Understanding Intrinsic Diversity in Web Search: Improving Whole-Session Relevance

KARTHIK RAMAN, PAUL N. BENNETT, and KEVYN COLLINS-THOMPSON,
Microsoft Research

Current research on Web search has focused on optimizing and evaluating single queries. However, a significant fraction of user queries are part of more complex tasks [Jones and Klinkner 2008] which span multiple queries across one or more search sessions [Liu and Belkin 2010; Kotov et al. 2011]. An ideal search engine would not only retrieve relevant results for a user's particular query but also be able to identify when the user is engaged in a more complex task and aid the user in completing that task [Morris et al. 2008; Agichtein et al. 2012]. Toward optimizing whole-session or task relevance, we characterize and address the problem of *intrinsic diversity* (ID) in retrieval [Radlinski et al. 2009], a type of complex task that requires multiple interactions with current search engines. Unlike existing work on extrinsic diversity [Carbonell and Goldstein 1998; Zhai et al. 2003; Chen and Karger 2006] that deals with ambiguity in intent across multiple users, ID queries often have little ambiguity in intent but seek content covering a variety of aspects on a shared theme. In such scenarios, the underlying needs are typically exploratory, comparative, or breadth-oriented in nature. We identify and address three key problems for ID retrieval: identifying authentic examples of ID tasks from post-hoc analysis of behavioral signals in search logs; learning to identify initiator queries that mark the start of an ID search task; and given an initiator query, predicting which content to prefetch and rank.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Retrieval models, search process*

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Search session analysis, diversity, proactive search, search behavior

ACM Reference Format:
Karthik Raman, Paul N. Bennett, and Kevyn Collins-Thompson. 2014. Understanding intrinsic diversity in Web search: Improving whole-session relevance. ACM Trans. Inf. Syst. 32, 4, Article 20 (October 2014), 45 pages.
DOI: http://dx.doi.org/10.1145/2629553

1. INTRODUCTION

Information retrieval research has primarily focused on improving retrieval for a single query at a time. However, many complex tasks such as vacation planning, comparative shopping, literature surveys, etc., require multiple queries to complete the task [Jones and Klinkner 2008; Bailey et al. 2012].

Within the context of this work, we focus on one specific type of information seeking need that drives interaction with Web search engines and often requires issuing multiple queries, namely, *intrinsically diverse* (ID) tasks [Radlinski et al. 2009]. Informally,

This article revises and extends material originally presented in *Proceedings of the 36th International ACM SIGIR on Research and Development in Information Retrieval (SIGIR'06)* [Raman et al. 2013].
Authors' addresses: K. Raman, (corresponding author) Dept. of Computer Science, Cornell University, Ithaca, NY; email: karthik@cs.cornell.edu; P. N. Bennett, Microsoft Research, One Microsoft Way, Redmond, WA; K. Collins-Thompson, School of Information, 105 S. State Street, University of Michigan, Ann Arbor, MI.

DOI: http://dx.doi.org/10.1145/2629553

Table I. Examples of Intrinsically Diverse Search Tasks

Initiator query	Successor queries
snow leopards	snow leopard pics where do snow leopards live snow leopard lifespan snow leopard population snow leopards in captivity
remodeling ideas	cost of typical remodel hardwood flooring earthquake retrofit paint colors kitchen remodel

Note: Showing the initiator (first) query and several successor (next) queries from the same search session.

an intrinsically diverse task is one in which the user requires information about multiple aspects of the same topical information need. Table I gives examples of two intrinsically diverse tasks based on those observed in commercial Web search engine logs. Intrinsic diversity, where diversity is a desired property of the retrieved set of results satisfying the current user's immediate need, is meant to indicate that diversity is intrinsic to the specific need itself; this is in contrast to techniques that provide diversity to cope with uncertainty in the intent of the query (e.g., the ambiguous query `[jaguar]`).

Intrinsically diverse tasks typically are exploratory, comprehensive, survey-like, or comparative in nature. They may result from users seeking different opinions on a topic, exploring or discovering aspects of a topic, or trying to ascertain an overview of a topic [Radlinski et al. 2009]. While a single, comprehensive result on the topic may satisfy the need when available, several or many results may be required to provide the user with adequate information coverage [Radlinski et al. 2009]. As seen in the examples, a user starting with `[snow leopards]` may be about to engage in an exploratory task covering many aspects of snow leopards, including their lifespan, geographic dispersion, and appearance. Likewise, when investigating remodeling ideas, a user may wish to explore a variety of aspects including cost, compliance with current codes, and common redecorating options. Note that the user may in fact discover these aspects to explore through the interaction process itself. Thus ID search may overlap in some cases with both exploratory and faceted search [Dakka et al. 2006; White et al. 2008]. However, unlike the more open-ended paradigm provided by exploratory search, we desire a solution that is both shaped by the current user's information need and is able to discover and associate relevant aspects to a topic automatically in a data-driven fashion. For example, given the query `[snow leopards]`, our goal is to enable deeper user-driven exploration of that topic by proactively searching for the relevant information that the user might want during the course of a session on that topic, thus reducing the time and effort involved in manual reformulations, aspect discovery, and so on.

To this end, we aim to design a system that addresses two key problems needed for ID retrieval: detecting the start of and continued engagement in an ID task, and computing an optimal set of ID documents to return to the user, given they are engaged in an ID task. For the first problem, the system must be capable of predicting when a user is likely to issue multiple queries to accomplish a task, based on seeing their first "initiator query". To do this, we first develop a set of heuristic rules to mine examples of authentic intrinsic diversity tasks from the query logs of a commercial search engine.

The resulting tasks provide a source of weak supervision for training classification methods that can predict when a query is initiating an intrinsically diverse task or continuing engagement in such a task. With these predictive models, we characterize how ID initiators differ from typical queries. We then present our approach to intrinsically diversifying a query. In particular, rather than simply considering different intents of a query, we incorporate queries that give rise to related aspects of a topic by estimating the relevance relationship between the aspect and the original query. Given the intrinsically diverse sessions identified through log analysis, we demonstrate that our approach to intrinsic diversification is able to identify more of the relevant material found during a session given less user effort, and furthermore, that the proposed approach outperforms a number of standard baselines.

2. RELATED WORK

The distinction between *extrinsic* and *intrinsic* diversity was first made by Radlinski et al. [2009]. In contrast to extrinsically-oriented approaches, which diversify search results due to ambiguity in user intent, intrinsic diversification requires that results are both relevant to a single topical intent as well as diverse across aspects, rather than simply covering additional topical interpretations. Existing methods like maximal marginal relevance (MMR) do not satisfy these requirements well, as we show in Section 6.3. While diversified retrieval has been a popular research topic over many years, much of the research has focused on extrinsic diversity: this includes both learning-based [Yue and Joachims 2008; Slivkins et al. 2010; Santos et al. 2010b; Raman and Joachims 2013] and non-learning-based approaches [Carbonell and Goldstein 1998; Zhai et al. 2003; Chen and Karger 2006; Clarke et al. 2008; Swaminathan et al. 2009; Agrawal et al. 2009; Dang and Croft 2013]. While there has been some work on online learning for intrinsic diversity [Raman et al. 2012], it has been limited to simulation studies and has not addressed the problem of intrinsic diversity in Web search. Recent work [Bailey et al. 2012] indicates that real-world Web search tasks are commonly intrinsically diverse and require significant user effort. For example, considering average number of queries, total time spent, and prevalence of such sessions, common tasks include discovering more information about a specific topic (6.8 queries, 13.5 minutes, 14% of all sessions); comparing products or services (6.8 queries, 24.8 minutes, 12% of all sessions); finding facts about a person (6.9 queries, 4.8 minutes, 3.5% of all sessions); and learning how to perform a task (13 queries, 8.5 minutes, 2.5% of all sessions). Thus, any improvements in retrieval quality that address intrinsically diverse needs have potential for broad impact.

Some previous TREC tracks, including the Interactive, Novelty, and QA tracks, studied intrinsic diversity-like problems in which retrieval effectiveness was partly measured in terms of coverage of relevant aspects of queries, along with the interactive cost to a user of achieving good coverage. While our work shares important goals with these tracks, our task and data assumptions differ. For example, the Interactive tracks focused more on coverage of fact- or website-oriented answers, while our definition of query aspect is broader and includes less-focused subtopics. In addition to optimizing rankings to allow efficient exploration of topics, we also predict queries that initiate intrinsically diverse tasks and show how to mine candidates for ID tasks from large-scale search log data.

Session-based retrieval is a topic that has become increasingly popular. Different research groups have studied trends observed in user search sessions and ways to improve search for such sessions [Guan et al. 2013; He et al. 2013]. For example, Radlinski and Joachims studied the benefit of using query chains in a learning-to-rank framework to improve ranking performance [2005]. Others have studied different means of evaluating search performance at the session level [Järvelin et al. 2008;

Clarke et al. 2009; Kanoulas et al. 2011a; Smucker and Clarke 2012; Sakai and Dou 2013]. Research in this area has been aided by the introduction of the Session track at TREC [Kanoulas et al. 2010, 2011b]: resulting in work on session analysis and classification [Liu et al. 2010; Liu et al. 2011]. Of particular interest is work by He et al., which proposed a random walk on a query graph to find other related queries which are then clustered and used as subtopics in their diversification system [2011]. In our re-ranking approach, we also use related queries to diversify the results, but maintain coherence with the original query. Specifically, we identify a common type of information need that often leads to longer, more complex search sessions. However, in contrast to previous work, rather than using the session interactions up to the current point to improve retrieval for the current query, we use a query to improve retrieval for the user's current and future session. We use sessions from query logs for analysis and evaluate the effectiveness of the proposed methods. While the TREC Session track evaluated the number of uncovered relevant examples for the final query, the emphasis is on the impact provided by the session context up to the present query; in our case, we assume no previous context, but instead are able to characterize the need for intrinsic diversity based on the single query alone.

Session data has also been used to identify and focus on complex, multistage user search tasks that require multiple searches to obtain the necessary information [White et al. 2010; Kotov et al. 2011]. This has led to research on *task-based* retrieval [Hassan et al. 2011; Liao et al. 2012; Hassan and White 2012; Feild and Allan 2013] where tasks are the unit of interest, as opposed to queries or sessions. *Trail-finding* research studies the influence of factors such as relevance, topic coverage, diversity, and expertise [Singla et al. 2010; Yuan and White 2012]. While these problems are certainly related to ours, *tasks* and *trails* tend to be more specialized and defined in terms of specific structures: for example, tasks are characterized as a set or sequence of subtasks to be accomplished, while trails are defined in terms of specific paths of user behavior on the Web graph. However, intrinsically diverse search sessions, for example, as in Table I, represent a broader, less-structured category of search behavior. Similarly, our approach complements work on faceted search [Kohlschutter et al. 2006; Kong and Allan 2013] and exploratory search [Marchionini 2006; White et al. 2006; Qvarfordt et al. 2013] by providing a data-driven manner of discovering common facets dependent on the particular topic.

Query suggestions are a well-established component of Web search results with a vast pertinent research literature: common approaches include using query similarity (e.g., [Zhao et al. 2006; De Bona et al. 2010; Dupret et al. 2010; Guo et al. 2011]) or query-log-based learning approaches (e.g., [Joachims et al. 2007; Dang et al. 2010]). Query suggestions play an important role for intrinsically diverse needs, because they provide an accessible and efficient mechanism for directing users towards potentially multiple diverse sets of relevant documents. Therefore, query suggestion techniques that do not merely provide simple reformulation of the initial query but correctly diversify across multiple facets of a topic, may be particularly helpful for intrinsically diverse needs. Thus our retrieval approach has been partly inspired by recent research on diversifying query suggestions [Ma et al. 2010; Sadikov et al. 2010; Santos et al. 2010a; Song et al. 2011; Fujita et al. 2012], including work that performs clustering of query refinements by user intent for categorical queries. Similarly, work on diversifying results using query reformulations [Radlinski and Dumais 2006; Capannini et al. 2011; Dang and Croft 2012] is also related to our approach.

Our approach is also motivated by recent work on *interactive ranking*. Brandt et al. propose the notion of dynamic rankings, where users navigate a path through the search results to maximize the likelihood of finding documents relevant to them [2011].

Our objective formulation closely relates to another recent work on two-level dynamic rankings [Raman et al. 2011], which studied the benefit of interaction for the problem of extrinsic diversity. Similarly, user interaction has been found to help in more structured and faceted search tasks [Zhang and Zhang 2010; Gollapudi et al. 2011; Pound et al. 2011], in cases such as product search. However, while presenting interactive, dynamic rankings is one user experience that offers a way to surface the improved relevance to users, our techniques are more general: they may be used to present a summary of the topic to the user, recommend unexplored options, anticipate and then crowdsource queries to trade off latency and quality by prefetching, and more.

In contrast to previous work, we provide a way not only to identify complex search tasks that will require multiple queries but to proactively retrieve results for future queries before the user has searched for them. Importantly, these future queries are neither simple reformulations nor completely unrelated, but are queries on the particular task that the user has started. Finally, we introduce diversification methods which, unlike previous methods, maintain coherence around the current theme while diversifying. Using these methods, we demonstrate that we can improve retrieval relevance for a task by detecting an intrinsically diverse need and providing whole-session retrieval at that point.

RELATION to Raman et al. [2013]. This article revises and extends the work in Raman et al. [2013]. In particular, this article makes the following additional contributions. (1) It presents improved evaluation of the session filtering performance using additional annotators, error estimation and further analysis (Section 3.1). (2) It characterizes and contrasts ID sessions with non-ID sessions in terms of different measures of effort and success, among others (Section 3.2). (3) Linguistic and topical traits of ID initiator queries are explored further (Section 4.5). (4) The problem of predicting ID task initiation is studied further, including the effects of training data, model choices and class-bias (Sections 4.6–4.8). (5) A new classification task (Predicting ID task engagement given context) is introduced and studied in detail (Section 5). (6) The proposed ID reranking method is compared with new baselines, evaluated on additional datasets and further analysis provided in terms of the effects of the user model and document set (Section 6.3). This article also provides additional details such as the session-mining algorithm (Appendix A), statistical characterization of ID initiators (Appendix B), proof of the submodularity of the re-ranking objective along with a corresponding approximation guarantee (Appendix C) and the labeling guidelines used for the annotation tasks (Appendix D). To further aid understanding, illustrative examples have been added across different sections including the session-filtering algorithm (Figure 1), ID initiator identification (Table VIII), ID query re-ranking (Figure 13) and the user interaction model (Figure 14).

3. INTRINSICALLY DIVERSE TASKS

An *intrinsically diverse task* is one in which the user requires information about multiple, different aspects of the same topical information need. In practice, a user most strongly demonstrates this interest by issuing multiple queries about different aspects of the same topic. We are particularly interested in identifying the common theme of an intrinsically diverse task and when a user initiated the task. We unify these into the concept of an *initiator query* where, given a set of queries on an intrinsically diverse task, the query among them that is most general and likely to have been the first among these set of queries is called the initiator query. If multiple such queries exist, then the first among them from the actual sequence (issued by the user) is considered the initiator. We give importance to the temporal sequence since the goal is to detect the initiation of the task and provide support for it as soon as possible.

