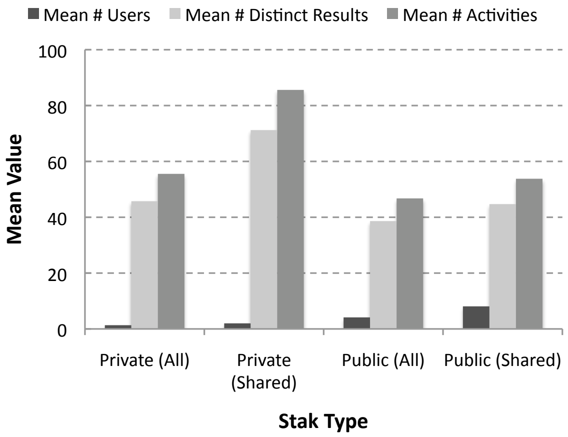


Figure 3: Summary statistics for active staks.

The above analysis shows the levels of activity within the various stak types across all user classes; we now examine the type of staks that the different classes of users tend to create and join. Results are presented in Figure 4 for readers, contributors and collaborators. It is clear from the figure that all users are members of more public staks than private staks. Moreover, of the 3.3 public staks of which contributors and collaborators are members, some 2.9 (87%) of these on average are shared. In contrast, of the 1.3 private staks that these users belong to, less than 20% are shared. Thus, these findings show that the stak type most favoured by all classes of users is the shared public stak, indicating that users have a strong preference to belong to staks where they can benefit from the search activity of other users.

4.3 Stak Reuse Coefficients

In the previous section, various statistics relating to stak activity were discussed. A key objective of HeyStaks is that users who have created or joined staks benefit from the previous search activity that they themselves and/or other members have performed in the context of these staks. Thus, an important measure of stak utility is to consider the number of times that stak members have selected promoted results relative to the number of organic result selections made in the stak.

Accordingly, we define the *reuse coefficient* for each stak as follows. Let n_p and n_o be the numbers of promoted and organic result selections made in stak S_i, respectively; thus the reuse coefficient, C_{S_i}, for the stak is given by:

$$C_{S_i} = \frac{n_p}{n_o} \tag{3}$$

With this approach, the effectiveness of staks in assisting users to locate search results can be readily assessed. For example, a reuse coefficient of 0 indicates that no promoted

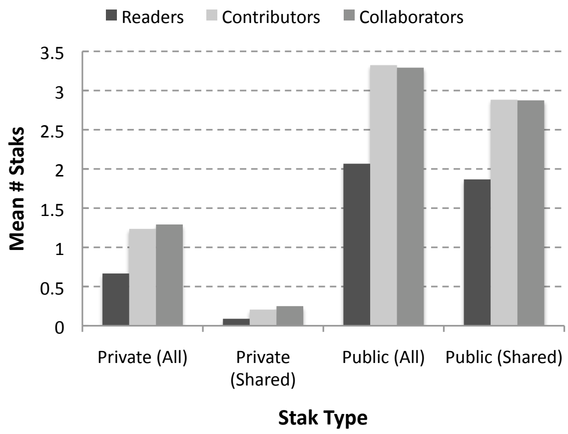


Figure 4: Mean number of staks that readers, contributors and collaborators are members of versus stak type.

results were selected in a stak, while a coefficient of 1 indicates that as many promoted results were selected as organic results. In general, higher reuse coefficients indicate that users benefit to a greater degree from the previous searches performed by themselves or by other stak members.

Figure 5 shows the mean stak reuse coefficients versus stak type for readers, contributors and collaborators. It is clear from the results that reuse coefficients are significantly greater for public staks. In the case of collaborators, for example, a mean reuse coefficient of 0.88 applies for public staks compared to only 0.07 for private staks. Similar differences are seen for the other user classes. Thus, the benefit to users of public stak membership is more than an order of magnitude greater than private stak membership, with users being 10 times more likely to benefit from useful result promotions in public staks.

Further, while contributors gain greater benefits in terms of reuse than readers, there is an additional benefit seen for collaborators. For example, in public staks, the reuse coefficients are 0.88, 0.65 and 0.25 for collaborators, contributors and readers respectively. These results clearly indicate the advantages of increased engagement with the system, with collaborators selecting on average over 1.3 times the number of promoted results than contributors and more than 3 times the number of promoted results than readers.

5. TOWARDS SEARCH LEADERS

Thus far, we have examined various summary statistics and reuse coefficients for the reader, contributor and collaborator classes of users, where each class is defined by its level of engagement with the system. We now turn our attention to the search *leaders* in the community. There are many ways in which leaders could be identified and defined; for example, those users who have created and joined the most staks or those who have added the most results to staks. In general, however, a high degree of activity does not guarantee that users play a *productive* role in the system. For example, a particular user may add many results to staks but few of these may be selected when promoted to other users, implying that these results are not considered to be

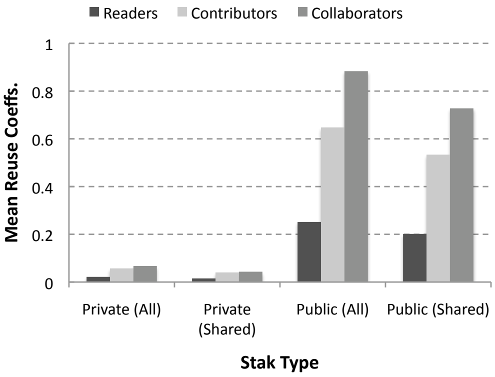


Figure 5: Mean stak reuse coefficients versus stak type for readers, contributors and collaborators.

