

# Provenance, Trust, and Sharing in Peer-to-Peer Case-Based Web Search\*

Peter Briggs and Barry Smyth

Adaptive Information Cluster,  
School of Computer Science & Informatics,  
University College Dublin, Ireland  
{Peter.Briggs,Barry.Smyth}@ucd.ie

**Abstract.** Despite the success of modern Web search engines, challenges remain when it comes to providing people with access to the right information at the right time. In this paper, we describe how a novel combination of case-based reasoning, Web search, and peer-to-peer networking can be used to develop a platform for personalized Web search. This novel approach benefits from better result quality and improved robustness against search spam, while offering an increased level of privacy to the individual user.

## 1 Introduction

Web search is one of the most important technologies in regular use, providing literally billions of users with access to online content every day; search activity reached more than 60 billion searches per month in 2007 [1]. However, Web search is far from perfect, and recent studies have highlighted the extent to which leading search engines struggle to provide users with relevant results. For example, Smyth et al. [2] describe how as many as 56% of Google Web searches fail to attract any result selections. Over the past few years, as “the business of search” has matured in to a major market sector, researchers have continued to look for new ways to enhance existing search engine technology. In this regard the idea of “social search” — that result-lists might usefully be influenced by the interests, preferences, or activities of other searchers — has gained some considerable attention as a way to improve search quality by personalizing result-lists.

Harnessing the search activities of users to improve result quality is a challenging task, but one that has benefited from a case-based perspective. The *collaborative Web search* (CWS) work of Balfe & Smyth [3] demonstrates how the search experiences (queries and result selections) of communities of like-minded users can be stored as search case bases and used as a source of result recommendations (*promotions* during future searches); in short, for a new target query, results that have been frequently selected by community members for similar queries in the past are promoted during the new search.

---

\* This research was supported by Science Foundation Ireland Grant No. 03/IN.3/I361.

There are limitations to this standard approach to collaborative Web search. First, it relies on some explicit representation of a search community, with individual users expected to register and search within specific communities. The reality, of course, is that users simply want to search, and may not find it convenient to select a community context beforehand. Another limitation is that individual community members cannot be identified. In fact this is often cited as a privacy benefit, but in truth it has a downside when it comes to auditing the source of a promoted result. As the seminal work of Leake & Whitehead highlights, the origin, or *provenance*, of a case can be an important quality indicator [4]. This is especially true in CWS because it is possible for malicious users to influence result promotions [5]. By recording the source of a promotion (the searcher who originally selected the result), it is possible to present provenance information alongside the promoted result as a form of explanation. But, this is only possible if individual users can be distinguished within a community.

In this paper we present an alternative model of collaborative Web search; one that avoids the need for explicit communities, and which facilitates the identification of individual searchers to determine the provenance of promotions. We demonstrate how provenance information can be used to enhance the conventional CWS interface, and show how it can help to improve the quality of results in two important ways. Firstly, such information can be used as the basis for a computational model of user-trust, which we can apply to filter result promotions. Secondly, we argue that exposing the provenance of promotions through the search interface encourages the formation of social relationships between searchers, helping them to avoid making spurious result selections. Furthermore, we explain how the advantages of this trust-based approach can be achieved while preserving the privacy of individual searchers by implementing a distributed peer-to-peer search network. In this network, the search histories of individuals are maintained by their own local search-agent and only shared on the basis of trusted relationships between search peers. As an added benefit, we explain how this peer-to-peer architecture facilitates a more flexible approach to CWS by doing away with the need for explicit communities; essentially, an individual's search community evolves as they develop implicit relationships with other online searchers via the sharing and promotion of search experiences.

## 2 Background

This paper focuses on the personalization of search results, and, to this end, there is a growing body of literature covering the many ways in which individual and community preferences can be used to influence search. For example, the SOAP system [6] builds user profiles from bookmark collections and employs a collaborative filtering approach to result recommendation. Alternatively, Glance [7] describes a community search assistant which recommends similar (based on result overlap) queries from the previous searches of other community members.

In this paper we adopt an *experience-based* approach to personalization by harnessing the previous search sessions of searchers. This technique is naturally

inspired by case-based reasoning, which of course emphasises the power of experience reuse in problem solving. Interestingly, work by [8] adopts a complementary perspective. In brief, searchers on the University of Oregon's library website are encouraged to supplement their queries with natural language questions that describe their information needs. New target queries are matched against any past questions that have led to result selections, and the matching questions are submitted alongside the user's actual query to identify additional results that may be of value to the user; see also the work of [2] for a similar approach.

