Do You Want to Take Notes? Identifying Research Missions in Yahoo! Search Pad

Debora Donato
Yahoo! Labs
Sunnyvale, CA
debora@yahoo-inc.com

Francesco Bonchi
Yahoo! Research
Barcelona, Spain
bonchi@yahoo-inc.com

Tom Chi
Yahoo! Inc.
Sunnyvale, CA
tomchi@yahoo-inc.com

Yoelle Maarek
Yahoo! Research
Haifa, Israel
yoelle.maarek@yahoo.com

ABSTRACT

Addressing user's information needs has been one of the main goals of Web search engines since their early days. In some cases, users cannot see their needs immediately answered by search results, simply because these needs are too complex and involve multiple aspects that are not covered by a single Web or search results page. This typically happens when users investigate a certain topic in domains such as education, travel or health, which often require collecting facts and information from many pages. We refer to this type of activities as "research missions". These research missions account for 10% of users' sessions and more than 25% of all query volume, as verified by a manual analysis that was conducted by Yahoo! editors.

We demonstrate in this paper that such missions can be automatically identified on-the-fly, as the user interacts with the search engine, through careful runtime analysis of query flows and query sessions.

The on-the-fly automatic identification of research missions has been implemented in Search Pad, a novel Yahoo! application that was launched in 2009, and that we present in this paper. Search Pad helps users keeping trace of results they have consulted. Its novelty however is that unlike previous notes taking products, it is automatically triggered only when the system decides, with a fair level of confidence, that the user is undertaking a research mission and thus is in the right context for gathering notes. Beyond the Search Pad specific application, we believe that changing the level of granularity of query modeling, from an isolated query to a list of queries pertaining to the same research missions, so as to better reflect a certain type of information needs, can be beneficial in a number of other Web search applications. Session-awareness is growing and it is likely to play, in the near future, a fundamental role in many on-line tasks: this paper presents a first step on this path.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Search process; H.3.5 [**Online Information Services**]: Web-based services

WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.
ACM 978-1-60558-799-8/10/04.

General Terms

Human Factors

Keywords

Task-oriented search, Persistent search, Query logs, Sessions

1. INTRODUCTION

Users turn to search engines in order to satisfy their information needs [8]. The classical scenario is for them to express their needs as a free-text query and then scan through results in order to identify full or partial answers. Their needs can then be:

- **Satisfied:** this happens when users obtain the answer they were seeking immediately on the first page of results. They can either directly read the answer on the results page itself, when generated for instance by services such as a Google's oneboxes [16] or a Yahoo shortcut [32], (*e.g.*, calculator, weather and sports results), or reach it after clicking on one of the top ranked pages.
- **Not satisfied:** this happens when (1) users did not succeed in formulating their query, when (2) relevant content does not exist, or finally when (3) relevant content does exist but the search engine can not identify it. By default, most engines will focus at this stage on the first case (since there is not much they can do in real time with the two other cases) and will offer users query assistance tools, pointing them to related queries that might bring better results.
- **Partially satisfied:** this usually happens when users have complex information needs, for which there does not exist one single Web page that holds all the needed information. In this scenario, the user will like some of results, disregard others and continue exploring, clicking on a few results, trying out related queries, while still gathering information. Such sessions can last from a few minutes to several days. Examples include travel needs (when a traveler verifies hotels, restaurants or transportation means around a certain location), education needs (when students work on an assignment), or medical needs (when a patient studies a specific illness, its symptoms and treatments).

We are focusing in this paper on the third case, in which the users' needs are too complex or too heterogeneous to be answered in one shot. We refer to these non-ephemeral information seeking tasks as *"research missions"*. This notion relates to the notion of *information gathering* on the Web as introduced by Kellar *et al.* in [20], and defined as the collection of information from multiple sources. In the user's study they conducted over a period of one week in 2005 with 21 university students, the authors reported that information gathering tasks accounted for 13.4% of overall Web usage and was the fourth most important activity on the Web after "transactions" (46.7%), "just browsing" (19.9%) and "fact finding" (18.3%).

Research missions, as we define them, restrict and refine the notion of information gathering as they can occur only during search sessions as opposed to overall Web usage. By manually analyzing actual query sessions over a period of 3 days, we verified that on average 10% of search sessions are "research missions" but more interestingly, that about 25% of query volume occurs in these sessions. Based on this data, we argue that research missions deserve special attention and treatment.

Our longer-term vision is that automatically identifying these research missions could potentially break the classical search paradigm, which maps one query to a list of results, into one where entire sessions are analyzed for intent, and results match session intent. We believe that by considering queries in the context of a session of related expressions of a common need, rather than in isolation, search engines might be able to achieve a better understanding of the real users' intent. Our belief is supported by some recent works showing that not only the previous query, but also the long-term interests of users are important for understanding their information needs [23, 27].

As a first step towards this long-term goal, we propose to demonstrate that research missions can indeed be discovered on the fly, while users are interacting with the search engine, with a good enough precision rate to make them valuable. In order to substantiate our claim, we will use the recently deployed Yahoo! Search Pad application, which integrates a research mission identification component that we have developed.

Search Pad is an application that has been designed precisely to help users undertake research missions. Search Pad allows users to easily keep trace of results they have consulted. They can arrange and annotate them for later personal usage or for sharing with others. Search Pad's novelty comes from its being triggered only when the search engine believes that the user is undertaking a research mission rather than looking for quick, disposable results. Visited pages are automatically added to the appropriate search pad (either the current one when the query belongs to the same research mission or a new one if necessary) without requiring the user to specifically "mark" them. There have been multiple attempts in the past to support similar functionality, such as Bharat's original Searchpad[1] [4] research tool, or Google's notebook [14], whose development has now been discontinued [22]. A major difference between these previous tools and Yahoo! Search Pad was that they required users to proactively invoke the notes taking tool and to manually mark relevant pages. In contrast, Search Pad's novelty comes from its automatic identification of research missions and its being triggered only when the system decides, with a fair level of confidence, that the user is in the right context for gathering notes.