Input: Set $\mathcal{A}$ of user activity tuples $\langle u, p, t, S, type\rangle$, set S of all staks, array $\mathcal{R}$ of user reputation scores
Output: Updated array $\mathcal{R}$ of user reputation scores

```
1.  UserReputation(𝒜, S, ℛ)
2.  begin
3.      foreach activity a ∈ 𝒜
4.          if a.type = promotion
5.              u_c ← a.u
6.              p_c ← a.p
7.              t_c ← a.t
8.              S_c ← staks(S, u_c)
9.              A ← {a′ ∈ 𝒜 : a′.p = p_c & a′.S ∈ S_c & a′.t < t_c}
10.             U ← {a′.u : a′ ∈ A} – {u_c}
11.
12.             foreach u ∈ U
13.                 ℛ[u] ← ℛ[u] + 1/|U|
14.             end
15.         end
16.     end
17. end
```

Figure 6: User reputation algorithm.

useful or relevant by other users of the system. Thus, in the next section, we explore leadership in the context of *user reputation*, in which those users that contribute the most from a search collaboration perspective can be identified.

5.1 From Collaboration to Reputation

In relation to the Reader-to-Leader framework, the leaders in the community will be drawn from the collaborator class of users, given that these users have the highest level of engagement with the system. Crucially, collaborators can be said to be productive users in the sense that they are responsible for *collaboration events*, i.e. they have added results to staks which have been promoted to and selected by other users of the system.

Our proposed user reputation algorithm [6] is given in Figure 6. For simplicity, the algorithm shown is one suitable for offline execution, but note that the algorithm can be readily modified such that user reputation scores are updated in real time when new activities are performed by users.

The algorithm takes as input a temporally ordered set of user activities $\mathcal{A}$ which are retrieved from the HeyStaks database. Each entry $a \in \mathcal{A}$ is a tuple $\langle u, p, t, S, type\rangle$, where $a.u$ is the user who performed the activity, $a.p$ is the associated result page, $a.t$ is the time when the activity occurred, $a.S$ is the active stak at the time of the activity and $a.type$ indicates whether or not the activity relates to a HeyStaks promotion. In addition, the set of all staks $\mathcal{S}$ and the current (previously calculated) set of user reputation scores $\mathcal{R}$ are provided as a starting point.

Briefly, the algorithm operates as follows. For each promotion activity $a \in \mathcal{A}$ (line 3), the set of staks S_c that the current user u_c is a member of is retrieved (line 8). Then, the set of prior activities relating to the current page p_c, in any of the staks in S_c, is determined (line 9) and the users who performed these activities are identified (line 10). Finally, a unit of reputation is distributed equally among these users and added to their existing reputation score (lines 12–14). This process continues until all activities are processed and the array $\mathcal{R}$, which contains each user's updated reputation scores, is returned. Thus, early producers of results, i.e. those who are among the first to add results to staks, benefit the most in terms of reputation when these results are subsequently selected by other users.

5.2 Results

The results of applying the reputation model to the 24 collaborators are shown in Figure 7. The trend is long-tailed, with 6 users achieving a reputation score of greater than 5, and with two users achieving a score of more than 10. In general, this is the kind of trend that is to be expected from a user reputation perspective, where a small subset of users contributes the most in terms of driving search collaboration, and the remaining users contributing some, but significantly less, search knowledge to the community.

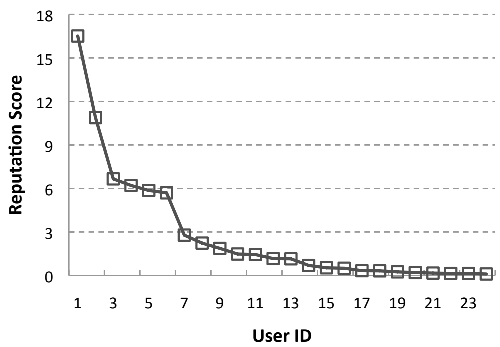


Figure 7: Reputation scores for 24 collaborators.

Given the trend observed in Figure 7, we can reasonably define search leaders as those users at the head of the reputation curve, with the knee-point acting as the cutoff point. Thus, given the set of HeyStaks users and corresponding activities analysed in this paper, we can identify 6 search leaders from the set of 24 collaborators. Note that, with this approach, leadership is a function of usage data and those users that are identified as search leaders at particular points in time can change as other (or new) users increase their level of activity within the system.