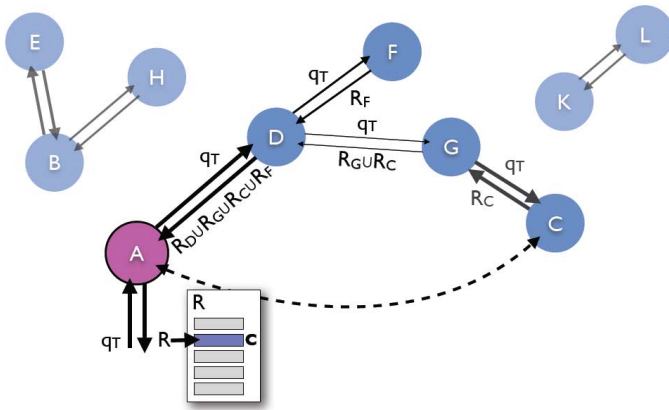
If experience-reuse is an important feature of this work, then the idea of *experience sharing* is equally vital. It is interesting to reflect on recent Web developments that have emphasised the value of cooperation and sharing between users. The so-called Web 2.0 movement is based on a more flexible model of user cooperation and information sharing, and these ideas have helped to inspire and inform our own approach to Web search which, in this paper, is based very much on the free exchange of search experiences between searchers. Our experience-based approach relies on the idea that each user is associated with a case base that reflects their own past search experiences (queries submitted and results selected). As searches unfold, result recommendations are also harvested from the case bases of other, potentially numerous, users who are similar to the target user. This too echoes recent work by the CBR community on the use of multiple case bases during problem solving, where the benefits of such multiple sources of problem solving experiences have been convincingly demonstrated [9].

Other recent work in case-based reasoning has begun to explore how understanding the origins of cases is important when it comes to guiding their future reuse. In particular, the work of [4] argues for the storage and reuse of *provenance* information — information about where a case has come from or who provided the case, for example — in CBR systems as a way to improve problem solving performance and solution quality, especially where case learning is actively employed. This research has helped to clarify the importance of provenance-type information in our own work: given that search recommendations can come from the search experiences of other users, it is important to understand who these users are and how reliable their recommendations are likely to be. To this end, we use provenance information during search to advise the searcher about the source of a recommendation, but also as the basis for a computational model of trust that is used to filter out recommendations from unreliable searchers.

Finally, it is worth commenting briefly on research related to our use of a peer-to-peer search network. Peer-to-peer networks are not uncommon in Web search (see, for example, [10,11]), but in the main they have been used to distribute the computational load associated with search, with individual peer nodes storing and indexing a sub-set of the document collection. In our work, the core search functionality is provided by a traditional search engine, such as Google, and the peer-to-peer network is used as an experience overlay for the purpose of adapting traditional search results according to the past experiences of like-minded users.

### 3 Peer-to-Peer Collaborative Web Search

A peer-to-peer approach to collaborative Web search (P2P-CWS) envisages an overlay network of search agents, each capturing the search experiences of a user,  $U$ . This so-called *search network* facilitates the sharing of search experiences between agents. When a given user performs a search, their query ( $q_T$ ) is used to access their local search experiences to identify relevant results that may be promoted. In addition, however, this query is also propagated along the search network links in order to probe the search expertise of trusted searchers with similar interests and identify further candidates for promotion (Figure 1). Any such promotions are then highlighted within, or added to, the result-list returned by the searcher’s primary search engine, for example Google or Yahoo.



**Fig. 1.** The search network is made up of a set of individual user search agents each with a local store of search experiences. Queries propagate throughout the network, allowing the searcher to benefit from the recommendations of others.

The basic search agent architecture is shown in Figure 2, and in the following sections we will describe this novel approach to collaborative Web search in detail, focusing on how local search expertise is represented, shared, and reused throughout a search network. We will describe how local search results can be ranked and combined with the results from similar agents by using a computational model of trust that reflects the reliability of users within the search network. In turn, we will explain how this trust model is fine-tuned by the search interactions of pairs of users, and how the overall search network adapts to these evolving search relationships.

#### 3.1 Experiences and Cases

Each search agent maintains a local case base of search experiences ( $C^U$ ) such that each individual *search case* reflects the result selections of the user,  $U$ , for a particular query — accepting that these result selections may have been made