As such, Search Pad represents the ideal application for us to verify our claim that identifying and using search missions is valuable to users. Search Pad is automatically triggered at query time when a search mission is identified.

The goal of this paper is thus to demonstrate, using actual data gathered by Search Pad since it has already been deployed in the US for a few months, that research missions can be automatically detected, at the scale of Web (meeting performance and scalability requirements) and with enough accuracy that they bring value to users.

The rest of this paper is organized as follows. Section 2 describes the Search Pad application that we will use as basis for demonstrating the validity of our approach. Section 3 provides the needed preliminaries and definitions of the technical problem tackled in this paper: automatically identifying research missions. Section 4 describes our solution, *i.e.*, the internals of the triggering component of Search Pad, a machine-learning based module aimed at detecting research missions to trigger Search Pad. Section 5 reports our evaluation using Search Pad and provides results inferred from actual usage data. We conclude with directions for future work, first for enhancing Search Pad, and then for leveraging the benefits of research missions in other domains such as core ranking.

2. SEARCH PAD: THE APPLICATION

In spite of recent efforts at personalizing search, and storing past interactions for logged-in users, Web search remains basically "stateless".

Search Pad is a feature of Yahoo! Search that was launched in July 2009 and precisely addresses this issue. It help users keep track of related searches and visited pages that relate to a same "research mission"[2].

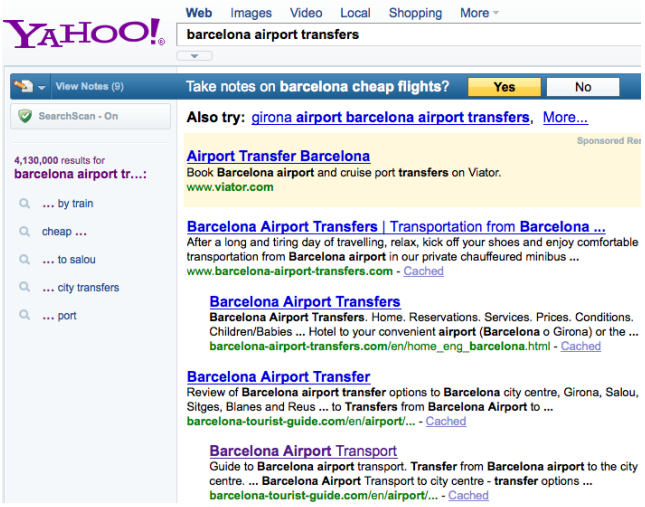

Figure 1: Take notes on Barcelona cheap flights

[1]Note that Bharat's tool was called "SearchPad" in one word, while ours is called "Search Pad" in two words.

[2]Note that we have defined this concept only intuitively so far, which is sufficient for understanding the application and usage scenarios. We will formalize the definition of search missions in the following section.

To illustrate, let us consider the following interaction scenario with Search Pad. A user is planning a trip to Barcelona and starts issuing a few related queries, such as `Barcelona cheap flights`, `Barcelona airport` and `Barcelona airport transfer`, clicking each time on a few results. When issuing the query `Barcelona airport transfer`, Search Pad detects that the user has been researching a topic rather than looking for quick answers and asks whether she wants to take notes, as shown in Figure 1.

If she accepts, she will then be shown a "Search Pad document" , or "pad" for short, already populated with the links of the pages she has visited during the current research mission, together with their thumbnails, as shown in Figure 2. At this stage, the user can delete some of the links, move them around, add some personal notes, and eventually save the pad.

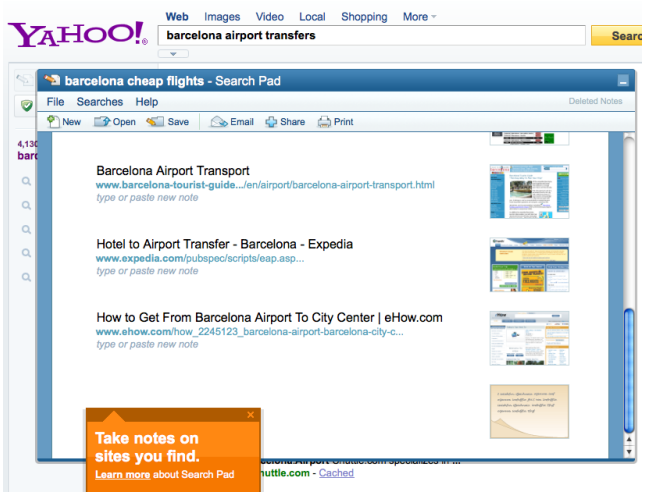

Figure 2: An opened pad object

The pad object is fully editable, supporting drag and drop and paste operations and can be reopened and reused at a later time, for sessions that span over a number of days or weeks. It can also be shared on Facebook, Twitter, and Delicious or simply embedded via a persistent HTML link into any Web page, as illustrated in Figure 3.