There are a number of improvements that could be made to the reputation model as described here. The current model does not reward those users who add results to staks that are subsequently promoted to and selected by many distinct users in the system. Further, additional reputation could be awarded when promoted results are selected from public shared staks, given that private shared staks are typically limited in membership to small ‘cliques’ of users. In future work, as the deployment of HeyStaks continues to expand and further usage data becomes available, we will explore the benefits of developing and applying more advanced user reputation models to identify search leaders within the system, and to deal with any attempts to game the system in order to manipulate reputation scores.

6. CONCLUSIONS

Although there is much evidence that many search tasks are inherently collaborative, mainstream search engines do not explicitly support collaboration during search. The main contribution of this paper is to analyse the activities of users of HeyStaks, a novel social search utility. In this regard, we have considered users in the context of the *Reader-to-Leader* social activity framework as proposed in [11]. We have examined the various stak types that users tend to create (e.g. public versus private staks), the degree to which staks are shared between users, and the benefits of stak membership through the introduction of stak reuse coefficients.

Our findings indicate that, for all classes of users, membership of shared public staks is by far the most popular choice, in which users are able to benefit from the previous search activity of other users. In addition, the results show that the greatest benefit in terms of promoted result selections applies to collaborators and leaders, the most engaged of the classes of users examined. These findings are promising from the point of view that the system rewards users with more benefit (i.e. useful result recommendations) as they increase their level of activity within the system.

In addition, we have described a user reputation model that is designed to identify the search leaders in the system; those users who contribute the most from a result collaboration perspective. While our model provides a more sophisticated approach to detecting leaders over more simple collaboration count approaches, we note that HeyStaks is in Beta deployment and hence our analysis is therefore performed on relatively small amounts of user activity data. In future, as HeyStaks gains traction in the wider community, further opportunity will exist for the analysis of user activity, stak creation, result sharing, and the development of more advanced reputation models to more fully understand the nature and benefits to users of the collaborative Web search approach as adopted in the HeyStaks application.

7. ACKNOWLEDGMENTS

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

8. REFERENCES

[1] S. Amershi and M. R. Morris. Cosearch: a system for co-located collaborative web search. In *CHI*, pages 1647–1656, 2008.

[2] B. M. Evans and E. H. Chi. An elaborated model of social search. *Information Processing and Management*, 2009.

[3] D. Horowitz and S. Kamvar. The Anatomy of a Large-Scale Social Search Engine. In *WWW*, 2010.

[4] A. J. Kim. *Community Building on the Web: Secret Strategies for Successful Online Communities.* Peachpit Press, 2000.

[5] C. Li and J. Bernoff. *Groundswell: Winning in a World Transformed by Social Technologies.* Harvard Business Review, 2008.

[6] K. McNally, M. P. O’Mahony, B. Smyth, M. Coyle, and P. Briggs. Towards a reputation-based model of social web search. In *IUI*, pages 179–188, 2010.

[7] M. R. Morris. A survey of collaborative web search practices. In *CHI*, pages 1657–1660, 2008.

[8] M. R. Morris and E. Horvitz. S^3: Storable, shareable search. In *INTERACT (1)*, pages 120–123, 2007.

[9] M. R. Morris and E. Horvitz. Searchtogether: an interface for collaborative web search. In *UIST*, pages 3–12, 2007.

[10] J. Porter. *Designing for the Social Web.* New Riders, 2008.

[11] J. Preece and B. Shneiderman. The reader to leader framework: Motivating technology-mediated social participation. *AIS Trans. on Human-Computer Interaction*, 1(1):13–32, 2009.

[12] M. C. Reddy and P. Dourish. A finger on the pulse: temporal rhythms and information seeking in medical work. In *CSCW*, pages 344–353, 2002.

[13] M. C. Reddy, P. Dourish, and W. Pratt. Coordinating heterogeneous work: Information and representation in medical care. In *ECSCW*, pages 239–258, 2001.

[14] M. C. Reddy and B. J. Jansen. A model for understanding collaborative information behavior in context: A study of two healthcare teams. *Inf. Process. Manage.*, 44(1):256–273, 2008.

[15] M. C. Reddy and P. R. Spence. Collaborative information seeking: A field study of a multidisciplinary patient care team. *Inf. Process. Manage.*, 44(1):242–255, 2008.

[16] A. F. Smeaton, C. Foley, D. Byrne, and G. J. F. Jones. ibingo mobile collaborative search. In *CIVR*, pages 547–548, 2008.

[17] A. F. Smeaton, H. Lee, C. Foley, and S. McGivney. Collaborative video searching on a tabletop. *Multimedia Syst.*, 12(4-5):375–391, 2007.

[18] B. Smyth. A community-based approach to personalizing web search. *IEEE Computer*, 40(8):42–50, 2007.

[19] B. Smyth, E. Balfe, J. Freyne, P. Briggs, M. Coyle, and O. Boydell. Exploiting query repetition and regularity in an adaptive community-based web search engine. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research*, 14(5):383–423, 2004.

[20] B. Smyth, P. Briggs, M. Coyle, and M. P. O’Mahony. A case-based perspective on social web search. In *ICCBR*, pages 494–508, 2009.

[21] B. Smyth, P. Briggs, M. Coyle, and M. P. O’Mahony. Google? shared! a case-study in social search. In *UMAP*, number 283-294. Springer-Verlag, June 2009.