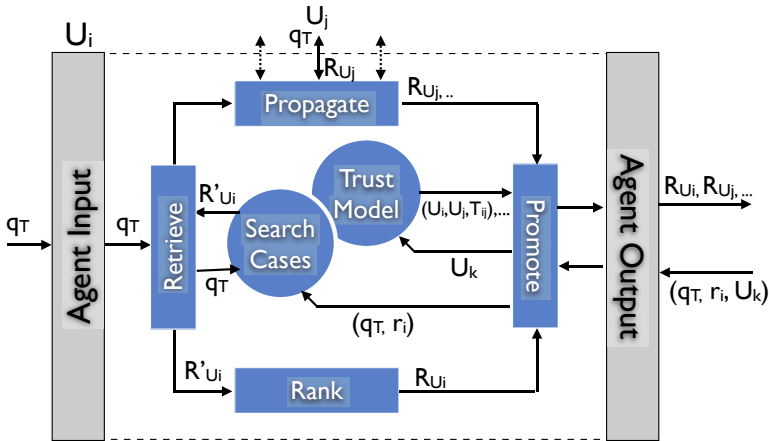


Fig. 2. The basic search agent architecture

over multiple search sessions. Thus each search case,  $c_i^U$ , is represented as a  $n+1$ -tuple made up of a query component and a result component; see Equation 1. The query component,  $q_i$ , is simply the set of terms that were used in the search query. The result component is made up of  $n$  result-pairs, with each comprised of a result-page id,  $r_j$ , and a hit-count,  $h_j$ , that reflects the number of times that  $U$  has selected  $r_j$  in response to  $q_i$ .

$$c_i^U = (q_i, (r_1, h_1), \dots, (r_n, h_n)) \quad (1)$$

It is important to note that, compared to the previous community-oriented versions of CWS [2], this peer-to-peer approach shifts the focus away from a community of searchers and on to the individual user. Instead of the case base corresponding to the community's search history, in this instance it corresponds to the search history of an individual. From a privacy viewpoint, however, it is worth highlighting that unlike the community-oriented version of CWS, where community case bases are stored centrally on a third-party server, this peer-to-peer approach facilitates a local, client-side store of search history information and thus provides the searcher with a further degree of security, privacy, and control over the use of their search data.

### 3.2 Retrieval and Ranking

The basic case retrieval implemented by each search agent is similar to that employed by community-based CWS [3]. In short, the target query,  $q_T$ , is compared to the search cases in the agent's local case base, and those cases that are deemed to be similar are retrieved ( $R'_{U_i}$ ). Case similarity is based on a simple term-overlap metric (see Equation 2), although more sophisticated approaches can be applied and have been evaluated elsewhere [12].

$$Sim(q_T, c_i) = \frac{|q_T \cap q_i|}{|q_T \cup q_i|} \quad (2)$$

Each retrieved case contributes a set of results that have previously been selected by the user for a query that is similar to the target query. The *local relevance* of a result is calculated based on how frequently it has been selected for a case, as shown in Equation 3.

$$Rel(r_j, c_i) = \frac{h_j}{\sum_{k=1}^n h_k} \quad (3)$$

An *overall relevance* score for a result  $r_j$ , with respect to  $q_T$ , is calculated as the weighted sum of these local relevance and query similarity scores (see Equation 4); once again, this overall relevance metric is borrowed from community-based CWS. Results that have been frequently selected for very similar queries should be considered more important than those that have been less frequently selected for less similar queries, and so the list of local search results,  $R_{U_i}$ , is ranked according to these overall relevance scores.

$$WRel(r_j, q_T, c_1, \dots, c_m) = \frac{\sum_{i=1}^m Rel(r_j, c_i) \cdot Sim(q_T, c_i)}{\sum_{i=1}^m Exists(r_j, c_i) \cdot Sim(q_T, c_i)} \quad (4)$$

### 3.3 Propagation and Collaboration

So far, we have described how a given agent retrieves and ranks a local set of search results based on its user's prior search experiences. Each agent is also connected to a number of peer nodes (search agents belonging to other users) in the search network. The agent propagates  $q_T$  to each of these peers in order to receive their local search recommendations, with each peer producing their recommendations using the same basic process. These agents will in turn propagate  $q_T$  on to *their* peers, and so on. As a practical matter, query propagation is limited to a fixed number of propagation steps according to a *time-to-live* counter that is decremented and passed on with each propagated query.

Ultimately, agents will be connected because there is some history of collaboration when it comes to prior search sessions. One agent may have suggested a search result which came to be selected by the receiving agent, for example. These positive examples of *search collaboration* serve to strengthen the trust between connected agents, which we shall discuss in the next section. Before we do, however, it is worth highlighting another way that the search network can adapt to search collaboration. As queries are propagated through the network, the *target agent* (the agent that is the original source of the target query) may receive recommendations from *distant* agents through a chain of network connections. If the target agent's user comes to select one of these distant recommendations, then it speaks to the potential for further positive search collaborations between these search agents in the future. This provides the basis for a simple approach

to network adaptation: by connecting agents that collaborate. In Figure 1, we can see that the searcher corresponding to agent A selects a recommendation that has come from agent C, resulting in the creation of a direct link between A and C. For simplicity, in this work we create a connection at the first sign of such collaboration, but in reality there is significant scope for further research on this particular topic to look for a more robust mechanism for adaptation that is not misled by what could be spurious collaborations. Similarly, if two connected agents fail to collaborate, then there is scope to sever their connections.