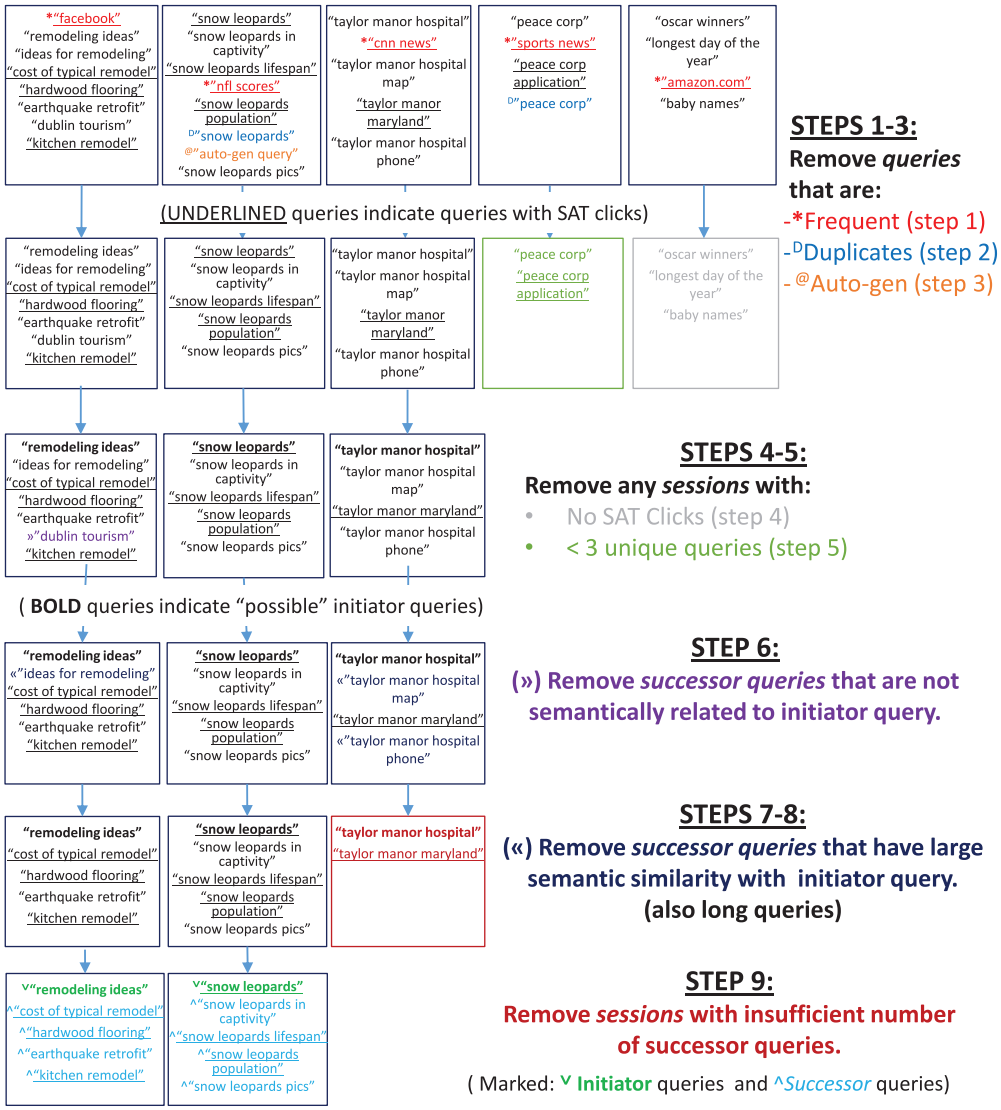

Fig. 1. Illustrative example of the filtering algorithm that mines intrinsically diverse needs within sessions. Each box denotes a session at a stage of processing, with the queries inside the box representing those observed in the session. Different processing steps, along with the corresponding filtering they perform, are marked either by formatted text or text color. Steps 1–3 remove *queries* unlikely to contribute to a session being ID. Steps 4 and 5 remove *sessions* that are unlikely to be ID. Steps 6–8 remove *successor queries* that are semantically unrelated to the initiator or not diverse enough. Step 9 removes sessions such as the red example, which has an insufficient number of syntactically distinct queries to be considered ID. The sessions remaining are considered ID and are marked with the initiator and successor queries.

While previous work has defined the concept of intrinsic diversity, there has been no real understanding or algorithmic development of the problem, including practical means to obtain data related to intrinsically diverse needs. We now identify and analyze authentic instances of intrinsically diverse search behavior extracted from large-scale mining and analysis of query logs from a commercial search engine.

3.1. Mining Intrinsically Diverse Sessions

Intuitively, intrinsically diverse (ID) tasks are topically coherent but cover many different aspects. To automatically identify ID tasks *in situ* where a user is attempting to accomplish the task, we seek to codify this intuition. Furthermore, rather than trying to cover all types of ID tasks, we focus on extracting with good precision and accuracy a set of tasks where each task is contained within a single search session. As a "session", we take the commonly used approach of demarcating session boundaries by 30 minutes of user inactivity [White and Drucker 2007]. Once identified, these mined instances could potentially be used to predict broader patterns of cross-session intrinsic diversity tasks [Kotov et al. 2011; Agichtein et al. 2012], but we restrict this study to mining and predicting the initiation of an ID task within a search session and performing whole-session retrieval at the point of detection.

To mine intrinsically diverse sessions from a post-hoc analysis of behavioral interactions with the search results, we developed a set of heuristics to detect when a session is topically coherent but covering many aspects. These can be summarized as finding sessions that are (1) longer—the user must display evidence of exploring multiple aspects; (2) topically coherent—the identified aspects should be related to the same overall theme rather than disparate tasks or topics; (3) diverse over aspects—the queries should demonstrate a pattern beyond simple reformulation by showing diversity. Furthermore, the user's interaction with the results will be used in lieu of a contextual relevance judgment for evaluation. Thus, we also desire that we have some "satisfied" or "long-click" results where, in line with previous work, we define a satisfied (SAT) click as having a dwell of $\geq$30*s* or terminating the search session [Fox et al. 2005; Gao et al. 2009].

Given these criteria, we propose a simple algorithm to mine intrinsically diverse user sessions. Our algorithm (Alg. 2), which is detailed in Appendix A, uses a series of filters as explained in more detail next. When we refer to "removing" queries during a processing stage, we mean they were treated as not having occurred for any subsequent analysis steps. For sessions, with the exception of those we "remove" from further analysis in Step 4, we label all other sessions as either intrinsically diverse or *regular* (i.e., not ID). We identify the *initiator* query as the first query that remains after all query removal steps, and likewise a *successor* query is any remaining query that follows the initiator in the session. More precisely, we use the following steps in sequence to filter sessions.

(1) *Remove frequent queries*. Frequent queries, such as *facebook* or *walmart*, are often interleaved with other tasks during a session, and such queries can obscure the more complex task the user may be accomplishing. Therefore, we remove the top 100 queries by search log frequency as well as frequent misspellings of these queries.
(2) *Collapse duplicates*. We collapse any duplicate of a query issued later in the session as representing the same aspect of the user's information need, but record all SAT clicks across the separate impressions.
(3) *Only preserve manually entered queries*. To focus on user-driven exploration and search, we removed queries that were not manually entered, for example, those queries obtained by clicking on query suggestion or related search links embedded on a page.
(4) *Remove sessions with no SAT document*. Since we would like to eventually measure the quality of rerankings for these session queries in a personal and contextual sense, we would like to ensure that there is at least one long-dwell click to treat as a relevance judgment. While this is not required for a session being an ID session, we simply require it for ease of evaluation. Thus, we removed sessions with no SAT clicks.

(5) *Minimum aspects explored*. To ensure a certain minimum complexity for the intrinsically diverse sessions, any session having less than three unique queries (i.e., the initiator query + two other queries) was deemed to be regular and not intrinsically diverse.
(6) *Ensure topical coherence (semantic similarity)*. As ID sessions have a common topic, we removed any successor query that did not share at least one common top-10 search result with the initiator query. Note that this need not be the same result for every aspect. Although this restricts the set of interaction patterns we identify, it enables us to be more precise, while also ensuring *semantic relatedness*. This approach also does not rely on the weakness of assuming a fixed, static topic ontology.
(7) *Ensure diversity in aspects*. We would like to avoid identifying trivial query differences, such as simple reformulations or spelling corrections, as different aspects, so we avoid queries that share a very high syntactic similarity with the initiator query. To measure query similarity robust to spelling variations, we consistently use *cosine similarity with character trigrams* in this work, and remove queries where the similarity was more than 0.5.
(8) *Remove long queries*. We observed a small fraction of sessions matching the preceding filters that appeared to consist of copy-and-pasted homework questions on a common topic. While potentially interesting, we focus in this article on completely user-generated aspects and introduce a constraint on query length, removing queries of length at least 50 characters, so as to filter these homework-based queries.
(9) *Threshold the number of distinct aspects*. Finally, to focus on diversity and complexity among the aspects, we threshold on the number of distinct successor queries. We identify a query as distinct when its maximum pairwise (trigram character cosine) similarity with any preceding query in the session is less than 0.6. Any sessions with less than three distinct aspects (including the initiator) are labeled as regular and those with three or more aspects are labeled as intrinsically diverse.

Figure 1 provides an illustrative example of the actions taken by the different steps of the filtering algorithm as well as the resulting sessions. The example regular session in green is typical of sessions when the user reformulates to further specify a need (*application*) the user was likely looking for to begin with (*peace corp*). Also, not all long sessions are ID, as seen in the gray example session, where the user has disjoint information needs.

Putting everything together, we ran this algorithm on a sample of user sessions from the logs of a commercial search engine in the period April 1–May 31, 2012. We used log entries generated in the English-speaking United States locale to reduce variability caused by geographical or linguistic variation in search behavior. Starting with 51.2M sessions comprising 134M queries, applying all but the *SAT-click* filter, with the *Number of Distinct Aspects* threshold at two, led to more than 497K ID sessions with 7.0M queries. These ID tasks accounted for 1.0% of all search sessions in our sample, and 3.5% of sessions having three queries or more (14.4M sessions).[1] Further applying the SAT-click filter reduced the number to 390K. Finally, focusing on the more complex sessions by setting the Number of Distinct Aspects filter to three, reduced this to 146,450 sessions. Varying this threshold leads to different sized datasets, shown in Table II, that we use in subsequent experiments.

Given that ID sessions require multiple queries, we hypothesized that ID sessions account for a disproportionately larger fraction of time spent searching by all users. To

[1]Because we do not focus on more complex ID information seeking, such as tasks that span multiple sessions, the true percentage associated with ID tasks is likely to be larger.

Table II. Different ID Datasets Extracted from the Search Logs by Changing the Threshold of the Number of Distinct Aspects Filter of Alg. 2.

Dataset	Distinct Aspect Threshold	Number of Sessions
MINED2+	2	390K
MINED3+	3	146450
MINED4+	4	55604
MINED5+	5	16527

test this, we estimated the time a user spent in a session by the elapsed time from the first query to the last action (i.e., query or click). Sessions with a single query and no clicks were assigned a constant duration of 5 seconds. Here, the time in session includes the whole session once an ID task was identified in that session. Our hypothesis was confirmed: while ID sessions with at least two distinct aspects represented 1.0% of all sessions, they accounted for 4.3% of total time spent searching, showing the significant role ID sessions play in overall search activity.

To assess the accuracy of our automatic labeling process, we sampled 150 sessions of length at least two queries: 75 each from the auto-labeled regular and MINED2+ intrinsic sets.[2] We ignored single query sessions since those are dominated by regular intents, which may result in a bias in labeling. In total, we had four assessors for this task: two authors (annotators A,B) and two external annotators (annotators C,D). The assessors were given an information sheet (provided in Appendix D.1), which contained instructions similar to the description in the first paragraph of Section 3 as well as examples of ID sessions, such as those in Table I. They were provided with all of the queries, including queries filtered by Algorithm 2, in the session in the order they were issued, and were asked to label each session as regular or ID. The order of the sessions was randomized and the automatic annotation for the session was not revealed to the annotators. This prevents inadvertent annotator bias that happens when all ID or regular sessions are presented together or when the annotator knows what automatic label a session had.

The resulting assessment statistics are shown in Table III. We find that the four assessors had a 68.7% pairwise-agreement with an inter-rater κ agreement of 0.353. At these values of κ, the annotator agreement is moderate, and thus we conclude that the provided labels are reliable. Using each assessor as a gold standard and taking the average, on sessions of length two or greater our extraction method has a precision of 70.4% and an accuracy of 65.5%, as shown in Table IV (the IND results). Note that overall accuracy is higher because single query sessions are always treated as regular. Using the majority labels leads to similar results.

A closer inspection of these results indicates that aspects for some of the filtered sessions were not diverse enough, which can be remedied by tightening Step 7 of the algorithm and using a more stringent threshold for the cosine similarity. Furthermore, we found that annotator D did not fully comprehend the labeling task and labeled only 42 of the 150 sessions as ID, compared to the average of 63 for the other three annotators. Annotator D's labels also tended to agree less with the other three annotators (as seen in Table III) and also disagreed with the labels output by the filtering algorithm (as seen in Table IV). This might be attributed to the need for further details and more examples of ID and non-ID sessions in the labeling instructions.

Overall, with both good agreement and moderate-to-strong accuracy and precision, the filtering method provides a suitable source of noisy supervised labels. Furthermore,

[2]We chose the MINED2+ dataset for this labeling task so as to minimize any bias that the session length may introduce in the annotation.

Table III. Statistics of the Session-Annotation Task with and without the Annotations from Annotator D

Statistic	Value	
	A,B,C	A,B,C,D
Pairwise Agreement	75.34% ±3.42	68.64% ±3.82
Cohen's Kappa	0.507±0.065	0.352 ±0.074
Fleiss Kappa	0.487 ±0.053	0.332 ±0.044

Note: The estimates were computed using 1,000 bootstrap samples of 150 sessions each.

Table IV. Performance of the Filtration Method w.r.t the Session Annotations

Performance	Value	
	A,B,C	A,B,C,D
(IND) Precision	73.85% ±5.63	70.39% ±6.09
(IND) Accuracy	70.06% ±3.69	65.49% ±3.89
(MAJ) Precision	76.77% ±5.15	65.18% ±4.31
(MAJ) Accuracy	73.22% ±3.64	65.31% ±2.97

Note: Estimated using 1,000 bootstrap samples with and without labels from annotator D. Performance is computed using both individual (IND) annotator labels as well as the majority (MAJ) label.

classical results from learning theory tell us that with enough data, we can hope to overcome the noise in the labels, as long as this noise is unbiased, with an appropriate risk-minimizing learning algorithm [Bartlett et al. 2004].

We note that our evaluation of these sessions was limited to a few annotators due to the data coming from real search user sessions and thus containing Personally Identifiable Information (PII). As it can be hard to identify what data is PII and what isn't, we were unable to release the data publicly[3] and thus could not crowdsource the annotation task. However, given the reviewer agreement on this data, as well as the TREC session track data (Section 6.5), we believe that the results provided are representative and likely to hold even with a large-scale evaluation of the algorithm's performance.

3.2. Characteristics of Identified Intrinsically Diverse Sessions

To further understand what differentiates the mined ID sessions from regular sessions, we compared different statistics of the two kinds of sessions. We selected 30K sessions randomly from each to have comparable sampling error. In particular, we considered three different sets of sessions.

(1) *MINED*. A sample of the MINED4+ dataset of size 30K sessions.[4]
(2) *ALLREG*. A sample of 30K regular sessions (with at least one SAT click).
(3) *REG4+*. A sample of 30K regular sessions all of which have at least one SAT click and four queries (to control for the minimum length of session relative to MINED4+).

We characterize statistical differences between the different session types in terms of different quantitative measures, as detailed in Table V. We observe that ID sessions (MINED) tend to involve more user effort, as measured by the average number of queries and total dwell-time, than that displayed for regular sessions (ALLREG), including longer regular sessions (REG4+). Figure 2 visualizes some of these differences. With regard to success while searching, we find that users tend to look for more information in ID sessions, as seen by the increased number of SAT clicks; however, they tend to require more queries and appear to be more selective for these sessions, resulting in a lower number of SAT clicks per query. This also indicates that users are not as satisfied with the quality of search engine results for these queries compared to the other kinds of sessions, indicating potential to improve retrieval for these kinds of sessions. We will further address this problem in Section 6.

[3]For this same reason we are unable to report real examples for the session filtering.
[4]To focus on more complex and (label) noise-free ID tasks, we use the MINED4+ dataset as the primary analysis dataset for ID sessions in the rest of the article.