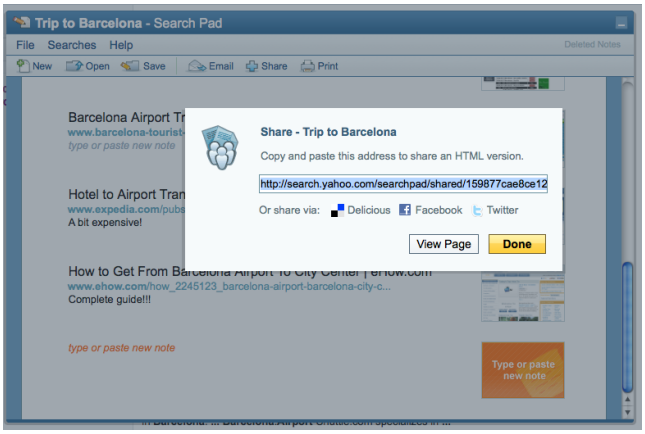

Figure 3: Sharing a pad object

Previous efforts at taking notes while searching or browsing the Web can be roughly classified into three categories:

Link centric efforts were the first to appear as it was clear that users would not easily remember URLs of interesting pages while surfing the Web or visiting search results. Bookmarks were invented for this purpose and were an integral part of Web browsers since the early days of the Web (under the name "Hotlist" in the original Mosaic, or "Favorites" in Internet Explorer). In spite of a few research attempts at automatically structuring them [21, 24], bookmarks have been shown to be complex to manage as soon as their number goes over a few dozens [1]. Some applications such as Zotero [33] or hosted bookmark services, as offered by major Web search engines, do a slightly better job at managing and searching bookmarks. Google launched in 2006 its Notebook under the form of a hosted service, and browser plug-in, that allowed users to store links in a collection of "notebooks", and annotate them with comments and labels. This tool saw its development discontinued in 2009 and stopped accepting new users. There exist numerous alternatives for note taking, such as Evernote [11], yet no clear winner has emerged to make the unanimity in this market. Finally bookmarks took an interesting turn with the social bookmarking[3] phenomenon, in the Web2.0 model of sharing comments and opinions with the community. Nevertheless, at the personal level, native browser bookmarks are still the most handy means of keeping track of interesting search results.

Page centric efforts work at a finer level of granularity by allowing users not only to save interesting results but to directly annotate paragraphs or passages in the page. An example of application providing this functionality is Diigo [10], which allows users to attach sticky notes to highlighted part of the Web page in a persistent manner. Hunter Gatherer [28] is a similar tool that allows to collect components from within Web Pages in just one single page. The tool supports both the within-page collection making and the information management. Another recent example is Google Sidewiki [15], a new feature of Google's toolbar that allows users to publish and share annotations on a given Web page.

Search centric efforts are slightly different as they are specific to search results, while the previous ones can also relate to arbitrary pages reached by other means. The first piece of work in this space was SearchPad [4], which used a proxy to decorate results from a list of search engines (namely AltaVista, Excite, Google and HotBot) with a "Mark" button. Clicking on this button resulted in storing the selected results in SearchPad together with their associated queries. Users could at any time visit SearchPad to retrieve all "bookmarked queries", and their associated "leads", *i.e* results manually marked by users. A few years later, Ask offered the related MyStuff service [2]. MyStuff also decorates each search result with a "save" link, which allows registered users to save interesting results for later usage.

Yahoo! Search Pad differs from these solutions on several aspects

[3]We do not discuss social bookmarks here as they are out of the scope this paper, which focuses on the personal benefits of identifying research missions.

1. The most critical differentiator is the triggering mechanism, which prompts the user for taking notes only when chances of its being useful are high. In other words, Search Pad is made clearly visible to users during these research missions for which it should be most needed, and stays out of the way otherwise. This feature is to our knowledge not available in any other existing tool and is supported by the live research mission detection mechanism, which is our key technical contribution in this paper.

2. Another important feature is the automatic distinction between different sessions based not only on time considerations but also on topical coherences between related queries as more formally defined later. Thus, when a user undertakes a different research mission, Search Pad should segment between those and automatically start a new pad, if given sufficient evidence.

3. Additionally, links are automatically added to a search pad in such a manner that when a pad is prompted to a user, it is prepopulated with content. While not all links might be relevant to the same extent, this motivates the user to continue taking notes. It is also easier to delete less relevant links, as opposed to breaking the search flow by selectively adding them.

4. The last difference is that Search Pad does not require users to use a dedicated plug-in, a specific toolbar or separate application, but is built-in within the main search service of Yahoo!. Users will be required to sign-in (but not to register to a specific service) only when saving a pad. This feature significantly reduces the adoption barrier.

We will focus in the rest of this paper on the first two aspects, which are the ones that required technical novelty, namely the automatic identification and segmentation of research missions.

3. PROBLEM STATEMENT

Numerous studies [5, 6, 7, 17, 19, 25, 26, 29] have investigated various aspects of users' behavior on the Web. As a result, it is now commonly agreed upon that the information inferred by mining users' search activities is extremely valuable to search engines. Given that the extracted information is adequately anonymized and aggregated, it is a critical source of information for delivering quality results in any type of search engines service, such as ranking, spelling and query assistance, to name just a few.

We define below the basic elements and objects that are used in mining users' interactions.

3.1 General Definitions

Query log. Search engines store information about user's interactions in "query logs". Query logs differ in format between search engines, but at a bare minimum typically associate with each submitted query, a list of results, as well as whether or not users clicked on them. More formally a query log $\mathscr{L}$ is a set of records $\langle q_i, u_i, t_i, V_i, C_i \rangle$, where: q_i is the submitted query, u_i is an anonymized identifier for the user who submitted the query, t_i is a timestamp, V_i is the set of documents returned as results to the query, and C_i is the set of documents clicked by the user (which can be empty).

Sessions. We reuse here the definition of query session (or session for short) that was given in [5]. A session is the *sequence of queries issued by a single user within a specific time limit.* If $u \in \mathscr{U}$ is the user identifier and t_θ is a timeout threshold, a user query session S is defined as a *maximal* ordered sequence

$$S = \big\langle \langle q_{i_1}, u_{i_1}, t_{i_1} \rangle, \ldots, \langle q_{i_k}, u_{i_k}, t_{i_k} \rangle \big\rangle,$$

where $u_{i_1} = \cdots = u_{i_k} = u \in \mathscr{U}$, $t_{i_1} \leq \cdots \leq t_{i_k}$, and $t_{i_{j+1}} - t_{i_j} \leq t_\theta$, for all $j = 1, 2, \ldots, k-1$.