Of course, when a user joins the search network for the first time, a set of seed connections is needed to initialise their search network. There are a number of ways that such connections might be identified in practice. For example, the user might be asked to provide a list of friends, or connections might be selected automatically from a centralised list of reputable searchers. In our evaluation in Section 4 we simply choose a set of initial connections at random and let each user’s local search network evolve from there.

### 3.4 Trust, Promotion and Provenance

Each agent is responsible for generating a set of result promotions based on the combination of its own *local recommendations* and the *remote recommendations* that have been returned by its neighbours as a result of query propagation. Remember that each of these recommendations is accompanied by a relevance score (as per Equation 4), and they must now be combined to produce a ranked promotion list. To do this, there is one further vital source of information that needs to be described: the *trust model*.

The previous section referred to the notion of collaboration between searchers via their search agents — in the sense that a result suggested by one user (or, more correctly, their agent) might be subsequently selected by another user — and how frequent collaboration could be used as the basis for a computational model of trust between users. Simply put then, we can model the trust between a pair of directly connected users,  $U_i$  and  $U_j$ , as the percentage of recommendations that  $U_j$  has made to  $U_i$  which have come to be selected by  $U_i$  (as shown in Equation 5). Obviously trust, as we have defined it, is an asymmetric relationship because  $U_j$  may be a better source of search recommendations to  $U_i$  than  $U_i$  is to  $U_j$ . This simple trust model is straightforward to implement, with each agent maintaining trust scores for its peers and updating them after each search session.

$$Trust(U_i, U_j) = \frac{SelectedRec(U_j, U_i)}{TotalRecs(U_j, U_i)} \quad (5)$$

The key point is that we can use an agent’s trust score as a way to weight its recommendations, so that the relevance score that accompanies a remote recommendation is modified by the trust score of its contributory agent as shown in Equation 6; where  $WRel(r_k)$  is the weighted relevance score of result  $r_k$  which has been recommended by  $U_j$  to  $U_i$ .

$$TRel(U_i, U_j, WRel(r_k)) = Trust(U_i, U_j) \cdot WRel(r_k) \quad (6)$$

But, via query propagation, users can also receive recommendations from agents that they are not directly connected to and that they have no trust score for. To accommodate this, the trust-weighted relevance score of the recommendation is updated at each step as it is propagated back to the agent that issued the query. In this way, the relevance score is scaled according to the trust scores that exist between connected agents. Thus, the *provenance* of a recommendation has a concrete influence on its final relevance score; see [4] for related work. If a remote recommendation propagates through a short chain of highly trusted peers, then its relevance score will be largely preserved. Alternatively, if a remote recommendation propagates through a long chain of less trustworthy peers, then its relevance score will be greatly discounted. Ultimately, the target agent will assemble a combined list of local and remote recommendations ranked according to their appropriate relevance scores. If a given recommendation has arrived from multiple sources, then its relevance scores can be combined additively.

The final step for the target agent is to promote the final set of recommendations within the result-list that is returned for the target query by the baseline search engine (e.g. Google, Yahoo etc.). In practice, this means highlighting those results in the result-list that also appear in the recommendation-list. Additionally, the *top-k* (with  $k = 3$  usually) most relevant recommendations are promoted to the top of the result-list.

### 3.5 An Example Session

Figure 3 presents a simple example of this peer-to-peer approach to Web search in operation. In this case the query used, ‘cbr’, is ambiguous (at least to Google), and produces a result-list where none of the first page of results refer to case-based reasoning. In this example, the query has been propagated through a search network of peers, many of whom have an interest in various aspects of case-based reasoning and related AI research. Consequently, the top ranking recommendations that are returned provide a more relevant set of results for the searcher than the default Google list. In this case the top-3 most relevant results have been promoted, and each refers to an important CBR site.

It is worth highlighting how each result recommendation is annotated with icons that provide the searcher with hints as to the origins of the recommendation. For example, the icon that depicts a lone individual (see Figure 3) indicates that the result in question is a local recommendation that, by definition, has been previously selected by the current searcher for a similar query. In contrast, the icon that depicts a group of individuals indicates that the result is a remote recommendation from the searcher’s peers. In the example shown, the top-ranking result is both a local and a remote recommendation. The screenshot also shows that “mousing-over” the group icon reveals further information about the origins of the recommendation, including the “names” of the contributing searchers and the queries that they have selected this result for in the past. In the example, we see that the searcher has chosen to view more information about the user ‘mabes25’, and is shown that this user has selected this particular result for two other queries: ‘research cbr’ and ‘cbr publications’.