Table V. Mean and Standard Deviation of Different Session Characteristics Characterizing the Different Kinds of Sessions

Measure of	Session Characteristic	MINED		ALLREG		REG4+	
		Mean	Dev.	Mean	Dev.	Mean	Dev.
Effort	Number of Queries	14.19 (5.21)	10.50 (1.73)	3.10	3.60	6.68	5.48
	(Relative) Total Dwell Time	3.42	3.66	1.00	2.22	2.59	3.07
Success	(Relative) No. of SAT clicks	3.23 (1.34)	2.65 (0.90)	1.00	1.03	2.04	1.69
	(Rel.) Avg. SAT clicks per query	0.79	0.51	1.00	0.65	0.96	0.61
Diversity	Number of Unique Documents (in Top 10)	38.79	10.54	24.21	33.67	58.76	49.15
	Number of Unique Documents Per Query (in Top 10)	7.61	1.13	7.24	2.31	8.86	1.15
Semantic Similarity	Fraction of Top-10 results common to previous queries	0.29	0.13	0.06	0.14	0.12	0.13
	Avg. No. of Top-10 results common to previous queries	2.84	1.24	0.60	1.36	1.19	1.25
Syntactic Similarity	Average Trigram Cosine Similarity to previous queries	0.28 (0.33)	0.12 (0.10)	0.17	0.25	0.26	0.19
	Avg. Trigram Cosine Similarity (among all query pairs)	0.26 (0.33)	0.12 (0.10)	0.16	0.25	0.23	0.18

Note: Numbers in parenthesis refer to those for the ID part of the session only, that is, for the selected queries. Some measures are reported relative to the mean value of the ALLREG sessions.

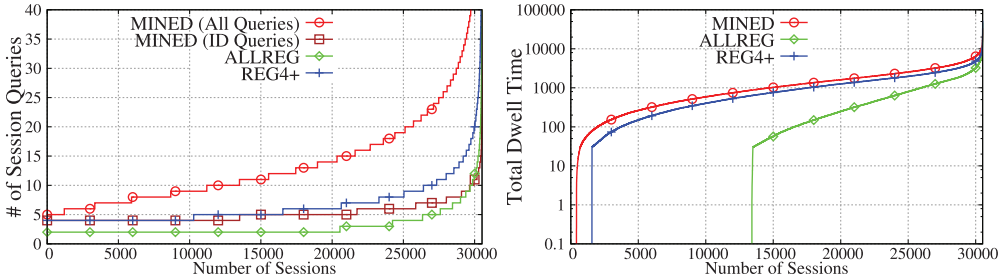

Fig. 2. Differences between different session types for (from left to right) (a) # queries in session, (b) total session dwell time.

Another interesting characteristic of ID sessions relative to regular sessions is seen by comparing the syntactic and semantic similarities between queries in a session. More specifically, we find that the average *syntactic similarity* across queries (via trigram cosine similarity) for ID sessions is quite comparable to that for more complex regular sessions, albeit a bit more than that for the average regular session. However, the semantic similarity measures clearly show that queries of an ID session tend to be far more semantically similar than those in regular sessions. This is also reflected in the *diversity* measures: Despite having more queries, the number of unique documents in the top 10 search results for ID sessions is less than those for long regular sessions and is quite comparable with that for any regular session. Similarly, the number of unique documents per query indicates that nearly 3 of the top 10 documents have been observed at least once previously in the session. As seen in the first row of the table, the number of ID queries in a session (mean: 5.21) comprise only around a third of all queries in ID sessions (mean: 14.19). These statistical observations are more than just a by-product of our extraction algorithm: they are inherent in this kind of search session and corresponding user search behavior. Thus, the previous observation of an increased semantic similarity among queries of an ID session agrees with our choice

of selecting ID session aspects based on semantic similarity, rather than syntactic similarity, in our extraction process (Step 6).

4. PREDICTING INTRINSICALLY DIVERSE TASK INITIATION

Given that we may want to alter retrieval depending on whether the user is seeking intrinsic diversity or not, we ask whether we can train a classifier that accurately predicts when a query has initiated an intrinsically diverse task. While in Section 3 we used the behavioral signals of interaction between the initiator and successor queries of a session to automatically label queries with a (weak) supervised label, here we ask if we can predict what the label would be in the absence of those interaction signals—a necessary ability if we are to detect the user's need for intrinsic diversity in an operational setting. Ultimately our goal is to enable a search engine to customize the search results for intrinsic diversity only when appropriate, while providing at least the same level of relevance on tasks predicted to be regular. Recognizing that, in most operative settings, it is likely important to invoke a specialized method of retrieval only when confident, we present a precision-recall trade-off but focus on the high precision portion of the curve.

4.1. Experimental Setting

Data. We used a sample of initiator queries from the intrinsically diverse sessions described in Section 3.1 (from the MINED4+ dataset) as our positive examples, and the first queries from *regular* sessions were used as negative examples, after removing common queries as in Step 1 of Section 3.1. Note that since the label of a query (e.g., `[foo]`), comes from the session context, it is possible that `[foo]` occurs in both positive and negative contexts. In order to only train to predict queries that were clearly either ID or regular, we dropped such conflicting queries from the dataset; this only occurred in 1 out of every 5,000 ID sessions. Also, to weight each task equally instead of by frequency, we sample by type, that is, we treat multiple occurrences of a query in the positive (resp. negative) set as a single occurrence. Finally, we downsample to obtain a 1:1 ratio from the positive and negative sets to create a balanced set. Unless otherwise mentioned, the dataset was sampled to contain 61K queries and split into an 80/5/15 proportion (50,000 training, 3,000 validation, 8,000 test) with no class bias.

Classification. We used SVMs [Joachims 1999][5] with linear kernels, unless mentioned otherwise. We varied the regularization parameter (C) over the values $\{10^{-4}, 2 \cdot 10^{-4}, 5 \cdot 10^{-4}, 10^{-3}, \ldots, 5 \cdot 10^{2}, 10^{3}\}$. Model selection was done by choosing the model with the best (validation set) precision using the default margin score threshold of 0.

Features. The features are broadly grouped into five classes as shown in Table VI. All features are computed only using the query of interest and not any of the other queries in the session. All features except the Text and POS features were normalized to have zero mean and unit variance. Features with values spanning multiple orders of magnitude, such as the *number of impressions*, were first scaled down via the log function. Due to the large scale of our data, coverage of some features is limited. In particular, query classification was done similar to Bennett et al. [2010] by selecting the top 9.4M queries by frequency from a year's query logs previously in time and then using a click-based weighting on the content-classified documents receiving clicks.[6] Likewise, Stats and QLOG features are built from four months' worth of query logs (December 2011–March 2012) and have limited coverage as a result. Note that the

[5]`http://svmlight.joachims.org/`.

[6]For greater coverage, this could be extended to a rank-weighted back-off as described in Bennett et al. [2010].

Table VI. Different Feature Sets Used for the Identification of Initiator Queries

Feature Set	Examples	Cardinality	Coverage	Normalized	Log-Scale
Text	Unigram Counts	44,140	100%	No	No
Stats	# Words, # Characters, # Impressions, Click Count, Click Entropy	10	81%	Yes	Yes
POS	Part-of-Speech Tag Counts	37	100%	No	No
ODP	Five Most Probable ODP Class Scores from Top Two Levels	219	25%	Yes	Yes
QLOG	Average Similarity with co-session queries, Average session length, Distribution of occurrences within session (start/middle/end)	55	44%	Yes	No

query logs chosen to build these features were from prior to April 2012 to ensure a fair experimental setting with no overlap with the data collection period of the intrinsically diverse or regular sessions. We found the coverage of these features to be roughly the same for both the positive and negative classes. Note that the QLOG features are statistics about the queries computed from the existing query logs. These features include the average length of a session such a query occurs in (since an ID initiator potentially occurs in longer sessions than regular queries), as well as the average similarities with other queries occurring in the same session as this query.

We also note that the cardinality of some feature sets will depend on the training set: for example, the vocabulary size of Text features grows with more training data. The values listed in Table VI are for the default training set of 50,000 queries. Most of our experiments will use all of the five feature sets; the effect of using only a subset of the feature sets is explored in Section 4.3.

4.2. Can We Predict ID Task Initiation?

To begin with, we would like to know the precision-recall trade-off that we can achieve on this problem. Figure 3 shows the precision-recall curve for a linear SVM trained on 50K examples with all the features. The result is a curve with clear regions of high precision, indicating that the SVM is able to identify initiator queries in these regions quite accurately. Furthermore, performance is better than random (precision of 50% since classes are balanced) along the entire recall spectrum.

As Table VII shows, we are able to achieve relatively high precision values at low recall values. For example, we can identify 20% of ID tasks with 79.3% precision. Table VIII provides randomly-chosen examples of initiator queries correctly identified with high confidence. Indicating the efficacy of our method, most of these queries do indeed appear to be exploratory in nature or indicative of deeper, multi-aspect information needs that would likely require multiple queries to satisfy.

Qualitatively, we wish to understand the types of errors in prediction the classifier makes in the high precision region. Table IX contains randomly-chosen examples of regular queries that are predicted with high confidence to be intrinsically diverse queries. While some of them could be considered as potential errors (`[bing maps]`), some could be argued to be intrinsically diverse or exploratory queries in many contexts (e.g., `[how old is my house]`, `[top pit masters in the state]`). This is possible because although our auto-labeling procedure has quite good precision, it may have mistakenly labeled some ID sessions as regular. Thus while predicting some of these queries to be ID initiators may hurt precision according to the auto-labels, it may still benefit by applying diversified retrieval (in Section 6, we will see that is indeed the case).

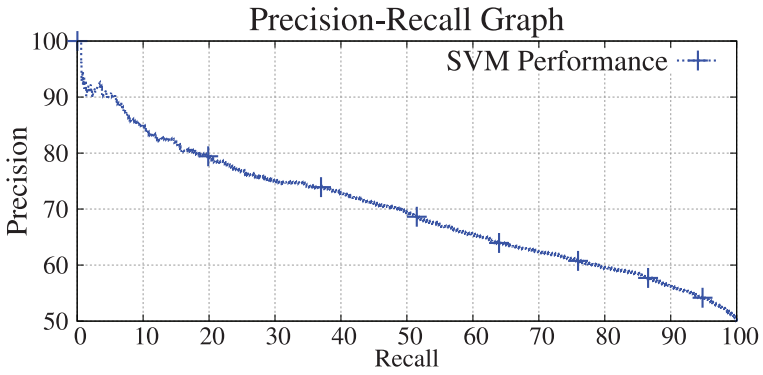

Fig. 3. Precision-recall curve for predicting ID task initiation.

Table VII. Recall at Different Precision Levels and Vice-Versa for Predicting ID Initiation

Recall@Prec		Prec@Recall	
5.9	90	84.9	10
9.8	85	79.3	20
18.3	80	75	30
30.3	75	72.8	40
49.0	70	69.4	50
61.4	65	65.4	60
78.8	60	62.4	70

Table VIII. Examples of Queries Initiating (Auto-Labeled) *ID* Tasks Correctly Predicted to be ID with High Confidence

Prec	Queries
>90	main character gears of war different types of cattle dogs facts about bat eared fox what is lobelia used for
~90	queen isabellas work during renaissance kingman paintball gun parts where is galileo buried what kind of diet does a clownfish have
~80	rms sinking of titanic roll runners for stairs and hallways gamboa rainforest resort panama bill cosby recordings

Table IX. Examples of Queries Initiating (Auto-Labeled) *Regular* Tasks Incorrectly Predicted to be ID with High Confidence

Prec	Queries
>90	adobe flash player 10 activex bing maps live satellite aerial maps how old is my house
~90	port orchard jail roster was is form 5498 java file reader example free ringtones downloads
~80	life lift top pit masters in the state promag 53 user manual pky properties llc nj

4.3. Which Features Were Most Important?

We next investigate the effect of using different subsets of the features on performance. The results are shown in Figure 4 and Table X. First, we note that Stats, QLOG, and ODP feature sets help identify only a small fraction of the initiator queries but do so with high precision. On the other hand, the Text and POS feature sets, which have high coverage, provide some meaningful signal for all the queries, but cannot lead to high precision classification. We also find that some combinations of features, such as Text and Stats, complement each other leading to higher precision (as well as higher recall) than is obtainable with either feature type alone. In fact, the Text and Stats combination performs almost as well as using all features (which in turn was the best of all feature combinations).

At this point, it is worth noting that while the Stats, QLOG, and ODP feature sets don't result in high recall, this is due in large part to their limited coverage: they can help distinguish class labels only for those queries that were seen in previous query logs. Thus, higher coverage of these features by using larger samples of query logs and smoothing the query classification, as described in Bennett et al. [2010], can only help improve performance for this classification task.

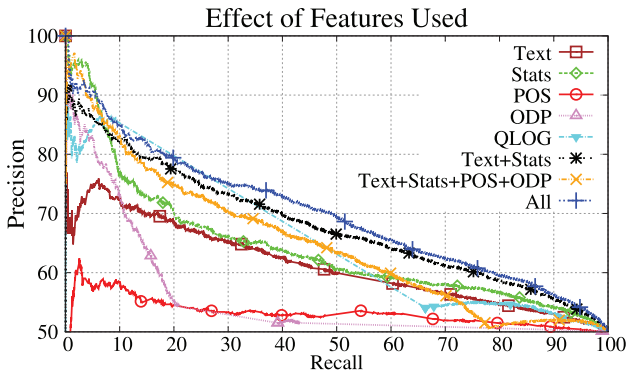

Fig. 4. Change in classification performance of ID initiation as feature sets are varied.

Table X. Effect of Feature Set on Precision & Recall

Feature Set	Rec@ 80%Prec	Prec@ 40%Rec
T	0.1	62.6
S	9.2	63.7
P	0.0	52.8
O	5.6	51.6
Q	9.4	54.1
TS	13.6	69.7
TSPO	12.2	67.0
TSPOQ	18.3	72.8

Note: T = Text, S = Stats, P = POS, O = ODP, Q = QLOG

Table XI. Top 10 Parts-of-Speech and Words with High Positive Association with ID Initiators According to Log-Odds Ratio (LOR)

Part of speech	LOR	Words	LOR
Wh-pronoun	0.41	information	1.64
Proper noun, singular	0.40	forms	1.59
Particle	0.27	account	1.50
Adjective, superlative	0.16	facts	1.45
Verb, present tense, 3rd person singular	0.14	log	1.39
Other	0.14	did	1.34
Noun, plural	0.13	army	1.22
Verb, past participle	0.12	manual	1.18
Preposition or subordinating conjunction	0.09	login	1.17
Cardinal number	0.09	form	1.16

Note: Results are restricted to tags and counts with ≥50 occurrences.

4.4. Linguistic Features of Initiator Queries

To further understand ID initiator queries, in Table XI we identified the part-of-speech and text features most strongly associated with them, by computing each feature's log-odds ratio (LOR)[7] compared to regular queries. Looking at the top-ranked features by LOR, we found that initiator queries are more likely to use question words (LOR = 0.41); focus on proper nouns (0.40) such as places and people; use more 'filler' words (particles) found in natural language (0.27); use fewer personal pronouns (LOR = −0.32) and when they use general nouns, these tend to be plural (0.13) instead of singular (−0.052). Predominant text features indicated the importance of *list-like* nouns such as *forms, facts, types, ideas* (LOR = 1.59, 1.45, 1.25, 0.92);[8] verbs that are commonly used in questions such as *did* (1.34); and words indicating a broad need such as *information* and *manual* (1.64, 1.18). Strong negative features tend to encode exceptions: for example, the word with most negative LOR *lyrics* (−2.25) is typically used to find words to specific songs.

[7]The LOR can be thought of as an approximation to the weight in a single-variable logistic regression.
[8]"Types" and "ideas" had slightly less than 50 occurrences and thus do not occur in Table XI.

Table XII. ODP Classes with Highest and Lowest LOR Association with ID Initiators

Level-1 Class	LOR	Notable sub-classes	LOR	Level-1	LOR	Sub-class	LOR
Science	0.29	Biology	0.72	Adult	−0.82	Images	−0.91
		Social-Sciences	0.24	News	−0.37	Newspapers	−1.05
Computers	0.27	Internet	0.55	Sports	−0.33	Basketball	−0.13
		Software	0.24	Arts	−0.24	Performing Arts	−1.09
Health	0.26	Pharmacy	1.10			Music	−0.53

Note: Results are restricted to classes with ≥50 occurrences.

4.5. Topics Associated with Intrinsically Diverse Tasks

We would like to know which topical categories are typically associated with initiator queries. Table XII shows the results of an LOR study of the ODP features, similar to that done for the linguistic features. We find that classes with information-seeking queries such as *Science* and *Health* (e.g., extensive searching on software problems, drug & information effects) tend to occur more frequently in ID sessions. On the other hand, targeted search categories like *News*, *Sports*, and *Music* (e.g., find an article on a current event, read game outcomes, find music lyrics) tend to be negatively associated with ID initiators.