The activity of a same user is split in two (or more) sessions whenever the time interval between two sequential queries exceeds the timeout threshold. The typical timeout is $t_\theta = 30$ minutes [9, 25, 30].

Chains and Missions, Radlinski and Joachims [26] define a query chain (or chain for short) as a sequence of reformulated queries that express a same information need. Baeza-Yates *et al.* define in [3] the related concept of *logical session* and Jones *et al.* in [19] define the concept of *mission*, as *"a related set of information needs, resulting in one or more goals"*, where a goal is *"an atomic information need, resulting in one or more queries"*. To illustrate, Jones *et al.* give, as example of a sequence of queries that can be mapped into a same mission, the following set of consecutive queries: `(brake pads, auto repair, auto body shop, batteries, car batteries, buy car battery online)`.

3.2 Research Sessions and Missions

An interesting distinction between chains and missions is that chains deal with the actual sequence of queries, while missions refer to the underlying information needs. Following this distinction, we define here the notions of *research session* and associated *research mission*.

A research session belongs to the same space as a chain and represents a set of queries (and associated information as provided by query logs) that fulfills certain constraints.

A research mission, in the same space as Jones' missions, is *a set of related and complex information needs.* Such complexity is reflected by the user's engagement such as the time spent or the number of atomic tasks accomplished by the user in the attempt of fulfilling these needs. Thus, a research session is defined as the set of *all the user activities (queries and clicks) needed to fulfill a research mission.*

We introduce these two definitions to insist on the fact that research sessions do not follow some arbitrary timeout rules, as sessions typically do via the previously mentioned t_θ threshold. Indeed, as research sessions reflect research missions, they can span over a number of hours if not days. To illustrate, consider our previous example of a user who is planning a trip to Barcelona and searches for cheap plane tickets and accommodations. She will typically pursue this task over a period of several days. Consequently the queries composing a research session do not need to be consecutive. Following the previous example, our user might search for plane tickets then search for reviews and show times of a few movies before going to the theater the same evening. She will then return to her trip planning the next day, searching for a hotel in Barcelona. Thus, a general session bounded by a given t_θ may contain queries reflecting several distinct

missions, *e.g.*, trip to Barcelona, best movie to see tonight, and a research session may contain queries originating from several distinct sessions, *e.g.*, one session on a day, and another on the following day.

As they are not limited in time, research sessions use other signals to identify a shared research mission, namely topical coherence and user's engagement.

To reflect topical coherence, we use the two functions f and s as defined below. First, given a query $q \in \mathscr{Q}$ and a set of topics $\mathscr{T}$, we use a function $f : \mathscr{Q} \to \mathscr{T}$ that maps each query q in the query log $\mathscr{L}$ to a topic p. In the extreme case, f can simply be the identity function, no topical association is provided, and we will not be able to identify that the queries `auto repair` and `car batteries`, as per our previous example, belong to the same research session.

We will verify in the evaluation section that an adequate choice of f does increase recall and thus recommend picking a function f that can be computed efficiently at the scale of Web traffic. A number of options exists to define f, depending on the desired level of precision, one possible solution being to use Wikipedia as source of world knowledge as done in [13]. Most popular Web search engines do have at their disposal such functions that they use to improve recall for sponsored search [12] and this what we recommend using in our case.

In addition, we use a similarity measure among topics in $\mathscr{T}$, that is a function $s : \mathscr{T} \times \mathscr{T} \to [0, 1]$. Again in the extreme case of f being the identity function, a simple similarity function can be the normalized ratio of common words or stems between two queries. In that case, in our previous example of "`batteries`" and "`car batteries`", the measure of similarity will be 0.5. More sophisticated techniques as proposed by [13] can be used.

To reflect user's engagement, we augment the basic $\langle q, u, t\rangle$ query/user/time session tuple with the click information stored in query logs, namely the set of clicked results C. We use C or rather the size of C, as will be seen below, as a strong signal of the complexity of a research mission. The larger $|C|$, the more probable it is that the user did not see her needs satisfied and is in the unsatisfied/partially satisfied category and therefore continues exploring. Intuitively, long sessions with a sufficient number of related queries, for which the user exhibits a stronger engagement in terms of actions performed, that is, with a high $|C|$, are good candidates for research sessions. A more sophisticated model that would allow to better distinguish between unsatisfied and partially satisfied needs would be to work at a finer level of granularity and verify that some of the clicks are "good clicks" in the sense that the user spent enough time reading the selected results and came back to the session to continue exploring. We reserve this finer analysis for future work and limit ourselves first to the simple observation of the number of clicks.

Thus, we formally define a research session R as a maximal order sequence

$$R = \big\langle\, \langle q_{i_1}, u_{i_1}, t_{i_1}, C_{i_1}\rangle, \ldots, \langle q_{i_k}, u_{i_k}, t_{i_k}, C_{i_k}\rangle \,\big\rangle,$$

where, for given thresholds s_θ, k_θ and c_θ, we have:
1. $u_{i_1} = \cdots = u_{i_k} = u \in \mathscr{U}$, and $t_{i_1} \leq \cdots \leq t_{i_k} \leq \tau$;
2. $\forall l, j \in \{i_1, \ldots, i_k\}\ \ s(f(q_l), f(q_j)) \geq s_\theta$;
3. $|R| = k \geq k_\theta$;
4. $\sum_{j=1}^{k} |C_{i_j}| \geq c_\theta$.

The second condition reflects topical coherence, while the third and fourth ones reflect user's engagement, via the total number of issued queries and the total number of clicked results.

In the following section, we present our solution for automatically detecting research sessions and thus identifying research missions.