Google Web Images Groups News more »

cbr Search Advanced Search Preferences

Search:  the web  pages from Ireland

**Web** Results 1 - 10 of at

**Cbr** Sponsored Link

[www.ebay.ie](http://www.ebay.ie) 1000s of new & used vehicles added every day. Find it here!

**Case-Based Reasoning [the AI-CBR Homepage]** research  
 Featured papers, researchers and projects, links to CBR people, research projects, along with a comprehensive guide to software tools, ...  
[www.ai-cbr.org/](http://www.ai-cbr.org/) - 24k - [Cached](#) - [Similar pages](#)

**Case-Based Reasoning [Commercial Organisations using CBR]**   
 This pages lists commercial organisations who are using Case-Based Reasoning, provide further information. AI-CBR.  
<http://www.ai-cbr.org/applied.html>

Result was selected by **you** and **9** others:

Recommender:	Trust:
+ kkoym	94.4%
+ evanp	76.9%
- mabes25	72.6%

**Queries:**  
 research cbr  
 cbr publications

[More recommenders...](#)

**Wikipedia: Case-based reasoning**   
 Wikipedia Free Encyclopedia's article on 'Case-based reasoning'  
[http://en.wikipedia.org/wiki/Case-based\\_reasoning](http://en.wikipedia.org/wiki/Case-based_reasoning)

**Comic Book Resources - Daily Comic Book News, Reviews, Previews ...**  
**Comic Book Resources** - Daily Comic Book News, Reviews, Previews, Commentary and Message Boards.  
[www.comicbookresources.com/](http://www.comicbookresources.com/) - 54k - 27 Aug 2006 - [Cached](#) - [Similar pages](#)  
[Community](http://forums.comicbookresources.com/) - <http://forums.comicbookresources.com/>  
[Lying in the](http://www.comicbookresources.com/r.cgi/litg) - <http://www.comicbookresources.com/r.cgi/litg>  
[\[CBR News\]](http://www.comicbookresources.com/news/) - <http://www.comicbookresources.com/news/>  
[Permanent](http://www.comicbookresources.com/r.cgi/pd) - <http://www.comicbookresources.com/r.cgi/pd>  
[More results from www.comicbookresources.com >](http://www.comicbookresources.com/)

**Welcome to Central Board of Revenue , Government of Pakistan**  
 Tax authority.  
[www.cbr.gov.pk/](http://www.cbr.gov.pk/) - 2k - [Cached](#) - [Similar pages](#)

Fig. 3. A search result-list from Google enhanced by CWS recommendations

### 3.6 Discussion

Identifying the individuals responsible for a result promotion is an important departure from the traditional (community-based) model of CWS [2]. It is not without its challenges, but it does bring significant potential benefits when it comes to the facilitation of high quality search collaborations between users.

First and foremost, this new P2P collaborative Web search (P2P-CWS) approach is proposed as an effective strategy for coping with *recommendation spam*: previous versions of CWS were found to be somewhat susceptible to the actions of malicious users promoting irrelevant results [5]. The trust model used in this peer-to-peer approach provides for a very practical coping strategy in the face of such attacks, because promotions can only be made by a remote user if there is already a path of trust connections to the target searcher. Of course, this does not preclude more sophisticated forms of attack. For example, a particularly devious user might ‘groom’ the searcher by baiting them with good recommendations

early on, in an attempt to gain their trust, before inserting irrelevant results into the recommendation stream. However, the searcher is likely to recognise and ignore such spurious promotions, which will quickly erode the false-trust that had been built up. Furthermore, the malicious user does not receive any direct feedback on the effectiveness of their efforts.

Ultimately, of course, trust is not simply a computational measure of collaboration between searchers. It is a social construct that develops as a result of social interactions. And the anonymous promotions of community-based CWS effectively limit the type of social relationships that can develop between searchers. It is clear from trial data that some searchers are better promotion sources than others, but this information is lost in community-based CWS. P2P-CWS is different. It provides information about the provenance of promotions by labeling recommendations with the names of the searchers who contributed to their recommendation. And this affords the searcher an opportunity to develop an implicit social connection with other searchers. If a searcher finds that they frequently benefit from the recommendations of a particular user then they will be naturally drawn to this user's recommendations in the future as they come to trust in their search experiences. Equally, if a searcher is seldom interested in the recommendations of another user then they will quickly learn to avoid recommendations from this user. All of this is independent of the computational model of trust that co-develops as such collaborations persist and mature.