4.6. How Much Training Data is Needed for Predicting ID Initiation?

Although we can achieve high precision for low recall values, we seek to understand how the quantity of training data impacts the precision-recall trade-offs we can obtain. To evaluate this, we varied the training set size while keeping the validation and test sets unchanged. In all cases we ensured there was no overlap between any of the training sets and the validation/test sets. To control the variance that results from changing both the training set size and the datapoints, the training sets were generated so that a training set of size N was a superset of all training sets of size less than N.[9] The results in Figure 5 show how the precision-recall curve changes when varying the quantity of training data.

We find that using more training data allows us to achieve higher precision for a fixed recall. While there is minimal learning possible for the small training data sizes, as the amount of training data increases, the precision improves significantly. Table XIII simply highlights this further, with large increases in recall numbers at the 80% precision mark, and large increases in precision at the 40% recall mark. Thus, we find using more data to train the models can further help obtain higher-precision numbers.

More generally, we expect the Stats and other non-Text features to prove most useful when training data is limited. With an increase in training data, we expect the Text features to become more important and to lead to better classification.

4.7. Model Choices for SVM Prediction

We investigated the sensitivity of predictions to the choice of kernel used in the SVM classifier. In particular, we considered the polynomial kernel of degree 2 and the radial basis (RBF) kernels, as compared to the default linear kernel. Figure 6 and Table XIV show the results for the different kernels. While using the higher-order kernels improved recall at the very highest precision levels, their results got progressively worse at increasing recall for lower precision levels, while also being computationally more expensive. We therefore used linear kernels for the remainder of our experiments.

Since the only tunable parameter in our setup is the regularization parameter C in the SVM, we examined the sensitivity of performance to the value of C. This also gives

[9]To obtain the largest training set size of 200K queries, we added queries sampled from the MINED3+ set.

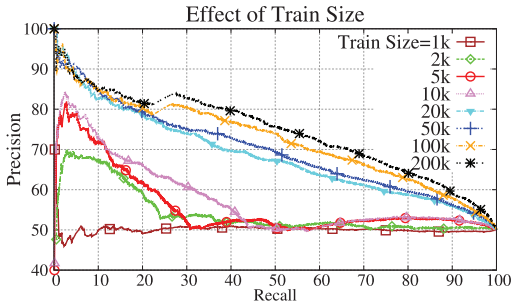

Fig. 5. Effect of changing the training set size on classification of initiator queries.

Table XIII. Change in Precision and Recall on Increasing Training Data Size

Train Size	Rec@80%Prec	Prec@40%Rec
1k	0.0	51.0
2k	0.0	52.0
5k	2.9	52.2
10k	4.9	55.3
20k	17.5	69.7
50k	18.3	72.8
100k	31.1	76.8
200k	39.3	79.6

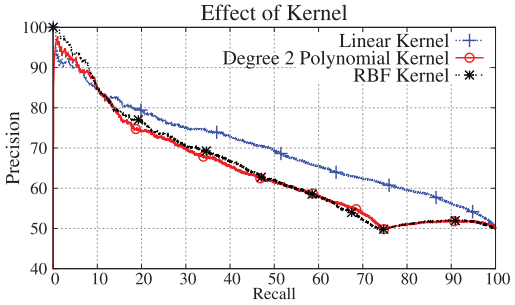

Fig. 6. Change in classification performance of initiator queries on varying the kernel.

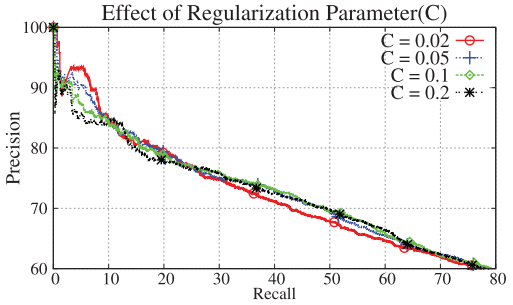

Fig. 7. Change in classification performance on varying the SVM regularization parameter.

us an idea of the degree of overfitting, since worse performance at higher C values indicates significant overfitting—a higher C places more emphasis on fitting the data and less on the regularization norm. Figure 7 and Table XV summarize these results. Apart from minor variations, classification performance was roughly the same across different values of C. This held true in other settings as well, indicating the prediction is not overly sensitive to this parameter value. We also noted interesting variations in precision across the parameter values: lower C values typically lead to initially higher precision, but with performance dropping off faster than that for higher choice of C. This corresponds with our intuition that stronger regularization should initially lead to higher precision, but eventually result in weaker performance for the more uncertain examples later.

4.8. The Impact of Class Skew on Predicting ID Task Initiation

An assumption made so far was that there were equal numbers of positive examples and negative examples, that is, a balanced dataset. However, in a practical setting, this ratio is likely to be skewed towards having many more negative examples. To begin with, we studied the effect of using models trained on balanced sets but tested on an unbalanced test set. More specifically, we changed the class ratio for the test (and validation) set by simply adding more negative examples to the original validation and test sets. Figure 8(a) shows how performance varies with the class ratio (skew) in the test set. Not surprisingly, increasing skew causes performance to drop. However, in all cases, we outperform a random classifier (1:1 = 50% prec, 1:2 = 33%, 1:5 = 16.7%, 1:15 = 6.25%). Estimating the class skew correctly in the training set helps improve classification

Table XIV. Change in Precision and Recall on Changing Kernel

	Linear	Poly-2	RBF
Rec@90%Prec	5.9	5.9	8.1
Rec@80%Prec	18.3	13.8	13.5
Prec@10%Rec	84.9	84.8	84.4
Prec@40%Rec	72.8	65.5	66.7

Table XV. Effect of Regularization Parameter

C	Rec@80%Prec	Prec@40%Rec
0.02	19.0	71.2
0.05	18.3	72.8
0.1	16.6	73.1
0.2	14.8	72.4

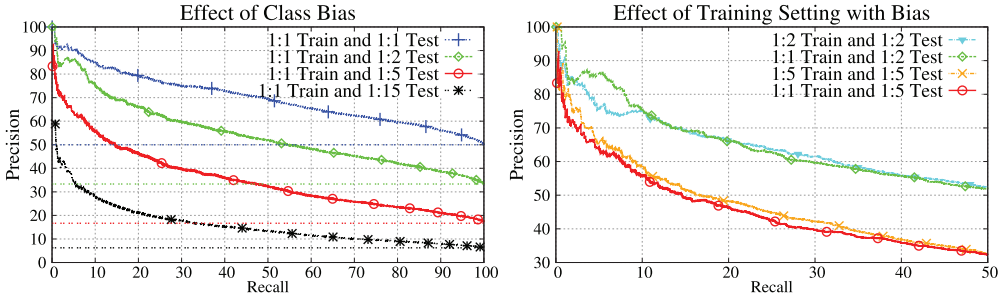

Fig. 8. (a) Effect of class bias on ID initiator classification performance of initiator queries for models trained on balanced datasets; (b) effect of difference in train-test class biases on classification performance of initiator queries.

performance somewhat, as shown in Figure 8(b). However, further improvements may require the use of cost-sensitive classifiers or the use of alternate learning techniques that are well suited to such imbalanced settings. Incorporating additional training data could also lead to improvements, as seen earlier in Section 4.6. Using cascaded models and more sophisticated feature sets could also lead to further improvements in classification accuracy.

4.9. ID Task Identification in Operational Settings

In a practical setting, we would have to determine on-the-fly if a user-issued query is initiating an ID session or not. Queries that are predicted to be ID initiators could have their retrieval process altered, for example, by reranking results, as in Section 6. Thus, to estimate the practical impact of any alteration to the retrieval process, we create datasets of sessions that are *predicted* to be ID based on their initiators. More specifically, we mixed an equal[10] number of (mined) ID sessions (sampled from either MINED3+, MINED4+, or MINED5+) and regular sessions. We then used an SVM classifier[11] on this mixture of sessions to identify those predicted to be ID, based on their initiator query, resulting in a PREDID dataset. Table XVI provides statistics of the different Predicted ID datasets. The accuracy of the resulting datasets is similar to the classification results from earlier in this section.

An interesting finding is that the classification accuracy improves on increasing the *number of distinct aspects* threshold used to train the SVM. This implies that sessions with more aspects are likely easier to identify. This also suggests that we could train a regression model that, given an initiator query, could estimate the number of query aspects the user is interested in finding. This in turn could potentially be used to inform the reranking/prefetching that is performed.

[10]We chose an unbiased mixture, as the classification results in Section 4.8 indicated deterioration in accuracy on adding class skew, which would require further training data to rectify.
[11]Trained using a 45-10-45 training-validation and test split of the mixed set.

Table XVI. Session Datasets Predicted to be ID (PREDID) When an SVM Was Run on a Mixture of Regular Sessions and Those from a Corresponding MINED Dataset

Dataset	Total Sessions		Accuracy
PREDID5+	6,501	(4,470 from MINED5+, 2,031 from Regular)	68.8%
PREDID4+	22,238	(14,960 from MINED4+, 7,278 from Regular)	67.3%
PREDID3+	59,013	(38,687 from MINED3+, 20,326 from Regular)	65.6%

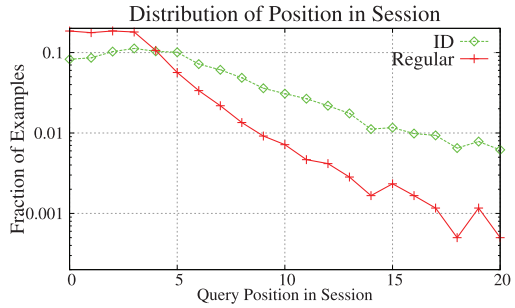

Fig. 9. Distribution of query positions in session across both classes for (all) intrinsic diversity query identification.

5. USING CONTEXT TO PREDICT ID TASK ENGAGEMENT

The previous section (Section 4) discussed how we can identify queries that initiate intrinsically diverse sessions. While such identification could be used for altering retrieval for *initiator* queries, it does not address the problem of the subsequent *successor* queries issued by the user, in the case where the user is not fully satisfied by the improved retrieval for the initiator. If we want to alter retrieval for all queries of an ID session as well, then we need to be able to identify any query within an ID session. In this section, we address this more general question: *Can we identify if a query is part of an intrinsically diverse session?* While studying this classification task, we focus on the effect of *context*. In particular, we will demonstrate that short-term context, in the form of using previous queries in the session, can greatly help us identify a user's need for intrinsic diversity.

5.1. Experimental Setting

Data. Similar to the initiator identification from Section 4, we sampled 30.5K queries from the MINED4+ ID session dataset as our positive examples. More specifically, we sampled a random query from within the *intrinsically diverse part* of each session. For the negative examples, we sampled 30.5K random queries from regular sessions. To ensure fairness in terms of the availability of the context across the two classes, we only considered regular sessions that were of length 4 or more, that is, using the REG4+ session dataset from Section 3.2. Figure 9 shows the distribution of the positions (within session) of the 61K positive and negative examples considered for this task. While the positive examples tend to have a little more context, the distributions are not significantly different across the two classes.

In the construction of this dataset, we again dropped queries that occurred as both positive and negative examples. As with initiator queries, this was an infrequent event, occurring less than once for every 5K queries. We also downsampled to obtain a balanced dataset, containing 30.5k positive and negative examples each. The dataset was

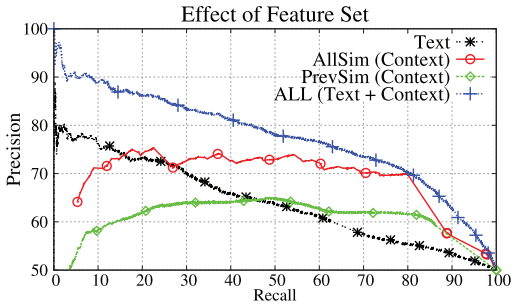

Fig. 10. Effect of contextual features on classification of (all) ID queries.

Table XVII. Recall at Different Precision Levels and Vice-Versa for Predicting All ID Queries Using ALL Features Compared to Using Only Text Features

Recall @ Precision			Precision @ Recall		
ALL	Text	Prec	ALL	Text	Rec
8.6	0.0	90	88.9	76.2	10
25.5	0.4	85	86.2	73.2	20
44.6	2.4	80	83.3	70.0	30
65.1	13.3	75	81.2	65.9	40
80.6	30.0	70	78.1	63.7	50
87.2	44.5	65	76.6	61.1	60
92.7	62.9	60	73.5	57.4	70

then split into a training set of size 44k, a validation set of size 5k, and a (balanced) test set of size 12k examples[12] for which performance is reported.

Classification. We again used SVMs for this classification problem and performed model selection using the validation set.

Features. Our goal in this section was to study the effect of query context. Thus, in addition to the standard context-independent bag-of-words (Text) features, we used two context-dependent feature sets (AllSim and PrevSim).

(1) *Text*. Unigram count of query terms. This led to a set of $\sim 44K$ features. While this is a significantly larger set of features than the context-dependent feature sets, unlike the context-dependent features, these features are sparse.
(2) *AllSim*. This comprises a set of five features. These features measure the trigram cosine similarity of the current query with all previously issued queries in the session. Each of the query similarities is placed in one of five equal-sized buckets: in this study, we used buckets with ranges $[0, 0.2]$, $(0.2, 0.4]$, $(0.4, 0.6]$, $(0.6, 0.8]$, $(0.8, 1.0]$. The fraction of the similarities with the previously issued queries that lie in each bucket correspond to the values of the five features belonging to this feature set. For example, a query in position 5, with similarity values of 0.1 (first query), 0.9 (second query), 0.35 (third query), and 0.7 (fourth query) would have a vector for these five features of $(\frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4})$.
(3) *PrevSim*. This is another set of three context-dependent features. However, this uses the immediate context of the three previous queries. In particular, the average trigram cosine similarity with the previous 1, 2, and 3 queries gives us the three different feature values. For the preceding example, these values would be $(0.7, \frac{0.7+0.35}{2}, \frac{0.7+0.35+0.9}{3})$.

Since our focus was primarily on the effect of context, we did not consider other kinds of features, such as those used in the earlier section. As all the features fall in a similar range, we did not perform any normalization or rescaling.

5.2. Can We Predict ID Task Engagement? Does Context Help?

We again plot the precision-recall trade-off observed on using the SVM in Figure 10. The black (Text) curve shows the performance on using just the unigram Text features.

[12]We chose a larger test set here so as to better estimate the effect of context and dependence of performance on position within session.

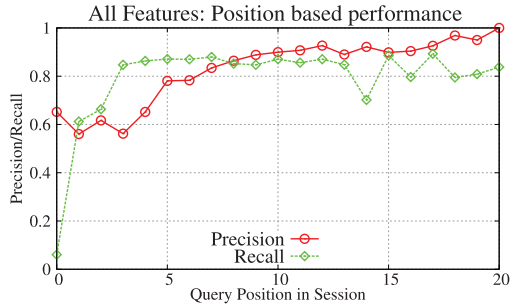

Fig. 11. Effect of query position for classification of all intrinsically diverse queries using all features.

While it performs better than random, there is room for improvement, as we can only identify 73.2% of the ID queries correctly at 20% recall levels, as shown in Table XVII.

However, on adding the eight *context-based* features (five AllSim + three PrevSim features), we find a significant improvement in classification performance, as shown in Figure 10 from the blue (ALL) curve. Classification precision improves to 86.2% (from 73.2%) at the 20% recall level. In fact, we find an improvement at all levels of precision and recall with the addition of the contextual features, thus confirming the value of context for this classification task.

Figure 10 also shows the classification performance achieved when using only the individual context feature sets. We find that for both kinds of context-based features, performance is (largely) better than random. In particular, we find that an SVM using only the AllSim features tends to be quite competitive compared to the Text-only SVM. We observe a >70% precision up to recall levels close to 80%, compared to the 30% obtained by using only Text features, indicating that these features are relatively robust and informative. These results also indicate that using contextual information from across the whole session is more potent than using just the immediate history of previous queries. Incorporating more context features may further improve classification performance. Similarly, using more sophisticated features, such as those used in the literature for identifying queries on the *same task* [Radlinski and Joachims 2005; Jones and Klinkner 2008; Kotov et al. 2011; Lucchese et al. 2011] may also be beneficial.