4. AUTOMATIC IDENTIFICATION OF RESEARCH MISSIONS

The Search Pad application previously introduced makes research missions persistent by storing in a pad relevant information pertaining to a given research session. As such, it is the ideal application for us to verify that research missions can be efficiently and effectively identified, either for direct usage as described in Section 2 or future research.

Our contribution to Search Pad was its Triggering System, whose goal is precisely to trigger Search Pad whenever a research mission is identified. The rest of this section describes the architecture and internals of this system.

4.1 Architecture of the Triggering System

As discussed in the previous session, Search Pad must be triggered when the user is involved in a research mission, which is a *long*, *complex* sequence of searches that are *topically coherent*. Therefore the Triggering System must focus on signals such as session length, user's engagement and topical coherence. Moreover, some query topics are more likely to be involved in research missions, such as, for instance, `travel`, `health`, or `job search`.

Keeping in mind these consideration, we built the Triggering System of Search Pad as a two-level classifier, whose architecture in depicted in Figure 4.

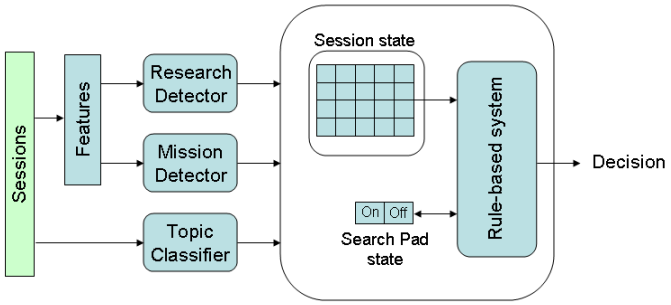

Figure 4: The core of the Triggering System.

The system is built around three base models, which are each responsible for different tasks, and a meta-classifier, called hereafter the *"mixer"*, that receives the signals arriving from the base classifiers and combines them in order to make the final decision of whether to trigger Search Pad or not.

Given the scale of deployment of a system like Search Pad, it was necessary to allow administrators to react fast and tune, boost or disable certain features in case of poor unexpected behavior. Consequently we opted for a two-tier architecture that gives control to the Search Pad administrators without requiring deep understanding of the internal operations.

More specifically, at the higher level, the details of the base classifiers remain hidden in a black-box mode, and only the semantics of the signal they produce need to be understood.

In addition, the output of the mixer is a probability value that expresses a confidence level. Search Pad is triggered if this probability is higher than a given threshold.

Consequently, depending on on-going marketing or assessment studies, the administrators can decide to favor precision over recall or conversely, without requiring any retraining or deep understanding of the base, simply by changing this triggering threshold.

Finally, the mixer combines the signals of the various classifiers via a simple interpretable formula. The influence of each classifier can thus be changed by the system administrator, and additional rules can be manually added so as to change the triggering behavior. An example of the rules that can be embedded in the decision mechanism are rules that promote or demote certain topics based on the *"Black Lists"* and the *"Boost Lists"*. These lists consist respectively of topic categories for which Search Pad *must not* be triggered, (because the topics might be offensive to users for instance) or for which Search Pad can trigger even with weaker signals (because they are more likely to relate to research missions such as health topics for instance).

While it is not recommended to abuse the system and arbitrarily change the different parameters and thresholds, this flexibility is necessary as mentioned before in order to adapt quickly to market needs and changes.

4.2 The Components of the Triggering System

The Mixer. As discussed above, the mixer receives the signals from the three base classifiers. In particular, for a sequence of two consecutive queries q_1, q_2 the mixer receives the following signals:

- $research(q_1, q_2)$. This signal indicates whether the two queries are part of a complex research mission or not. The mixer also receives as auxiliary signal the confidence of such prediction.
- $same\_mission(q_1, q_2)$. This signal indicates whether the two queries are topically coherent and thus susceptible to be part of the same mission. Also in this case, the mixer receives as auxiliary signal the confidence of the prediction.
- $f(q_1)$ and $f(q_2)$ provide the topics of q_1 and q_2, and $s(f(q_1), f(q_2))$ estimates the similarity between these two topics, as per the notation introduced in Section 3.

The users' engagement in research missions is estimated by the total number of queries issued in the session (among other variables). For this system, we fixed the number of queries (k@) to 3, as a rough heuristic for having a long enough session to do our analysis. Hence, the signals produced by the base classifiers for the last 3 queries are kept in a "Session State" and used by the mixer to carry its decision.

Based on all these signals, a formula is learned by means of logistic regression [31]. The formula returns a probability p such that the higher p the more likely it is the current session be a research session. The mixer produces its final recommendation, namely trigger/do not trigger, based on the value of T, the *triggering threshold*, where $T \in [0, 1]$, B a *boosting factor*, whose value belongs to the interval $[1, 5]$.

The decision is made according to the following set of rules:

- if the topics of the research session belong to the previously mentioned *topics black list*, **do not trigger**.
- else if the topics of the research session belong to the previously mentioned *topics boost list* and $p \geq T/B$ (the larger the value of B, the easier the triggering for those topics becomes), then **trigger**
- else if $p \geq T$, then **trigger**.

We next describe the three base classifiers and their signals.

Research Detector and Mission Detector. The Research Detector component is in charge of the classification task behind the signal $research(q_1, q_2)$ described above. The Mission Detector component is in charge of the classification task behind the signal $same\_mission(q_1, q_2)$ also described above. Both the Research Detector and the Mission Detector components are *boosted decision trees* classifiers [31], trained on the same large set of data (approximately 40K pairs of consecutive queries) that we gathered as follows.

We sampled several day-sessions from a subset of Yahoo! Search users during a week toward the end of 2008. The sampling was stratified over the days of the week, so as not to over-represent any particular day of the week. The time period was long enough to capture extended search patterns for some users, exceeding typical 30-minute timeouts, and allowing for missions to extend over multiple days.