Where community-based CWS neatly side-stepped the privacy issue by obscuring any personal search histories within community case bases, the new model's requirement of individual search histories clearly raises some significant privacy demons. The peer-to-peer architecture is a direct response to this. It provides for an increased level of privacy and security by eliminating the need for a *central* store of search histories. Instead, each user's searches are stored locally on their client and accessed by their personal search agent. This provides the individual user with a significant level of control over the sharing of their valuable search data. For example, it is feasible to allow the user to control their local search network and to influence which other search agents they are connected to. In this way, only other trusted users are permitted to contribute to, or benefit from, a given user's search experiences. When it comes to the propagation of queries, privacy is aided by the fact that when an agent receives a query request it knows only of the *forwarding agent*, and nothing of the agent that initiated the search. However, although agents handle such query requests automatically, it is possible for a motivated user to intercept them. Consequently, as is the case with search logs, personal information in the query could pose a privacy risk.

Finally, it has been noted that, with our current trust model, a peer who makes useful recommendations on one topic may have their trust score reduced unfairly if their recommendations for an entirely different topic are rarely selected. Future work may address this issue by adjusting the trust model so that scores are not reduced in such cases, or by maintaining topic-specific trust scores.

## 4 Evaluation

We have described an alternative approach to CWS which provides a searcher with personalized search recommendations that are drawn from the related search experiences of a set of trusted searchers. In this section we test this approach by evaluating the recommendations that are generated within an evolving search network. In addition to the traditional precision-recall study, we also examine the evolution of the search network as collaboration and cooperation between search agents unfold, with a view to better understanding how the trust model adapts during the course of an extended period of time.

### 4.1 Data

Ideally we would have liked to test P2P-CWS in a live-user setting, but this was not feasible. We considered a small-scale laboratory trial, but our previous experience tells us that such limited studies are rarely very revealing. At the same time, the alternative strategy of using simulated users is equally problematic even though it offers greater scope for large-scale evaluation. In this work we have chosen to adopt a middle-ground by using the search profiles of 50 real users as the basis for our search network, and then applying a leave-one-out methodology to evaluate various performance metrics such as precision and recall.

As a source of search data, we used the profiles of 50 users from the Del.icio.us<sup>1</sup> online social bookmarking service. In doing so, we follow the work of [13,14] by treating each bookmarked page as a result selection with the user's tags acting as query terms. Thus, each tag and its bookmarks acted as a proxy for a search case with its query and associated result selections. Obviously, the core assumption behind P2P-CWS is that there will be some opportunity for collaboration between the various searchers in the network, and this can only come about if there is overlap between their various search interests. Thus we focused on the first 50 users that Del.icio.us listed as having tagged the <http://www.foaf-project.org> URL (the home page of the *Friend of a Friend* project), on the grounds that there would be a reasonable opportunity for naturally overlapping search interests from this group without actually biasing the results by forcing overlap. For each user, we retrieved all their bookmarked URLs and their associated tags. This produced an average of 406 bookmarks (pages and queries) per user, with the typical profile containing an average of 242 unique tags (query terms).

The search network corresponding to these 50 users is initialised by randomly connecting each user to 10 other users, and all trust scores are set to the default of 0.5. An alternative would have been to connect each individual user to a set of other users based on some assessment of their similarity (for example, query or page overlap), but we chose this more challenging initialisation strategy in part because it provides a tougher test of network adaptation and trust evolution.

---

<sup>1</sup> <http://www.del.icio.us>

## 4.2 Methodology

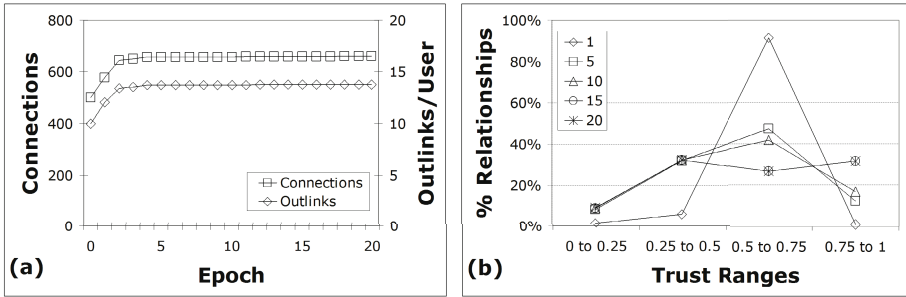
To evaluate the performance of P2P-CWS, we adopt a leave-one-out methodology in which each user in turn is designated as the target user to whom recommendations will be made. We re-run each of the target user’s search queries through the search network and examine whether the recommendations produced contain any pages bookmarked by the target user for that current query. During each search we remove the corresponding search case from the target user’s local search case base so that they cannot receive recommendations based on their own result selections. Obviously this is a fairly strict notion of result relevance. Many recommendations may actually be relevant to the query, but will not be judged as such unless the user had deemed to bookmark them in the past. Nevertheless, this approach at least provides a lower-bound on relevance and has the advantage that it can be fully automated.