5.3. Effect of Query Position in Session on Performance

We would like to determine the effect of the query's position within a session on the classification performance of the SVM. As shown previously, position becomes particularly important given the important role context plays, since the later a query occurs in a session, the more context we have available. Figure 11 shows how precision and recall vary with the query position for the SVM classifier that uses all features. We find that the later the query occurs in the session, the larger the precision of the classification. In particular, we observe near-perfect precision for later query positions. Recall also improves drastically from the case of no context, that is, the 0th position (∼6% recall) to the case of having three queries as context (∼85% recall). Note that we do not provide any information about the query position as a feature to the SVM.

Next, we studied how this effect varies for the different feature sets. Figure 12 shows how precision and recall change for the SVM trained on each one of the different feature types. For both sets of contextual features, we find that both precision and recall increase rapidly with a small amount of context. Surprisingly, we find that for the text features as well, precision continues to increase. This may be attributed to the increase in query length typically seen later in the session as more *topical* terms are added to refine the search. A key difference that is evident is the difference in recall

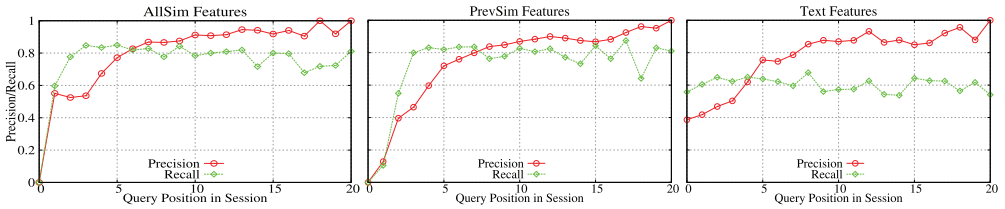

Fig. 12. Effect of query position on classification performance of all intrinsically diverse queries using (a) AllSim features, (b) PrevSim features, (c) Text features.

level between contextual features (recall around 80%) and text features (recall around 60%), reinforcing the earlier observation that as more context becomes available, the contextual features become more valuable.

Implications. The results from this section clearly show that added context can greatly help improve the identification of ID session queries. Thus, these contextual signals can help in determining if a session is ID, and in turn help trigger the use of alternate rankings for such ID sessions, such as the reranking method proposed in Section 6. This approach can also be used to trigger a switch back to the regular rankings when the classifier detects that the ID part of the session is likely to be complete, or to avoid ID-optimized rankings for off-topic queries that occur in such sessions.

6. RERANKING FOR INTRINSIC DIVERSITY

So far, we have discussed mining authentic ID sessions from search logs and the identification of ID initiator and successor queries that lead to, and occur in, ID tasks, respectively. In this section, we focus on changes that can be made to the search results page to support retrieval for queries in intrinsically diverse tasks. As ID tasks tend to be complex and involve significant user effort (a finding supported by the statistics in [Bailey et al. 2012]) we would like to reduce user effort and correspondingly time spent searching, as emphasized in recent work [Smucker and Clarke 2012; Sakai and Dou 2013]. To this end, we propose a reranking scheme that aims to reduce user effort by satisfying the information need of both the current query issued as well as future queries that the user is likely to issue on other aspects of the task. To the best of our knowledge, we are the first to address this problem of *pro-active search augmentation*, that is, jointly satisfying the current query as well as future queries. This is in contrast to the work on anticipatory search [Liebling et al. 2012], which focuses solely on the latter.

We base our approach on an interactive ranking of aspect-document pairs. Given an issued query representing the start of an ID task with multiple aspects, we consider rankings where each result can be attributed to some aspect. We represent each aspect of the ID task using a *related query* of the issued query. One way this could be surfaced on a results page for a user is in a manner similar to the two-level rankings proposed in Raman et al. [2011], where the related query (representing a specific aspect) is placed adjacent to its corresponding search result (representing the best result for that aspect). In such a setting, clicking on the related query could lead to the full set of results for that query being presented, thus enabling the user to explore documents for that aspect. This leads to the question of how to find such a joint ranking.

6.1. Ranking via Submodular Optimization

We first describe precisely what we consider as an interactive ranking. In response to an initial query q, an interactive ranking $\mathbf{y} = (\mathbf{y}_D, \mathbf{y}_Q)$ comprises two parts: a ranking of documents $\mathbf{y}_D = d_1, d_2, \ldots,$ which we refer to as the *primary* ranking; and a corresponding list of related queries $\mathbf{y}_Q = q_1, q_2, \ldots,$ which represent the *aspects* associated

with the documents of the primary ranking. The ith query in the list, q_i, represents the aspect associated with d_i. Structurally, $\mathbf{y}$ can also be thought of as a ranked list of (document, related query) pairs $(d_i, q_i)_{i=1,2,\ldots}$.

Given this structure, let us consider four conditions that comprise a good interactive ranking of document-related query pairs.

(1) Since the documents d_i in the primary ranking were displayed in response to the issued query q, they should be relevant to q.
(2) As document d_i is associated with the aspect represented by the related query q_i, document d_i should also be relevant to query q_i.
(3) Aspects, represented by related queries q_i, should be relevant to the ID task being initiated by the query q.
(4) At the same time, the aspects should not be repetitive, that is, there should be diversity in the aspects covered.

We now design a ranking objective function that satisfies these four conditions to jointly optimize the selection of documents and queries $(\mathbf{y}_D, \mathbf{y}_Q)$. Suppose we have an existing interactive ranking $\mathbf{y}^{(k-1)}$ that has $k-1$ (document, related query) pairs; our goal is to construct a new ranking $\mathbf{y}^{(k)}$ by adding the optimal (document, related query) pair to $\mathbf{y}^{(k-1)}$: an operation we denote by $\mathbf{y}^{(k)} = \mathbf{y}^{(k-1)} \oplus (d_k, q_k)$.

Condition 1 can be met by selecting d_k such that $R(d_k|q)$ is large, where $R(d|q)$ denotes the probability of relevance of document d given query q. Condition 2 can be met by selecting d_k such that its relevance to the corresponding related query q_k, $R(d_k|q_k)$, is large. Conditions 3 and 4 imply a standard diversification trade-off, but here we have that the aspects q_k should be related to the initial query q and diverse. If we use a similarity function $Sim(\cdot,\cdot)$ between queries to estimate the relevance between queries, Condition 3 implies that the similarity between q_k and q should be large. Condition 4 requires that the diversity should be maximized between q_k and all previous queries $\mathcal{Q}^{(k-1)} = q_1, \ldots, q_{k-1}$. Both Conditions 3 and 4 can be jointly obtained by optimizing an MMR-like *diversity function* [Carbonell and Goldstein 1998], $Div_\lambda(q_k, \mathcal{Q}^{(k-1)})$, described as follows:

$$Div_\lambda(q_i, \mathcal{Q}^{(k-1)}) = \lambda \cdot Sim(q_i, Snip(q)) - (1-\lambda) \max_{q' \in \mathcal{Q}^{(k-1)}} Sim(Snip(q_i), Snip(q')), \quad (1)$$

where $\lambda \in [0,1]$ controls the trade-off between relevance of the related query aspect and diversity across aspects. In this study, we define $Sim(a,b)$ as the cosine similarity between word-TF representations of a and b, and $Snip(q')$ is the bag-of-words representation of caption text from the top-10 search results for a simple relevance-based retrieval for q' (i.e., using $R(d|q')$ alone).

We now need to combine these different mathematical terms to obtain a joint objective function. Intuitively, we would also like the change in the objective function on adding a document-query pair (d_i, q_i) to the ranking $\mathbf{y}$ to be no smaller than what we would gain if adding the pair to a larger ranking $\mathbf{y} \oplus \mathbf{y}'$: that is, the objective function should be *monotone* and *submodular*. Consider the following objective function:

$$F_{\beta,\lambda}(d_1, q_1, \ldots, d_n, q_n) = \sum_{i=1}^{n} R(d_i|q) \cdot R(d_i|q_i) \cdot e^{\beta Div_\lambda(q_i, \mathcal{Q}^{(i-1)})}, \quad (2)$$

where $\beta > 0$ is a parameter controlling the rate at which returns diminish from additional coverage. This parameter, along with the diversity trade-off parameter λ, need to be learned during training. The $Div_\lambda(\cdots)$ term appears within the exponent to ensure the objective is monotone, which in turn leads to the following theorem.

THEOREM 6.1. *The resultant objective $F_{\beta,\lambda}(\cdot)$ is a submodular function.*

The proof is provided in Appendix C. To compute the interactive, document-relevant query ranking given query q, we need to optimize this objective F:

$$\mathbf{y} = (\mathbf{y}_D, \mathbf{y}_Q) = \text{argmax}_{(d_1,\ldots,d_n),(q_1,\ldots,q_n)} F_{\beta,\lambda}(d_1, q_1, \ldots, d_n, q_n). \tag{3}$$

This optimization problem can be interpreted as maximizing an expected utility (the exponential term involving $Div_\lambda(\cdot)$) of covering related and diverse aspects where the expectation is over the maximum joint relevance of a document to both the initial query and the related query aspect. Furthermore, the joint probability is assumed to be conditionally independent to factor into the two relevance terms. Note that while the final objective optimizes for an interactive ranking, the primary ranking itself aims to present results from other aspects, which we validate empirically in Section 6.4.

ALGORITHM 1: Greedy-DynRR(Query q; Relevance $R(\cdot|\cdot)$; Documents $\mathcal{D}$; Params β, λ)

```
1:  (y_D, y_Q) ← (φ, φ)                                   ▷ Initialize to be empty
2:  for all q' ∈ RelQ(q) do                               ▷ Iterate over aspects/related queries
3:      Next(q') ← Rank documents in D by R(·|q) · R(·|q')   ▷ Ordering for q'
4:  for i = 1 → n do                                      ▷ Get ranking of size n
5:      bestU ← −∞
6:      for all q' ∈ RelQ(q) \ y_Q do                     ▷ Iterate over queries not in y_Q
7:          d' ← Top(Next(q') \ y_D)                      ▷ Highest document not in y_D
8:          v ← R(d'|q) · R(d'|q') · e^{β Div_λ(q', y_Q)}   ▷ Marginal benefit
9:          if v > bestU then                             ▷ Check if best so far
10:             bestU ← v                                 ▷ Update values
11:             bestQ ← q'
12:             bestD ← d'
13:     (y_D, y_Q) ← (y_D ⊕ bestD), (y_Q ⊕ bestQ)          ▷ Add best pair
14: return (y_D, y_Q)
```

To solve the optimization problem in Eq. (3), we shall use the fact that F is submodular and hence can be optimized using a simple and efficient *greedy* algorithm. The corresponding greedy algorithm for this problem is presented in Algorithm 1, which we refer to as the DynRR reranking method. The algorithm begins by finding $RelQ(q)$, that is, all the aspects/related queries for query q (Line 2). Iterating over each of them, it precomputes an ordering of the candidate documents by $R(\cdot|q)R(\cdot|q')$ (Line 3). This precomputation helps us avoid repeating computation during the greedy algorithm's iterations. Next the greedy algorithm computes the ranking, by iteratively finding the next best element (i.e., (d, q) pair) to add to the ranking at each step (Lines 5–13). To do so, it iterates over uncovered aspects (Line 6), finding the marginal benefit of the best uncovered document related to that aspect (Line 8): $Top(\mathbf{y}' \setminus \mathbf{y}_D)$ returns the top element in the ranking $\mathbf{y}'$ that is not covered in $\mathbf{y}_D$. Finally, the overall best solution is appended to the ranking (Line 13).

In addition to being simple, easy-to-implement, and efficient, this algorithm has the benefit of theoretical guarantees that ensure that the computed solution is comparable to the optimal, as described by the following theorem.

THEOREM 6.2. *The greedy algorithm (Algorithm 1) has an approximation factor of $\eta = \frac{e^{-\beta(1-\lambda)}}{2}$. Thus the greedy solution is at least η times as good as the optimal solution.*

The proof for Theorem 6.2 is also provided in Appendix C. In addition to theoretical guarantees, we evaluate this reranking algorithm empirically later in this section. For these empirical studies, we next discuss the evaluation measures used.

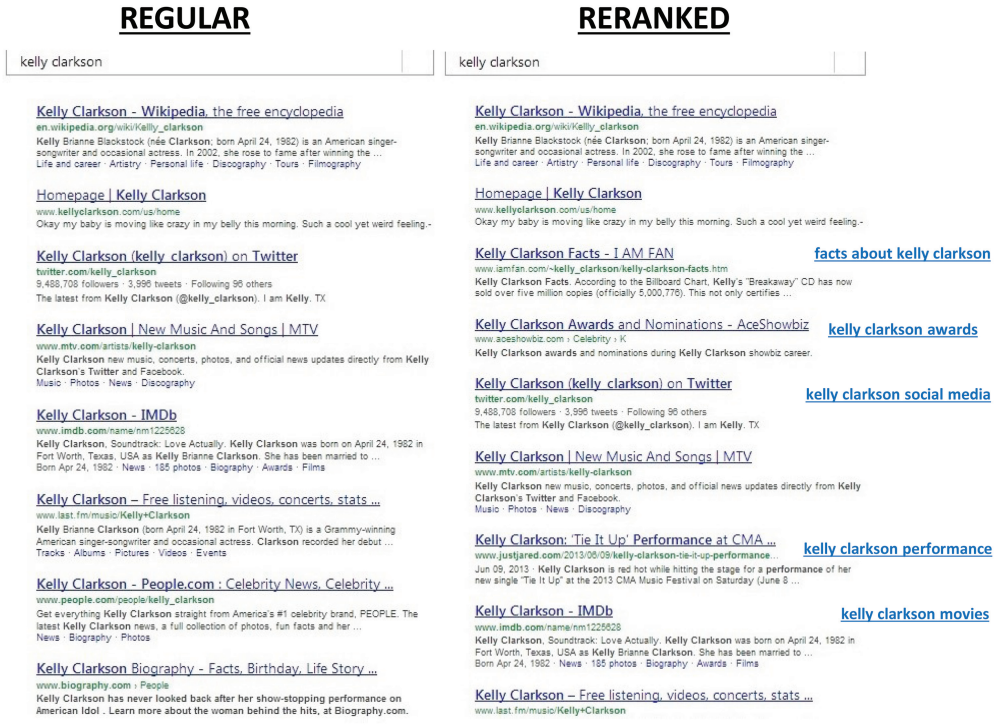

Fig. 13. Illustrative example of the reranking algorithm. The ranking on the left is a conventional ranking produced by a search engine. This ranking covers few aspects of the query. On the other hand, the ranking on the right is the reranking for intrinsic diversity. The ranking promotes documents about different (uncovered) aspects leading to a more diverse ranking. Alongside each of these documents, the ranking also presents the aspect corresponding to that result.

To better understand what this reranking seeks to achieve, we can consider the illustrative example shown in Figure 13.

6.2. Evaluation Measures

As the problem of presenting results for both the current as well as future queries is a new one, we first discuss the evaluation methodology used. In particular, we use two kinds of evaluation metrics.

Primary Ranking Metrics. To compare against standard non-interactive methods of ranking, we simply evaluate the quality of the primary ranking, that is, completely ignore the related query suggestions attributed to documents. Since our goal is *whole-session relevance*, documents are considered *relevant* if and only if they are relevant to any query in the session. Given this notion of relevance, we compute the Precision, MAP, DCG, and NDCG values. We evaluate these metrics at rank cutoffs of 1 (precision of top result); 3, which emphasizes the typical fold, that is, number of search results visible without having to scroll down; and 10, the classical cutoff used which signifies the number of search results per page.

Interactive Ranking Metrics. To evaluate the offline effectiveness and accuracy of the *predicted* future aspects (queries) and results (documents), we need to assume some model of human interaction. Consider the following search user model.

(1) Users begin at the top of the ranking.