A group of annotators were instructed to exhaustively examine each session and to annotate each pair of consecutive queries q_1, q_2 and with the labels $research(q_1, q_2)$ and $same\_mission(q_1, q_2)$. The annotators inspected the entire search results page for each query, including URLs, page titles, relevant snippets, and features such as spelling and other query suggestions. They were also shown clicks to aid them in their judgments.

Both classifiers used a set of 30 features that we computed for each pair of consecutive queries q_1, q_2. Many of these features were shown to be effective for query segmentation [17, 18, 19] and can be summarized as follows:

- **Textual features.** We compute the textual similarity of queries q_1 and q_2 using various similarity measures, including cosine similarity, Jaccard coefficient, and size of intersection. Those measures are computed on sets of stemmed words and on character-level 3-grams.
- **Session features.** We compute the number of clicks and queries in the current session, the number of queries since last click, number of clicks since last queries, etc.
- **Time-related features.** We compute the time difference between q_1 and q_2, the sum of reciprocals of time difference for the pair q_1, q_2, total session time, etc.

The prediction task of the Mission Detector is quite easy, while the one of the Research Detector is hard. Indeed, the Mission Detector achieves a very high accuracy, approximately close to the 95%, while the Research Detector exhibits an accuracy around 75%. The most predictive features for the mission boundaries detection are textual features, among which size of the intersection on character-level

3-grams and cosine similarity computed on sets of stemmed words. Also temporal features play an important role: the closer q_1 and q_2 are in time, the more likely is that they are part of the same mission.

For the research detection, the most relevant features are the session-based ones. In particular, the number of clicks and number of queries since the beginning of the session. Also the length of the queries is a predictive feature: intuitively, a longer query is likely to be a complex query.

Topic Classifier. The last component, the Topic Classifier, is responsible for the signals $f(q_1)$ and $f(q_2)$. For this purpose, we re-used an existing in-house Yahoo! tool as a black box. As mentioned before, most Web search engines have built a similar component for various usages. For each query q it returns a topic category taken from taxonomy of 1026 categories hierarchically organized in a tree with maximum depth 7. Therefore, the similarity among topics, $s(f(q_1), f(q_2))$, is defined as a distance on the tree.
The boosting list described above was created observing the most likely topics returned by this classifier for queries belonging to research missions. We limited the topics in the boosting list to the first two levels of the category tree.

5. EVALUATION

For the training and the evaluation, we took advantage of the editorial services internally available at Yahoo!, and 15 Yahoo! editors manually examined 7,303 users' sessions from a pool of 10,000, each session gathering all the queries issued by a same user over a period of 3 days.

The editors were instructed to decorate each query with one of the following labels (1) research, (2) maybe research (3) not research (4) adult and (5) can't tell. In the case of a "research" label, since research sequences may not necessarily be contiguous, the editors were asked to qualify the label with a counter, namely "research 1", "research 2", etc. in order to make clear which queries belong to which research session.

Finally the editors were given qualitative guidelines to help them decide what a research mission is. It was critical for us to verify that underlying research missions can be qualitatively identified so as for us to devise techniques to quantitatively identify the associated research sessions, which sufficiently good approximation. The guidelines were mostly given "by example" to insist on the qualitative aspects. They included actual examples of research missions and some of their representative queries, some less intuitive than the usual academic, travel or health examples we previously gave, such as:

- shopping research, *e.g.*, "I need to find the best deal on HDTVs".
- political research *e.g.*, "who should I vote for?" or "how do I finally win that argument with my dad about global warming?".
- local research *e.g.*, "I'm trying to choose a karate dojo for my kids" or "what are the best sushi bars in town?".
- how-to research *e.g.*, "I'm learning how to play the guitar" or "I'm collecting recipes for the big pie bake-off next month".

Additional research/not research instances of variations of similar queries were also provided in order to help the editors distinguish between them at a finer grain. Such examples included:

- Not research: a user looking for the correct spelling of "iambic pentameter", since a single fact is sufficient to completely satisfy the need
- Research: a user collecting spelling information on a variety of verse forms, *e.g.*, "trochaic hexameter".
- Not research: a user looking for the score of a particular football game, since this score is final and will not change
- Research: a user collecting the latest stats on the players in their fantasy football team, since these stats will change, and the user will have to conduct similar research again

In order to distinguish between these fine cases, the editors had to go beyond the isolated query expressions and do their best at "guessing" the user's actual needs.

This huge editorial effort allowed us to verify that research sessions were significant enough in Web search traffic to justify special attention. The result of this study was that on average 10% of sessions qualify as research sessions as an underlying research mission could be identified for them by our editors. We then computed the numbers of queries that occurred during these research sessions as compared to the overall number of queries in all sessions and found out that were responsible for more than 25% of query volume. This result alone was in our view sufficient to motivate further study of research missions.

The first type of experiments we had to conduct was to verify that Search Pad would trigger without significantly affecting latency at the Web scale traffic. We conducted in March and April 2009, two "bucket experiments", over 1% and 3% of Yahoo! search traffic to this effect. We verified in both cases that the incurred average latency did not go over 12ms and was thus acceptable from a user's perspective.

We also measured during these experiments that the average number of queries in a research session amounted to 5 queries.

We did not rely on the triggering prompt as validation of search sessions as there is a discoverability issue with such artifacts. Indeed we noticed that the Click-Through Rate (commonly referred to as CTR) remained constant independently of the coverage and this signaled to us that more efforts need to be invested to make the prompt more discoverable. However, an encouraging figure is that we see the number of users steadily increasing.