The above methodology is repeated for a number of iterations or, *epochs*, to allow for the trust models to evolve as a result of sharing and collaboration between search agents. This also allows us to explore how search performance changes as the network adapts to search collaborations. After each search session, the trust model of the searcher is updated to reflect any selections — according to the above strict notion of relevance, we assume that the searcher will select any relevant recommendations that have been made.

## 4.3 The Evolution of Trust

Before we come to look at the precision-recall performance of P2P-CWS, it is interesting to examine how the search network and the trust models evolve during the experiment. In Figure 4(a), we present a graph of the number of network connections within the network. The experiment begins (epoch 0) with 500 connections (since each user is randomly connected to 10 other users), but as the experiment progresses we see new connections being formed as searchers collaborate successfully. Interestingly, we see that the majority of new connections are forged during the first 4 epochs as the network structure quickly converges. As a matter for future work, it would be interesting to validate this convergence behaviour over different and larger-scale networks.

Just as the structure of the search network evolves over time, so too do the trust models employed by the individual search agents. The results in Figure 4(b) show a series of trust-score histograms that highlight the distribution of searcher-pairs with different trust scores; each histogram was generated at the end of a full epoch by counting the number of searcher-pairs with a trust score that fell within a given range of values. At the end of the first epoch, the majority of the trust relationships remain close to their default strength of 0.5; there are 579 trust relationships in our search network, and over 90% of these (529) have a score of between 0.5 and 0.75 at the end of epoch 1. However, the trust scores gradually settle as a result of search activity and, by the end of epoch 20, just under 30% of the relationships have a trust score in this range. Overall, we see a gradual flattening of the trust distribution curve, indicating that a broad range



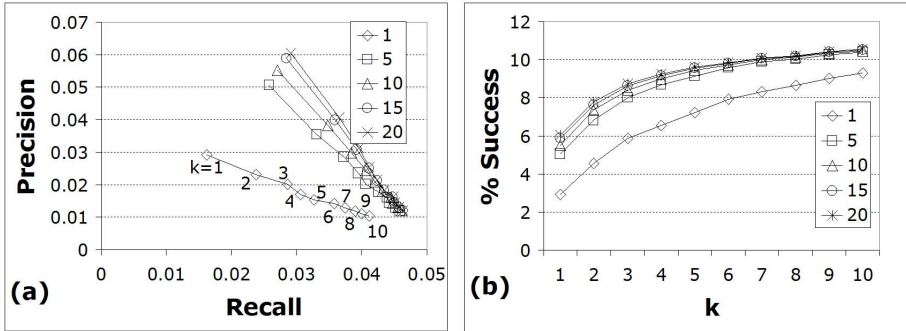
**Fig. 4.** (a) The number of peer-to-peer connections per epoch; (b) The changing distribution of trust scores per epoch

of trust scores are distributed throughout the network as searchers collaborate with varying degrees of success. Since, by design, the interests of this network of searchers are likely to overlap to some degree (they share a common interest in FOAF research), it is perhaps not surprising to see that, on the basis of the trust values presented, there is a considerable degree of productive collaboration within the network. For example, after 10 epochs we see that approximately 60% of trust scores fall in the 0.5-1 range, indicating a strong history of search collaboration between at least half of the search relationships encoded by the search network. Indeed, less than 10% of the relationships are weak, in the sense that they have trust scores below the 0.25 threshold.

#### 4.4 Recommendation Quality

The traditional metrics of information retrieval success are precision and recall. The former measures the percentage of results (recommendations) that are relevant, while the latter measures the percentage of relevant results that are recommended. In Figure 5 (a), we present a precision-recall graph in which each plot represents the precision-recall characteristics for recommendation lists of various sizes ( $k = 1, \dots, 10$ ) during each epoch. For example, in Figure 5 (a) the points that represent epoch 1 are labeled with their respective values of  $k$  so that the point corresponding to  $k = 1$  indicates that during the first epoch, when only the top recommendation was presented to the searcher, we found an average precision score of 0.03 and a recall score of 0.015.

There are a number of points to be made about these results. First, the precision and recall scores are unusually low, more because of the strict nature of our relevance judgement than any underlying shortcoming of the recommendations themselves. As is usually the case in this type of experiment, precision tends to decrease with increasing  $k$ , while recall tends to increase; as  $k$  increases it becomes less likely that additional recommendations will be relevant, but it is more likely that a greater number of relevant recommendations will be produced. Perhaps most importantly, we see a sustained improvement in precision-recall



**Fig. 5.** (a) Precision versus recall for result-lists sizes from 1 to 10; (b) Percentage of sessions with recommendations containing a relevant result within the top  $k$

during later epochs. This means that as the search network evolves, and as trust models adapt, better recommendations are being made. For example, by epoch 20 the precision and recall characteristics of the recommendations at the top of the list have effectively doubled.