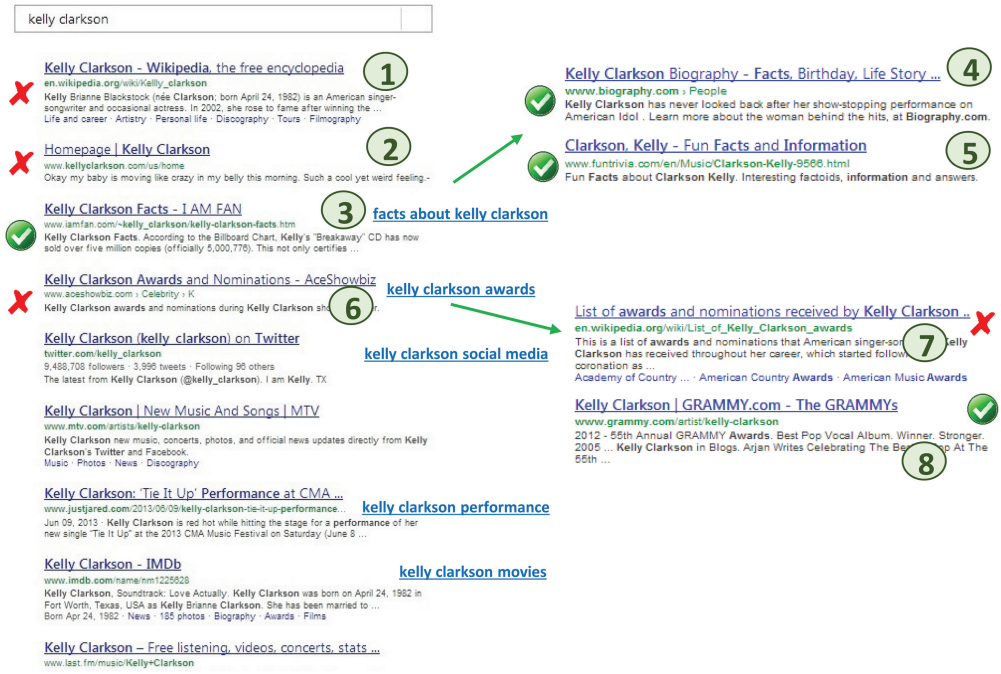

Fig. 14. Illustrative example showing how a user can interact with the results of such a dynamic ranking (with $k = 2$). This user is interested in learning about facts on Kelly Clarkson and about the Grammy Awards she has won. The user therefore clicks on the *facts* aspect as the corresponding primary result is relevant. They also click on the *awards* aspect since the second result for that aspect is relevant to that user. The figure also shows the relevance of the different documents to this user along with the order in which we model the user as perusing the results.

(2) They click the related query attributed to a document if and only if the document is relevant or the query is relevant. We say a query is *relevant* if the top-k results of the query contain a (new) relevant document.
(3) On clicking the related query, the user views the top-k results for that related query, before returning to the original document ranking, and continuing on.
(4) Users ignore previously seen documents and click on all new relevant documents.

An example of a user's interaction with the ranking under this model is shown in Figure 14, which illustrates how users skip and click different (related query) aspects. Under this 4-step user model, we can easily trace the exact ranking of documents that such a user would have navigated and thus evaluate Precision@10 and DCG@10 for this ranking. We refer to these metrics as $PrecU_k$ and $DCGU_k$, and compare them with the primary Prec@10 and DCG@10 metrics.

Note that we do not claim that this user model accurately captures all online users, nor that it is sophisticated. This is simply a well-motivated model for analyzing a rational user's actions, assuming the user is relatively accurate at predicting the relevance of an aspect based on either the top document or its related query. This, in turn, is intended to inform us about trends and relative differences we may see in online studies.

6.3. Experimental Setup

Data. To evaluate the efficacy of the method, we used the data obtained from mining the search logs, as described in Section 3. We used four main datasets as shown in

Table XVIII. Datasets Used in Reranking Experiments

Dataset	# Train	# Test
MINED5+	8,888	2,219
MINED4+	33,004	8,247
PREDID5+	4,120	1,027
PREDID4+	13,608	3,401

Table XVIII: The first two were sampled[13] from the MINED datasets involving at least four or five distinct aspects; the latter two were sampled from the corresponding PREDID datasets, which were obtained by running an SVM classifier on the initiators as described in Section 4. The training-test splits are shown in the table.

Obtaining Probability of Relevance. For our algorithm, we required the computation of the conditional relevance of a document given a query, that is, $R(d|q)$. Thus, to enable easier reproducibility by others, we learned a model using Boosted Regression Trees, on a dataset labeled with the relevance values for query-document pairs with 20,000 queries using graded relevance judgments (∼60 documents per query). The features used are given in Table XIX. All features were normalized to have zero mean and unit variance. To obtain the relevance model, we optimized for NDCG@5.

Baselines. As baselines, we used the following methods.

—*Baseline*. A state-of-the-art commercial search engine ranker, also used to compute the rank feature mentioned earlier.
—*RelDQ*. Ranking obtained by sorting as per the conditional relevance model $R(d|q)$.

We also experimented with other baselines including relevance-based methods, such as BM-25[14], cosine similarity using TF-IDF and KL-Divergence based methods, and diversity-based methods such as MMR. We used the weighted anchor text of the documents as the *textual representation* required to run these methods. However, we found that all these methods perform far worse than the baseline ranker, as well as the RelDQ method, as seen in Table XX[15], which shows the performance on a sample of 3,000 sessions of MINED4+. The underlying reason for this is that these methods rely solely on a text representation of the documents and cannot utilize additional features. The same is true for other classical diversification methods [Zhai et al. 2003; Chen and Karger 2006; Clarke et al. 2008; Swaminathan et al. 2009; Agrawal et al. 2009], making them ill-suited for this problem. While there are methods that can use richer feature representations [Raman et al. 2011; Raman and Joachims 2013], coming up with meaningful features to reflect the required intrinsic diversity is a hard problem itself. Hence we do not present any further results for these other baseline techniques.

Related Queries. To study the effect of the related queries, we used four different sources.

—*API*. We used the publicly available API of a commercial search engine. The API returns 6–10 related queries per input query.

[13]A fraction of the datasets (∼30%) was missing anchor text for a few documents (due to issues obtaining this from the server) and hence left out from this evaluation.
[14]Parameter validation was performed for BM25 with the k parameter varied from 1.2 to 2.0 and the b parameter varied from 0.6 to 0.9.
[15]Due to company policy, we unfortunately cannot provide absolute performance measures and thus report relative performance. To help readers gauge the effectiveness of the baseline ranker, we note that it's performance on the ID sessions is comparable to its average performance.

Table XIX. Features Used to Train the Relevance Model $R(d|q)$ via Boosted Trees

Query	Length
Website	Log(PageRank)
Baseline Ranker	Reciprocal Rank (if in top 10)
URL	Length # of Query Terms Covered Fraction of Query Covered TF-Cosine Similarity LM Score (KLD) Jaccard Boolean AND Match Boolean OR Match
Anchor (Weighted)	Same as URL
Anchor (Unweighted)	TF-Cosine Similarity KLD Score

Table XX. Performances of Commonly Used Ranking Techniques (Using the Weighted Anchor Text) as a Ratio with the Corresponding Performance Measure of the Baseline Ranker

Method	DCG		
	@1	@3	@10
KLD	.255	.415	.739
Cosine (TF)	.236	.425	.787
Cosine (TF-IDF)	.272	.407	.768
BM-25	.159	.267	.608
MMR	.275	.404	.735

—*Click-Graph*. Using click data from a previous time span, we built a graph of queries and the corresponding results that were clicked on. We obtained a set of 10–20 related queries from this *co-click* graph by finding queries that had the greatest overlap in the results clicked on as the issued query, while also ensuring some diversity in the queries.
—*Co-Session Graph*. Using data of queries cooccurring in the same session, we built a graph and obtained 10–20 related queries by finding queries most frequently cooccurring with the issued query, while maintaining diversity in the queries.
—*Oracle*. As an approximate upper bound, we used the actual queries issued by the user during the intrinsically diverse part of the session.

As the session data used in our experiments was collected during the period of April 1–May 31, 2012, to ensure a fair evaluation, the just-mentioned *click* and *co-session graphs* were constructed using search log data collected prior to April 2012, in the period December 2011–March 2012. For most experiments, we use either the first three related query sources described previously, or only the second and third sources, which we distinguish by the suffix C+S.

Settings. The parameters for DynRR were set by optimizing for $DCGU_3$ on the training data.[16] All numbers reported are for the test sets. We considered all SAT clicked results in the session as relevant documents. Since our comparison is relative to the baseline search engine, the assumption is that placing the SAT-clicked documents higher is better, rather than being an indication of absolute performance. Unless otherwise mentioned, the candidate document set for reranking comprises the union of (a) the top-100 results of the initiator query, and (b) the top-10 results from each related query, using the baseline ranking method.

6.4. Results

Primary Evaluation. We first study the reranking without any interactivity, using the *primary* ranking metrics to evaluate the quality of the top-level ranking. As seen in the results of Table XXI, the reranking leads to improvements across the different metrics for both datasets. Thus, even without interactivity, the method is able to outperform the baselines in predicting future results of interest to the user, while also providing

[16]We varied the λ parameter from 0 to 1 in increments of 0.1, while the β parameter was varied across the values $\{0.1, 0.3, 1, 3, 10\}$.

Table XXI. Primary Effectiveness of Different Methods, Reported as a Ratio Compared to the Corresponding Effectiveness Measure for the Baseline Ranker

Dataset	Method	Prec			MAP			DCG			NDCG		
		@1	@3	@10	@1	@3	@10	@1	@3	@10	@1	@3	@10
MINED5+	RelDQ	1.00	0.94	0.99	1.00	0.97	0.98	1.00	0.97	0.99	1.00	0.97	0.99
	DynRR	**1.06**	**1.03**	**1.04**	**1.06**	**1.05**	**1.04**	**1.06**	**1.04**	**1.04**	**1.06**	**1.05**	**1.05**
	DynRR C+S	**1.10**	**1.10**	**1.12**	**1.10**	**1.10**	**1.10**	**1.10**	**1.10**	**1.11**	**1.09**	**1.10**	**1.11**
MINED4+	RelDQ	1.00	0.97	0.98	1.00	0.98	0.98	1.00	0.98	0.98	1.00	0.98	0.99
	DynRR	**1.07**	**1.05**	**1.10**	**1.07**	**1.06**	**1.07**	**1.07**	**1.05**	**1.08**	**1.07**	**1.06**	**1.09**
PREDID5+	RelDQ	1.00	0.94	0.99	1.00	0.98	0.98	1.00	0.96	0.98	1.00	0.97	0.98
	DynRR	**1.03**	**1.02**	**1.05**	**1.03**	**1.04**	**1.03**	**1.03**	**1.03**	**1.03**	**1.03**	**1.03**	**1.05**
PREDID4+	RelDQ	1.00	0.97	0.99	1.00	0.98	0.99	1.00	0.98	0.99	1.00	0.98	0.99
	DynRR	**1.06**	**1.02**	**1.06**	**1.06**	**1.03**	**1.03**	**1.06**	**1.04**	**1.04**	**1.05**	**1.03**	**1.05**

Table XXII. Interactive Performance of DynRR for Different User Models (as Ratios Compared to the Baseline Prec@10 and DCG@10)

Dataset	Method	PREC@10				DCG@10			
		$PrecU_1$	$PrecU_2$	$PrecU_3$	$PrecU_5$	$DCGU_1$	$DCGU_2$	$DCGU_3$	$DCGU_5$
MINED5+	DynRR	1.093	1.247	1.347	1.401	1.075	1.188	1.242	1.254
	DynRR C+S	1.166	1.310	1.413	1.464	1.146	1.251	1.306	1.313
MINED4+	DynRR	1.152	1.292	1.380	1.438	1.114	1.212	1.258	1.277
PREDID5+	DynRR	1.103	1.223	1.295	1.345	1.074	1.153	1.190	1.204
PREDID4+	DynRR	1.097	1.207	1.271	1.311	1.075	1.147	1.182	1.191

results for the current query. In particular, we found the DynRR method works best using the C+S related queries (which we return to later) with 9–11% gains over the baselines at position 10 across the various metrics, and 3–5% in relative gains. We also find that the method improves on the PREDID datasets, suggesting that the method can be robustly used in practical scenarios. These performance differences were also found to be statistically significant: across all four datasets, a binomial test shows that the difference between the DCG@10 performance of the DynRR and the baseline is statistically significant at the 99.99% significance level. Thus, we improve an important segment of tasks while maintaining high levels of performance elsewhere. Further improvements to the initiator classification model are likely to result in additional robustness gains.

Interactive Evaluation. Next we evaluate the performance of the method when incorporating user interactivity. As seen in Table XXII, accounting for interactivity leads to large increases in both the precision and DCG of the user paths navigated across the different user models and datasets. In fact, we find 30–40% improvements in precision and 20–25% improvements in DCG, indicating that our approach is able to do a far better job in predicting future relevant results (and potentially, queries). These results also show that the method improvements are relatively robust to the user model. We also confirmed that the DynRR improvement (for $DCGU_3$) is statistically significant compared to the baseline at the 99.99% significance level, using a binomial test.

Robustness. A key concern when comparing a new method against a baseline is the robustness of the method. In particular, we are interested in the number of queries that are either improved or hurt on switching from the baseline method to the proposed reranking method. This is particularly crucial for the PREDID datasets, since we would not want retrieval effectiveness on non-ID sessions to be adversely affected. Table XXIII displays the % of examples for which the method either gains or loses above a certain threshold, compared to the baseline. We see that the percentage of queries with a performance gain exceeds those with a performance loss, especially while interactivity

Table XXIII. Distribution of (Absolute) Performance Difference between DynRR and the Baseline DCG@10 across Individual Sessions

Dataset	Δ = Difference in evaluation metrics	% Sessions Improved			% Sessions Worsened		
		$\geq$0.2	$\geq$0.5	$\geq$1.0	≤ -0.2	≤ -0.5	≤ -1.0
MINED5+	DynRR $DCGU_3$ - Baseline DCG@10	34.4	13.0	1.6	9.9	2.7	0.1
	DynRR DCG@10 - Baseline DCG@10	19.6	5.2	0.3	12.7	3.8	0.3
MINED4+	DynRR $DCGU_3$ - Baseline DCG@10	31.8	13.1	2.1	9.5	2.7	0.2
	DynRR DCG@10 - Baseline DCG@10	17.7	5.6	0.8	13.0	4.0	0.2
PREDID5+	DynRR $DCGU_3$ - Baseline DCG@10	29.1	12.0	1.6	10.8	3.7	0.2
	DynRR DCG@10 - Baseline DCG@10	17.7	6.0	0.8	12.9	4.0	0.2
PREDID4+	DynRR $DCGU_3$ - Baseline DCG@10	27.5	11.0	2.0	10.5	3.1	0.3
	DynRR DCG@10 - Baseline DCG@10	16.8	5.3	1.4	11.5	3.5	0.4

Table XXIV. Examples of Sessions with Significant Increase in the Primary DCG@10 of Reranked Results Compared to that of the Baseline

Initiator Query	ID Successor Queries
`what does a positive r wave in avr look like`	`avr on ekg; r wave in avr with tricyclic od; terminal r wave in avr; what is 3mm on ekg`
`is a high level of dhea a sign of cancer`	`what can be done to lower dhea levels in women; affects of elevated dhea; high dhea numbers`
`accomplishments kirk franklin`	`kirk franklin gospel songs; kirk franklin gets married; when did kirk franklin make his first cd; where did kirk franklin go to college at`

Table XXV. Performance Change on Varying the Related Queries for the MINED5+ Dataset

RelQ	Prec	DCG	$PrecU_3$	$DCGU_3$
A	0.927	0.880	1.082	0.997
C	1.039	1.014	1.333	1.214
S	1.076	1.074	1.248	1.198
O	1.511	1.397	2.211	1.827
AS	0.984	0.961	1.271	1.157
AC	1.010	1.013	1.244	1.176
CS	1.115	1.106	1.413	1.306
ASC	1.019	1.039	1.347	1.242
ASCO	1.207	1.144	1.580	1.386

Note: All measures are @10 and reported as a ratio to the baseline values.
(A = API; C = Co-click; S = Co-Session; O = Oracle).

is incorporated in the comparison. Table XXIV contains examples of sessions (initiator and successors) where the method shows improvements over the baseline.