We also conducted a few live experiments in order to set the a priori value of some of the thresholds we previously introduced and that were added to make the system easily tunable to market needs. We run the system off-line varying the value of the threshold T in $[0.1, 0.2, ..., 0.8, 0.9]$. For each value of T we count the number of triggering events. Following the standard definitions in Information Retrieval, we then measured the precision and recall of the system (in Figures 5 and 6). In particular the precision is the number of triggering events correspondent to research sessions divided by the total number of triggering events. The recall is the number of triggering events correspondent to research sessions divided by the total number of research sessions.

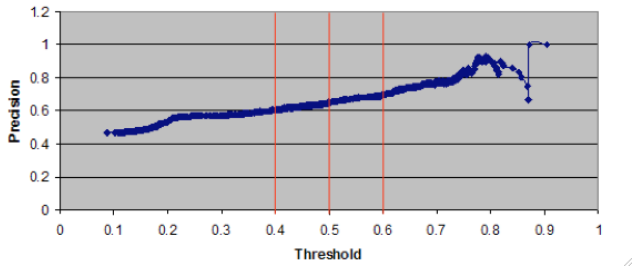

Figure 5: Precision in function of threshold T

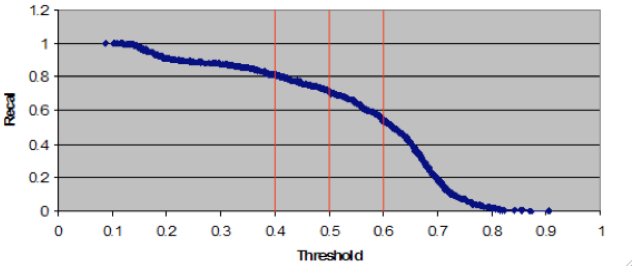

Figure 6: Recall in function of threshold T

These plots allowed us change the threshold in function of the desired recall and precision ratios in given markets. More responsive and early-adopter markets, such as Taiwan for instance, will tolerate higher recall at the cost of lower precision. In Table 1, we report the coverage, defined as the total number of prompts over the total number of searches, for two different values of the threshold T.

We measured the coverage over two different datasets collected by sampling 3% and 1% of the Yahoo! search traffic during the same March, and April 2009 bucket experiments. In the first case, we set $T = 0.5$ and observed a coverage of 11.5%, while in the second case, we set $T = 0.6$ with a corresponding coverage of 6%. A threshold set at 0.6% thus represents the most conservative choice: it guarantees a reasonable precision but a very low recall, given our ground truth estimation that 10% of sessions are research sessions.

Coverage is obviously increased when using a boosting list that favors specific topics. In-house user behavior studies identified 511 categories, *i.e.*, at level 2 in the hierarchical tree classification generated by the topical classifier module. These sub-categories belong to the following 11 main classes: Automotive, Consumer Packaged Goods, Finance, Health Pharma, Issues and Causes, Life Stages, Miscellaneous, Retail, Small Business and B2B, Sport, Technology and Travel. To evaluate the increase in coverage due to the boost list, we simulated off-line the behavior of the triggering component over $6,418$ consecutive query triples and we counted how many of the triggered events are indeed determined by the boosting parameter B. Our simulation showed that 393 prompt events over a total of $2,764$, that is, a relative increase of 14%, can be credited to the boosting list.

6. CONCLUSIONS

We have defined in this paper the concepts of research missions and associated research sessions. Research missions represent a certain type of information needs that require complex and lengthy interaction with search engines. Research session are excerpts of query log sessions that exhibit certain signals indicating that users are undertaking a research mission.

Traffic	US	T	Prompt Coverage
1%	April, 23th	0.6	6%
3%	Mar, 26th	0.5	11.5%

Table 1: Coverage for different values of T

It has been empirically verified through the manual analysis of more than 7000 users' sessions, covering 3-day long activities each, that about 10% of all users' sessions are associated with an underlying research mission, and even more interestingly, that they were responsible for more than 25% of the query volume.

A distinctive characteristic of such research missions is that they are complex and that in many cases, users will return to them over a period of time collecting information that will help them research the topic at hand. The current ephemeral nature of search sessions in Web search engines does not provide any direct support for this type of activities. As this under-addressed need was identified, a novel application was developed, Search Pad, that automatically gathers notes about results visited during such sessions. One unique feature of Search Pad is that it is visibly triggered at run time when the search engine detects that the user is undertaking a research mission. We explained in this paper how we devised an approach for automatically detecting research sessions, and how we embodied it in the triggering component of the Search Pad application.

A few months after the initial launch of Search Pad, we have now sufficient data to verify that detecting research sessions at run time at the scale of Web traffic is feasible and that the quality of our triggering is sufficient for users to continue using and saving Search Pad documents.

We believe that identifying research sessions for the simple purpose of making them persistent is only a very first, yet critical step, in the exploitation of research missions. We plan in the future to continue the studies on research missions in several directions.

First, we would like to continue improving Search Pad in various manners. We would like to refine our understanding of research missions, and increase our recall and precision measures, by finer analysis of users' activities. An immediate improvement, which was mentioned earlier, would be not to measure user's engagement by the simple number of results visited for each query, but by a compound score that would also take into account the time spent by users on these results (too short a time would indicate irrelevance, too long a time that would indicate abandonment of the task) in order to signal that the result contain relevant information (in an implicit relevant feedback spirit).

A better analysis of Search Pad session data would also help us understand the more common usage patterns and thus allow us not only to improve its user experience, but automate the tuning of certain parameters such as the boosting and triggering thresholds, as well as the content of the Boost and Black Lists.

In addition, we would like to study the exploitation of research missions in other contexts and other search activities.

An extremely challenging yet promising one, would be the improvement of search results for queries that are identified as part of a search mission. These queries are typically "hard queries" in the sense that the search engine can verify that the user's information needs were not immediately satisfied. We could learn from common patterns in research sessions over a large population of users. By verifying that a query q_4 has a very high probability to follow queries, q_1, q_2 and q_3, as part of a same research mission, we could for instance, if given sufficient evidence, start bringing good results from q_4 already at the q_3 stage even before the user had a chance to issue q_4. We believe that, in general, considering research sessions as an entity, as opposed to an atomic query, could provide tremendous value in many aspects of ranking.