In Figure 5 (b), we present an alternative performance graph which computes the average percentage of sessions that include a relevant result within the top  $k$  recommendations in sessions where recommendations are actually made. Once again, we see a steady increase in the percentage of *successful* recommendations as the trust network evolves. For example, during epoch 1, successful results are found in the top result-list position about 3% of the time, rising to just over 9% of the time if we consider the top 10 result-list positions. By epoch 20, this success rate has more than doubled for  $k = 1$ , with a success rate of over 6% at this position, and reaching nearly 11% for the top 10 results.

## 5 Conclusion

This work has been inspired by recent approaches to CWS [3] in which CBR techniques are used to harness the search experiences of communities of searchers. The research presented here is novel in that it provides for a more flexible CWS architecture; one that avoids the need for explicit search communities while delivering similar benefits in terms of search quality. Moreover, the peer-to-peer architecture provides a level of privacy and security that is sufficient to merit the use of individual user search profiles in place of community-based profiles, resulting in significant benefits when it comes to regulating the exchange of search experiences within the network. By profiling individual users, for example, it is possible to evaluate the reliability of searchers when it comes to recommending relevant results to others, and this can be used as an effective way to cope with search spam that may be introduced by malicious searchers within the network.

## References

1. Search Engine Watch: Worldwide internet: Now serving 61 billion searches per month, <http://searchenginewatch.com/showPage.html?page=3627304>
2. Smyth, B., Balfe, E., Boydell, O., Bradley, K., Briggs, P., Coyle, M., Freyne, J.: A live-user evaluation of collaborative web search. In: Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI 2005), pp. 1419–1424 (2005)
3. Balfe, E., Smyth, B.: Case-Based Collaborative Web Search. In: Proceedings of the 7th European Conference on Cased Based Reasoning, pp. 489–503 (2004)
4. Leake, D.B., Whitehead, M.: Case provenance: The value of remembering case sources. In: Weber, R.O., Richter, M.M. (eds.) ICCBR 2007. LNCS (LNAI), vol. 4626, pp. 194–208. Springer, Heidelberg (2007)
5. O'Mahony, M.P., Smyth, B.: Collaborative Web Search: A Robustness Analysis. In: Artificial Intelligence Review, Special Issue on the 18th Artificial Intelligence and Cognitive Science Conference (AICS 2007) (to appear, 2007)
6. Voss, A., Kreifelts, T.: SOAP: Social Agents Providing People with Useful Information. In: Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work, pp. 291–298. ACM Press, New York (1997)
7. Glance, N.S.: Community Search Assistant. In: Proceedings of the 6th International Conference on Intelligent User Interfaces (IUI 2001), pp. 91–96. ACM Press, New York (2001)
8. Jung, S., Harris, K., Webster, J., Herlocker, J.L.: Serf: integrating human recommendations with search. In: CIKM 2004: Proceedings of the thirteenth ACM international conference on Information and knowledge management, pp. 571–580. ACM, New York (2004)
9. Leake, D., Sooriamurthi, R.: When Two Cases Are Better Than One: Exploiting Multiple Casebases. In: Proceedings of the International Conference on Case-Based Reasoning. Springer, Heidelberg (2001)
10. Bender, M., Michel, S., Weikum, G., Zimmer, C.: Bookmark-driven query routing in peer-to-peer web search. In: Proceedings of the SIGIR Workshop on Peer-to-Peer Information Retrieval, 27th Annual International ACM SIGIR Conference (2004)
11. Suel, T., Mathur, C., wen Wu, J., Zhang, J., Delis, A., Kharrazi, M., Long, X., Shanmugasundaram, K.: Odissea: A peer-to-peer architecture for scalable web search and information retrieval. In: International Workshop on Web and Databases (WebDB), pp. 67–72 (2003)
12. Smyth, B., Balfe, E.: Anonymous personalization in collaborative web search. *Inf. Retr.* 9(2), 165–190 (2006)
13. Boydell, O., Smyth, B.: Enhancing case-based, collaborative web search. *Case-Based Reasoning Research and Development* 4626, 329–343 (2007)
14. Bao, S., Xue, G., Wu, X., Yu, Y., Fei, B., Su, Z.: Optimizing web search using social annotations. In: WWW 2007: Proceedings of the 16th international conference on World Wide Web, pp. 501–510. ACM, New York (2007)