Effect of Related Query Set. Next, we study how varying the nature of the related queries affects retrieval performance, using the MINED dataset. To do this, we constructed different combinations of the four related query sources: API (A), Click-Graph (C), Co-Session (S), and Oracle (O).

The results are summarized in Table XXV. As we clearly see, the choice of related query source has a large impact on both the primary ranking performance and the interactive performance. In particular, one result that stands out is the extremely strong performance using the Oracle-related queries. This suggests that if we were able to improve the quality of the suggested related queries, it would only increase our algorithm's effectiveness. On the other hand, we see that using the API-related queries almost always hurts retrieval effectiveness. In fact, simply using only the related queries from the click-graph and the co-session data leads to much better

Table XXVI. Effect of the Candidate Document Set Quality on DynRR Performance (for the MINED5+ Dataset)

Candidate Doc Set $\mathcal{D}$	DynRR DCG			DynRR $DCGU_j$			% $d_i \in Top-k(q_i)$		
	@1	@3	@10	$j=2$	$j=3$	$j=5$	$k=1$	$k=3$	$k=10$
Top-10 (Baseline) of query q	**1.080**	**1.088**	1.039	**1.258**	**1.300**	**1.290**	23.3	36.3	51.0
Top-10 (Base) of $q \cup$ Top-10 (Base) of all $q' \in RelQ(q)$	1.055	1.041	1.039	1.188	1.242	1.254	**78.9**	**89.7**	**93.7**
Top-100 (Base) of q	**1.080**	**1.084**	**1.054**	**1.254**	**1.292**	**1.287**	34.7	50.8	63.9
Top-100 (Base) of $q \cup$ Top-10 (Base) of $q' \in RelQ(q)$ (*Default*)	1.055	1.041	1.039	1.188	1.242	1.254	**78.9**	**89.7**	**93.7**

Note: All measures are reported as a ratio to the baseline values.

Table XXVII. Effect of the Candidate Document Set Quality on DynRR Performance for the Other Datasets

Dataset	Candidate Doc Set $\mathcal{D}$	DynRR DCG			DynRR $DCGU_j$			% $d_i \in Top-k(q_i)$		
		@1	@3	@10	$j=2$	$j=3$	$j=5$	$k=1$	$k=3$	$k=10$
MINED4+	Top100(q)$\cup$ Top10 $q' \in RelQ(q)$	1.066	1.054	**1.075**	1.212	1.258	1.277	**75.4**	**87.0**	**92.7**
	Top-100 of q	**1.086**	**1.089**	1.071	**1.261**	**1.294**	**1.294**	35.2	51.4	64.3
PREDID5+	Top100(q)$\cup$ Top10 $q' \in RelQ(q)$	1.027	1.029	**1.033**	1.153	1.190	1.204	**74.0**	**85.5**	**90.8**
	Top-100 of q	**1.055**	**1.030**	1.019	**1.196**	**1.225**	**1.226**	31.8	45.9	58.4
PREDID4+	Top100(q)$\cup$ Top10 $q' \in RelQ(q)$	1.057	1.035	1.041	1.147	1.182	1.191	**74.0**	**85.3**	**90.6**
	Top-100 of q	**1.084**	**1.076**	**1.050**	**1.197**	**1.218**	**1.216**	34.9	50.6	63.2

Note: All measures are reported as a ratio to the baseline values.

performance compared to using the API queries as well. Further analysis reveals that this is due to two reasons: (a) in many cases, the queries returned by the API are spelling corrections or reformulations, with no difference in aspect; (b) more importantly, there is little to no diversity in the queries obtained from the API as opposed to those from the other sources.

Effect of Candidate Document Quality. As the proposed approach is a reranking approach, performance is affected by the quality of the original candidate documents. Ideally, we would like the method to work well both when the starting candidate document set is low quality (containing many irrelevant documents but easy to obtain) or when the candidate set is of high quality (but requires running a computationally heavier ranking/filtering algorithm to obtain). Table XXVI shows the change in some of the performance measures as we change the quality of the candidate set. We find that starting with a less-noisy candidate set, by restricting to the top 100 without documents from other queries, tends to improve performance. Encouragingly, our approach does well even if the candidate set contains a large fraction of irrelevant documents, as our default experimental setting does. This robustness is also observed on the other datasets, as shown in Table XXVII.

However, the trade-off of adding more diverse documents into the candidate set, as seen in Tables XXVI and XXVII, is that the documents of the ranking are less relevant to the query aspect to which they are attributed. The last three columns of both tables indicate how common it is for the document d_i of the ranking to be in the top-k for the corresponding query aspect q_i, for different k. We find that when documents from the related query aspects are included, a large fraction of the time the document attributed to the query aspect turns out to be the most relevant document. This comes at the cost of a slight reduction in the ranking effectiveness of the primary ranking.

6.5. TREC Session Data

We also ran experiments using the publicly available TREC 2011 Session data using only publicly reproducible components. To do so, three annotators labeled the different sessions as potentially being intrinsically diverse or not, based on (a) only the queries

Table XXVIII. Annotator Agreement on TREC Data

Task	Fleiss Kappa	% All agree	% 2 agree
IsTopicID?	.423	85.5	100
AreQueriesID?	.452	67.1	100
BestInitiatorQ	.694	55.3	98.7

Table XXIX. Absolute Performance on TREC Session Data

Initiator	Method	Pr@1	Pr@3	DCG@1	DCG@3
Title	Baseline	0.58	0.60	0.84	2.13
Title	DynRR	$0.71^{\dagger}$	0.60	$1.39^{\dagger}$	2.41
First	Baseline	0.53	0.47	0.94	1.94
First	DynRR	0.5	0.48	0.92	1.97
Label	Baseline	0.55	0.51	0.87	1.95
Label	DynRR	0.61	0.5	1.13	2.09

Note: †indicates significance at $p = 0.05$ by a paired one-tailed t-test.

issued; and (b) the narration and title of the session as well. We also asked annotators to label their opinion on the query best suited to be the initiator query, among the queries issued. Annotators were provided the description of ID sessions as described at the start of Section 3 and provided with the information sheet given in Appendix D.2.

We found good agreement among the different annotators for all of the different labeling tasks, as seen from Table XXVIII. In fact, in 63 of the 76 total sessions, all three annotators agreed the sessions were ID based on the narration, title, and queries.[17]

For training, we used a 50-50 training-test split on all sets, with the training data used for selecting the parameters of the ranking methods. To obtain the conditional relevance $R(d|q)$, we trained a regularized linear regression model with features based on the scores of two standard ranking algorithms: BM-25, and TFIDF. As labeled data, we used the TREC Web data from 2010 and 2011 by converting the graded relevance scores for relevant and above from the $\{1, 2, 3\}$ scale to $\{\frac{1}{3}, 1, 1\}$. We used related queries from the Van Dang-Croft [Dang et al. 2010] method (Q) on the ClueWeb '09 anchor text, where the starting seed for the random walk would use the most similar anchor text to the query by *tf.idf*-weighted cosine if an exact match was not available. Our candidate document pool was set similar to the previous experiments.

To evaluate, we again use the same metrics as before, but using the TREC assessor relevance labels instead of clicks. We considered three different candidates for the initiator query: (a) topic, (b) first query in the session, and (c) labeled initiator query. As a baseline, we considered the method that ranked as per $R(d|q)$. For the DynRR method, we used the titles of the top-10 results of a query (as per the baseline), as the *snippet* of the query.

The results for the primary metric comparison are shown in Table XXIX. As we see from the table, the method improves in precision and DCG for most of the cases with particularly large improvements when the title of the *topic* is used as the initiator query. This matches feedback the annotators gave us, that the titles looked much more like the general queries issued by Web users. In contrast, the TREC sessions would often start with a specific query before moving to a more general query. It could be that supplying the user with a well-formulated topic description before starting the search task influences the users to search for a particular aspect rather than issue a more general query, as they might when no topic description is explicitly formulated.

[17]Using a slightly different classification scheme, Liu et al. [2011] also found 66 of the 76 sessions to have the same type.

7. WHOLE SESSION RELEVANCE, PROACTIVE SEARCH, AND FUTURE WORK

Our work is a first step toward *whole-page relevance*, as motivated in Bailey et al. [2010], and eventually the goal of *whole-session relevance*. Just as whole-page relevance considers how the entire set of elements on a result page can work together to address a user's information need, whole-session relevance aims to optimize an effectiveness objective based on the entire series of past, present, and future queries and result pages shown to a user over time. Such whole-session objectives can capture important longer-term qualities of search needs beyond a single query, such as time-evolving information needs, task- and subtask-completion, and this article's focus on intrinsic diversity in exploring a topic. As this is just the first step into this problem, this also opens many interesting future directions, a few of which we now summarize.

In the initial stages of our pipeline, we could consider iterative ways to combine or jointly optimize the mining and query identification processes, so that information gained in one stage could be used to improve the accuracy of the other. Also, various aspects of the filtering algorithm in Section 3 could be implemented with more general mechanisms. For example, Step 6 relies on identifying pairs of related queries using co-surfaced URLs as evidence, but this could be replaced with improved methods for identifying related queries that could improve identification of tasks and proactive retrieval performance. In other work, identifying more "general" definitions of initiators would help improve robustness and applicability: in this article, we weighted heavily toward temporal precedence as a key feature of a broader initiator query, but some users may start with specific queries (e.g., in order to reduce ambiguity) and then generalize once the specific area of interest has been accurately identified. In the search interface itself, we envision richer display elements or modes of interaction for supporting result exploration for intrinsically diverse tasks. Extending these techniques to related problems like exploratory search is another fruitful direction for future research.

Looking beyond single sessions, identifying intrinsically diverse tasks that bridge session boundaries (i.e., cross-session intrinsically diverse tasks) is a natural extension. Moreover, the ability to detect cross-session tasks could be combined with our ability to predict future queries. This would provide a form of proactive search that uses the time between sessions to pre-fetch result elements likely to be of use when the task is contined in a future session. In these more extended search time scales, human computation could play a significant role in our pipeline, either for query mining and prediction, or to provide entirely new capabilities for interpreting complex or difficult intrinsically diverse queries or optimizing whole-page results. Even for the specific subtask of selecting likely future queries, it would be interesting to see how using more time, via human computation or other means, could help close the existing gap against oracle performance that we identified in Section 6.3.

8. CONCLUSIONS

Our work is the first to characterize the nature of *intrinsically diverse tasks* in information retrieval and to develop algorithms that support such tasks. Intrinsically diverse tasks are those that typically require multiple user queries to a search engine to cover different aspects of the same information need. First, we motivated our work using real-world data and presented an algorithm to mine intrinsically diverse sessions from search logs, using behavioral interaction signals within a session. We then examined the question of predicting when a query has initiated the start of an intrinsically diverse task, by casting the problem in terms of binary classification. We conducted an analysis of the resulting queries, sessions, and classification results. We also looked at the more general problem of predicting which queries were part of an ID task engagement within a session, and examined the role of session context in

prediction effectiveness. Finally, we presented a new class of algorithm designed to optimize retrieval for intrinsically diverse tasks. Our approach alters the search result rankings presented to the user so as to provide information relevant to aspects of the ID task for which the user is likely to search in the future. We validated our approach empirically using search log data, as well as TREC data, demonstrating significant improvement over competitive baselines in both cases.

APPENDIXES

A. MINING ALGORITHM

Appendix A contains the detailed algorithm used for mining ID sessions from query logs (Alg. 2).

ALGORITHM 2: Obtaining Intrinsic Diversity Data

function RemoveCommonAndLong(Session $s = \{q_1, q_2, \ldots\}$, QueryLength l) ▷ Removes Common Queries as well as Long Queries from Query Session s
  $s' = \{\}$
  for all $q_i \in s$ **do**
    if `IsCommon`$(q_i) =$ `false` and $len(q_i) \leq l$ **then** ▷ Discards common/long query
      $s' = s' \cup \{q_i\}$
  return s'

function RemoveDups(Session $s = \{q_1, q_2, ...\}$) ▷ Removes Repeated Query Instances *i.e.,* Duplicate Queries
  $s' = \{\}$
  for all $q_i \in s$ **do**
    if $q_i \in s'$ **then** ▷ Discards the query if it is a common query
      Merge SAT Clicked Results in $s'[q_i]$
    else
      $s' = s' \cup \{q_i\}$
  return s'

function GetNumDistinct(Session $s = \{q_1, q_2, ...q_n\}$, Threshold η) ▷ Counts number of distinct queries
  $s' = \{\}$
  for $i = 2 \rightarrow n$ **do**
    $flag \leftarrow$ `true`
    for all $q \in s'$ **do**
      if $Sim(q, q_i) \geq \eta$ **then** ▷ Don't add if similar to previous query
        $flag \leftarrow$ `false`
        break
    if $flag =$ `true` **then**
      $s' = s' \cup \{q_i\}$
  return $len(s')$

function GetIDSession(Session $s = \{q_1, q_2, ...\}$, Similarity Threshold η) ▷ Gets all related ID queries with q_1 as Initiator.
  $s' = \{q_1\}$
  $hasSat \leftarrow hasSatResult(q_1)$ ▷ Set to true if ∃ SAT click for q_i
  for $i = 2 \rightarrow n$ **do**
    if $Sim(q_1, q_i) \leq \eta$ and $Top10Results(q_1) \cap Top10Results(q_i) \neq \phi$ **then** ▷ Syntactically not too similar to initiator but at least 1 common result in Top 10
      $s' = s' \cup \{q_i\}$
      $hasSat \leftarrow hasSat \vee hasSatResult(q_i)$

if $hasSat =$ `true` **then**
  return s'
else
  return ϕ

$numDistinct \leftarrow new\ Dict()$
for all QuerySessions $\{q_1, q_2...\}$ **do**
  $\{q'_1, q'_2...\} \leftarrow$ `RemoveCommon`$(\{q_1, q_2...\}, ql)$ ▷ ql is parameter for max query length.
  $\{q''_1, q''_2..., q''_n\} \leftarrow$ `RemoveDups`$(\{q'_1, q'_2...\})$
  $best \leftarrow \phi$ ▷ At most 1 ID Session per actual session and thus choose longest
  $bestVal \leftarrow -1$
  for $i = 1 \rightarrow n + 1 - l$ **do** ▷ l is parameter for Minimum ID Session Length
    $\{q^*_1, q^*_2..., q^*_m\} \leftarrow$ `GetIDSession`$(\{q''_i, q''_{i+1}..., q''_n\}, \eta_1)$ ▷ η_1 and η_2 are similarity threshold parameters
    if $m \geq l$ and $bestVal < m$ **then** ▷ Has to meet minimum length condition
      $best \leftarrow \{q^*_1, q^*_2..., q^*_m\}$
      $bestVal \leftarrow m$
  if $bestVal > 1$ **then**
    $numDistinct[best] =$ `GetNumDistinct`$(best, \eta_2)$ ▷ Add the best seen
Sort $numDistinct$ in descending order of value and choose top k.

B. STATISTICAL PROPERTIES OF ID INITIATORS

Table XXX provides a breakdown of aggregate statistical properties of the two kinds of queries: ID initiators and regular queries. While their lexical length is roughly the same, we find that ID initiators tend to appear more frequently and in longer sessions (in the Dec. 2011–Mar. 2012 period) that last up to 50% longer on average (Figure 15(c)). For the same date range, if we look at the average similarity of a query to all the queries it co-occurred with, we find that regular queries tend to have slightly higher similarity on average. Further analysis reveals that regular queries are more likely to have very low (e.g., for off-topic queries) or very high (e.g., for reformulations) similarities (Figure 15(b)). Note that since this analysis uses data from a non-intersecting date range, compared to that used for mining the data, we can conclude that these query characteristics are intrinsic to ID initiators and not a function of our mining algorithm.