7. ACKNOWLEDGMENTS

We would like to thank for their valuable suggestions and help many colleagues at Yahoo! (in alphabetical order): Rob Aseron, Ricado Baeza-Yates, Carlos Castillo, Marcus Chan, Vivian Li Dufour, Sarah Ellinger, Ashley Hall, Isabelle Peyrichoux, Flavian Vasile and Shen Hong Zhu.

In addition, we would like to acknowledge the tremendous effort done by the entire Search Pad team at Yahoo! – the application would not have launched without their collective effort – and finally, our users, who by their clicks and queries make our research possible.

8. REFERENCES

[1] D. Abrams, R. Baecker, and M. Chignell. Information archiving with bookmarks: personal web space construction and organization. In *CHI '98: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 41–48, New York, NY, USA, 1998. ACM Press/Addison-Wesley Publishing Co.

[2] Ask mystuff. http://about.ask.com/en/docs/mystuff/tour.shtml.

[3] R. Baeza-Yates. Graphs from search engine queries. In *Theory and Practice of Computer Science (SOFSEM)*, volume 4362 of *LNCS*, pages 1–8, Harrachov, Czech Republic, January 2007. Springer.

[4] K. Bharat. Searchpad: Explicit capture of search context to support web search. In *WWW'00: Proceedings of the 9th World Wide Web Conference*. ACM Press, 2000.

[5] P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna. The query-flow graph: model and applications. In *CIKM'08: Proceeding of the Information and Knowledge Management Conference*, pages 10 pp.+, October 2008.

[6] P. Boldi, F. Bonchi, C. Castillo, D. Donato, and S. Vigna. Query suggestions using query-flow graphs. In *WSCD '09: Proceedings of the 2009 workshop on Web Search Click Data*, pages 56–63, New York, NY, USA, 2009. ACM.

[7] P. Boldi, F. Bonchi, C. Castillo, and S. Vigna. From "dango" to "japanese cakes": Query reformulation models and patterns. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Web Intelligence, (WI 2009)*, pages 183–190.

[8] A. Z. Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.

[9] L. Catledge and J. Pitkow. Characterizing browsing behaviors on the world wide web. *Computer Networks and ISDN Systems*, 6(27), 1995.

[10] Diigo. http://www.diigo.com.

[11] Evernote. http://www.evernote.com.

[12] E. Gabrilovich, M. Fontoura, A. Joshi, V. Josifovski, L. Riedel, and T. Zhang. Classifying search queries using the web as a source of knowledge. *ACM Transactions on the Web*, 3(2):1–28, 2009.

[13] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI'07: Proceedings of the 20th international joint conference on Artifical intelligence*, pages 1606–1611, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[14] Google notebook. http://www.google.com/notebook.

[15] Google sidewiki. http://www.google.com/sidewiki.

[16] Google. Search features. http://www.google.com/help/features.html.

[17] D. He and A. Göker. Detecting session boundaries from web user logs. In *Proceedings of the BCS-IRSG 22nd annual colloquium on information retrieval research*, pages 57–66, Cambridge, UK, 2000.

[18] D. He, A. Göker, and D. J. Harper. Combining evidence for automatic web session identification. *Inf. Process. Manage.*, 38(5):727–742, September 2002.

[19] R. Jones and K. L. Klinkner. Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge mining*, pages 699–708, New York, NY, USA, 2008. ACM.

[20] M. Kellar, C. Watters, and M. Shepherd. A field study characterizing web-based information-seeking tasks. *J. Am. Soc. Inf. Sci. Technol.*, 58(7):999–1018, May 2007.

[21] R. M. Keller, S. R. Wolfe, J. R. Chen, J. L. Rabinowitz, and N. Mathe. A bookmarking service for organizing and sharing urls. *Comput. Netw. ISDN Syst.*, 29(8-13):1103–1114, 1997. Also appeared in the Proceedings of WWW'6, Santa-Clara, CA, USA.

[22] R. Krishnan. Google notebook blog. http://googlenotebookblog.blogspot.com/2009/01/stopping-development-on-google-notebook.html, Jan 2009.

[23] J. Luxenburger, S. Elbassuoni, and G. Weikum. Matching task profiles and user needs in personalized web search. In *CIKM*, 2008.

[24] Y. S. Maarek and I. Z. B. Shaul. Automatically organizing bookmarks per contents. *Comput. Netw. ISDN Syst.*, 28(7-11):1321–1333, 1996.

[25] B. Piwowarski and H. Zaragoza. Predictive user click models based on click-through history. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 175–182, New York, NY, USA, 2007. ACM.

[26] F. Radlinski and T. Joachims. Query chains: learning to rank from implicit feedback. In *KDD '05: Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in*

data mining, pages 239–248, New York, NY, USA, 2005. ACM Press.

[27] M. Richardson. Learning about the world through long-term query logs. *ACM Trans. Web*, 2(4), 2008.

[28] M. C. Schraefel, Y. Zhu, D. Modjeska, D. Wigdor, and S. Zhao. Hunter gatherer: interaction support for the creation and management of within-web-page collections. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 172–181, New York, NY, USA, 2002. ACM.

[29] C. Silverstein, H. Marais, M. Henzinger, and M. Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33(1):6–12, 1999.

[30] J. Teevan, E. Adar, R. Jones, and M. A. S. Potts. Information re-retrieval: repeat queries in yahoo's logs. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 151–158, New York, NY, USA, 2007. ACM.

[31] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (The Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, 1st edition, October 1999.

[32] Yahoo. Search features. http://tools.search.yahoo.com/newsearch/resources.

[33] Zotero. http://www.zotero.org.