Table XXX. Mean and Standard Deviation of Different Query Characteristics

	REGULAR		ID	
Query Characteristic	Mean	Dev.	Mean	Dev.
Total number of characters	23.01	13.82	22.11	10.40
Total number of words	3.73	2.50	3.70	1.85
Log(Number of sessions) previously occurred in	2.30	1.29	3.08	1.75
Avg. length of sessions previously occurred in	10.02	15.21	15.22	19.26
Avg. similarity with all co-session queries (from logs)	0.188	0.167	0.152	0.193
Fraction of co-session similarities $\in [0, 0.25)$	69.83	27.20	67.15	25.69
Fraction of co-session similarities $\in [0.25, 0.5)$	12.29	17.13	16.77	17.46
Fraction of co-session similarities $\in [0.5, 0.75)$	11.53	17.24	11.69	15.47
Fraction of co-session similarities $\in [0.75, 1]$	6.35	14.09	4.39	10.52

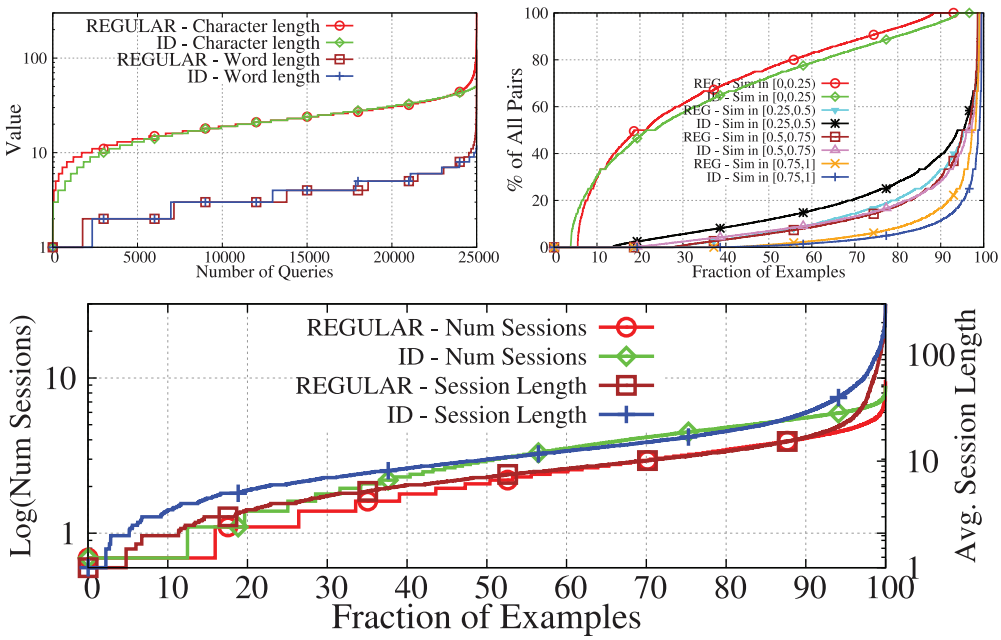

Fig. 15. Differences between regular and ID initiator queries for (a) lexical statistics; (b) for each query, the fraction of co-session similarity pairs that fall within different ranges; (c) session frequency and length statistics of previous occurrences in logs.

C. SUBMODULARITY OF OBJECTIVE & APPROXIMATION GUARANTEE

C.1. Proof of Theorem 6.1

PROOF. To show that the objective function $F_{\beta,\lambda}$ from Eq. (2) is submodular, we will construct a ground set of elements, define a set-based function on this set matching the objective, and show the marginal benefit to be decreasing. Recollect the following objective:

$$F_{\beta,\lambda}(d_1, q_1, \ldots, d_n, q_n) = \sum_{i=1}^{n} R(d_i|q) \cdot R(d_i|q_i) \cdot e^{\beta Div_\lambda(q_i, \mathcal{Q}^{(i-1)})}.$$

Given a set of related query aspects $\mathcal{Q}$, a set of candidate documents $\mathcal{D}$, and a ranking length n, we define $\mathcal{B}$—the ground set of elements—to be $\mathcal{B} = \mathcal{D} \times \mathcal{Q} \times \mathcal{K}$, that is, the set of all possible triples of *document* d, *query aspect* q', and *ranking position* k:

$$\mathcal{B} = \{(d, q', k) : d \in \mathcal{D} \wedge q' \in \mathcal{Q} \wedge k \in [1, 2, \ldots, 2n]\},$$

where $\mathcal{K} = [1, 2, \ldots, 2n]$.

Let $g_1(q_1)$ denote $Sim(q_1, Snip(q))$ and $g_2(q_1, q_2) = Sim(Snip(q_1), Snip(q_2))$. Given a subset of the ground set $S \subseteq \mathcal{B}$, we can write the objective function as

$$G(S) = \sum_{(d,q',k) \in S} R(d|q) \cdot R(d|q') \cdot exp\Big(\beta\lambda \cdot g_1(q') - \beta(1-\lambda) \cdot \max_{(d',q_i,k') \in S \wedge k' < k} g_2(q', q_i)\Big).$$

Given a set $S = \{(d_1, q_1, 1), \ldots, (d_i, q_i, i), \ldots, (d_n, q_n, n)$, we find that $G(S) = F_{\beta,\lambda}(d_1, q_1, \ldots, d_n, q_n)$. Thus, in other words, $G(S)$ generalizes the objective function defined over a subset of the ground set $\mathcal{B}$. It is not hard to see that $G(S)$ is a valid set function, as it is agnostic to the order of elements within the set S.

We now try to prove a submodularity-like property of this set function $G(S)$. To do so, let us define the *maxpos* function for a set S as $mp(S) = \max_{(d,q',k)\in S} k$. We can now state the following monotonicity lemma.

LEMMA C.1. *The function G is not monotone in general, that is, for $S \subseteq \mathcal{B}$: $G(S \cup (d, q', k)) \not\geq G(S)$. However this function is monotone in a specific order (order-monotone):*

$$\forall S \subseteq \mathcal{B}, (d, q', k) \in \mathcal{B} : G(S \cup (d, q', k)) \geq G(S) \text{ if } k > mp(S).$$

Thus in other words, the function is increasing when an element at a later position is added to the set. We refer to this as *order-monotone*.

PROOF. Let us consider the marginal benefit $\Delta_{G(S)}(\{(d, q', k)\})$ of adding $\{(d, q', k)\}$ to S, where $k > mp(S)$:

$$\begin{aligned}\Delta_{G(S)}(\{(d, q', k)\}) &= G(S \cup \{(d, q', k)\}) - G(S) \\ &= R(d|q) \cdot R(d|q') \cdot exp\Big(\beta\lambda g_1(q') - \beta(1-\lambda) \max_{(d',q_i,k')\in S} g_2(q', q_i)\Big). \qquad (4)\end{aligned}$$

The expression in Eq. (4) is a product of three positive terms, and hence is ≥ 0, thus completing the proof. □

We now state our main lemma about the submodularity of G.

LEMMA C.2. *The function G is not submodular in general, that is, for sets $S_1 \subseteq S_2 \subseteq \mathcal{B}$: $\Delta_{G(S_1)}(\{(d, q', k)\}) \not\geq \Delta_{G(S_2)}(\{(d, q', k)\})$. However, this function is submodular in a specific order (order-submodular):*

$$\forall S_1 \subseteq S_2 \subseteq \mathcal{B}, (d, q', k) \in \mathcal{B} : \Delta_{G(S_1)}(\{(d, q', k)\}) \geq \Delta_{G(S_2)}(\{(d, q', k)\}) \text{ if } k > mp(S_2).$$

PROOF. To prove this let us revisit the marginal benefit expressions:

$$\Delta_{G(S_1)}(\{(d, q', k)\}) = R(d|q) \cdot R(d|q') \cdot exp\Big(\beta\lambda g_1(q') - \beta(1-\lambda) \max_{(d',q_i,k')\in S_1} g_2(q', q_i)\Big),$$

$$\Delta_{G(S_2)}(\{(d, q', k)\}) = R(d|q) \cdot R(d|q') \cdot exp\Big(\beta\lambda g_1(q') - \beta(1-\lambda) \max_{(d',q_i,k')\in S_2} g_2(q', q_i)\Big).$$

Since both of these terms are positive (from Lemma C.1), consider the ratio of the two marginal benefits:

$$\frac{\Delta_{G(S_1)}(\{(d, q', k)\})}{\Delta_{G(S_2)}(\{(d, q', k)\})} = exp\left(\beta(1-\lambda)\left[\max_{(d',q_i,k')\in S_2} g_2(q', q_i) - \max_{(d',q_i,k')\in S_1} g_2(q', q_i)\right]\right). \qquad (5)$$

However, since $S_1 \subseteq S_2$, we have that $\max_{(d',q_i,k')\in S_2} g_2(q', q_i) \geq \max_{(d',q_i,k')\in S_1} g_2(q', q_i)$. Hence the RHS of Eq. (5) is at least 1, thus completing the proof. □

Since we add elements to the ranking only in the order of their positions, we thus get that our objective function F is *submodular*. □

C.2. Proof of Theorem 6.2

PROOF. To prove an approximation guarantee for the greedy algorithm on the objective function Eq. (3), we will instead consider a constrained optimization problem involving the generalized set function G. In particular, while trying to maximize $G(S)$, we are dealing with three *matroid* constraints here.

(1) *Documents Cannot Be Repeated*. This constraint ensures that we do not present the same document twice in the ranking. This can be represented as a matroid constraint $\mathcal{I}_\mathcal{D}$ (set of all independent sets), where $\forall S \in \mathcal{I}_\mathcal{D}$: $S \cup \{(d, q', k)\} \in \mathcal{I}_\mathcal{D}$ if and only if $\nexists (d, q_1, k') \in S$.

(2) *Queries Cannot Be Repeated*. This constraint ensures that we do not repeat the same related query aspect in the interactive ranking. As a matroid constraint $\mathcal{I}_\mathcal{Q}$, this requires $\forall S \in \mathcal{I}_\mathcal{Q}$: $S \cup \{(d, q', k)\} \in \mathcal{I}_\mathcal{Q}$ if and only if $\nexists (d', q', k') \in S$.
(3) *Ranking Positions Cannot Be Repeated*. This constraint ensures that there is at most one document-related query pair at each position of the ranking. As a matroid constraint $\mathcal{I}_\mathcal{K}$, this requires $\forall S \in \mathcal{I}_\mathcal{K}$: $S \cup \{(d, q', k)\} \in \mathcal{I}_\mathcal{K}$ if and only if $\nexists (d', q_1, k) \in S$.

Thus the optimization problem reduces to

$$\text{argmax}_{S: S \in \mathcal{I}_\mathcal{D} \wedge S \in \mathcal{I}_\mathcal{Q} \wedge S \in \mathcal{I}_\mathcal{K}} G(S).$$

It is a well-known result that the greedy algorithm has an approximation guarantee of $\frac{1}{p+1}$ for submodular maximization under the intersection of p matroids [Fisher et al. 1978]. However, we do not use this result as it requires G to be submodular (which it is not). Instead, we prove a stronger result by exploiting the structure of our problem.

Let T_m $(1 \leq m \leq n)$ be the solution of the greedy algorithm after m steps. Let (d_m, q_m) represent the document-query pair selected by the greedy algorithm (and we can safely assume that this was at position m). Let δ_m represent the marginal benefit of this element: $\delta_m = G_{T_{m-1}}(\{(d_m, q_m, m)\})$. By order-submodularity, we have $\forall m : \delta_m \leq \delta_{m-1}$.

Consider the optimal solution for a ranking of length n : O. Let (d^*_m, q^*_m) represent the mth (ordered) document-query pair in O. Without loss of generality, we can assume that $O = \{(d^*_1, q^*_1, n+1), (d^*_2, q^*_2, n+2), \ldots, (d^*_i, q^*_i, n+i), \ldots, (d^*_{2n}, q^*_n, 2n)\}$, since the function value of G does not depend on the specific position value of the highest-position element. Let e_i represent $(d^*_i, q^*_i, n+i)$

It is not hard to show that O can be partitioned into n subsets (each of size at most 2) : $O_1, \ldots, O_i, \ldots, O_n$, where the elements in O_i are valid document-query pairs such that $T_{i-1} \cup O_i \in (\mathcal{I}_\mathcal{D} \cap \mathcal{I}_\mathcal{Q} \cap \mathcal{I}_\mathcal{K})$. The existence of such a partitioning can be shown using an argument similar to that used in Appendix B of Călinescu et al. [2011].

Given such a partition, we can show that

$$|O_i|\delta_i \geq \sum_{e \in O_i} G_{T_{i-1}}(e) \geq \sum_{e \in O_i} G_{T_n}(e), \tag{6}$$

where the first inequality utilizes the fact that the greedy algorithm always chooses the element maximizing the marginal benefit. The second inequality utilizes the order-submodularity of G. We now obtain

$$G(T_n) = \sum_{i=1}^{n} \delta_i \geq \frac{1}{2} \sum_{i=1}^{n} |O_i|\delta_i \geq \frac{1}{2} \sum_{i=1}^{n} \sum_{e \in O_i} G_{T_n}(e) = \frac{1}{2} \sum_{e \in O} G_{T_n}(e), \tag{7}$$

where the first equality uses the definition of G and δ, the next inequality the fact that $\forall i : 0 \leq |O_i| \leq 2$, the next using Eq. (6), and the last by realizing that $O_1 \cup O_2 \cdots \cup O_n = O$.

We can now use order-submodularity as follows:

$$\sum_{e \in O} G_{T_n}(e) = \sum_{i=1}^{n} G_{T_n}(e_i) \geq \sum_{i=1}^{n} G_{T_n \cup \{e_1, e_2, \ldots, e_{i-1}\}}(e_i) = G(T_n \cup O) - G(T_n), \tag{8}$$

where the first inequality uses the order-submodularity property and the last equality by realizing that we have a telescoping sum of marginal benefits.

Using the definition of G and the fact that $0 \leq g_2(\cdot, \cdot) \leq 1$, we have that

$$G(T_n \cup O) - G(T_n) \geq e^{-\beta(1-\lambda)} G(O). \tag{9}$$

Combining Equations (7), (8), and (9), we obtain the required bound:

$$G(T_n) \geq \frac{e^{-\beta(1-\lambda)}}{2} G(O). \quad \square$$

D. INSTRUCTIONS FOR LABELING INTRINSICALLY DIVERSE SESSIONS

This appendix contains the information sheet provided to the annotators for labeling sessions as Intrinsically Diverse or Regular as used in the analysis of the filtering process (Section 3.1) and the TREC data (Section 6.5). This sheet contains information about what constitutes an ID session and an initiator query. It also contains instructions for the labeling process. We provide here the different instruction sets used for the different tasks. A key difference in the instructions is the absence of an initiator query identification task in the annotation of the filtered sessions.

D.1. Guidelines for Filtered Session Labeling

Goal of the Study: Intrinsic Diversity

The goal behind this study is to identify **intrinsic diversity**. *Intrinsic diversity* in queries is when there are multiple queries about different aspects of the same information need.

For example, suppose I wanted to learn about *"Why Kelly Clarkson is popular?"*, I can issue queries about her, her participation at American Idol, her performances, the awards she has won and so on.. Such a set of queries would be considered intrinsically diverse.

Another example of intrinsic diversity is suppose you wanted to learn about *"snow leopards (the animals)"*. Here you could issue queries like: *"what habitats are snow leopards found in", "what do snow leopards eat", "how long do snow leopards live"*

To determine if a set of queries (or a topic) is intrinsically diverse, follow this general rule of thumb: *If the required information can be obtained more efficiently by issuing multiple queries (about different aspects of the same topic) instead of any single query, then the set of queries/topic is considered intrinsically diverse.*

Note: If there is more than one aspect/requires more than one query then it is considered intrinsically diverse.

Labeling Guidelines

You will be given a set of queries from a single user session (in the order they were queried).

Based on this information you will be asked to label each of these sessions with the following information:

Are the queries intrinsically diverse?: Taking into account all the queries listed for the session, please label if you believe the set of queries is intrinsically diverse or not **(Yes=1/No=0)**. You may place the label next to the first query listed for the session only one label is necessary per session (demarcated by a row with "———-"). **NOTE:** Even if there are multiple queries, the set of queries need not be intrinsically diverse. For example, when the queries are simply spelling corrections or reformulations that do not provide evidence of multiple aspects, it is conceivable that a single well-formed query would have retrieved information to satisfy the users need.

Thank you for your participation in this study!

D.2. Guidelines for TREC Labeling

Goal of the Study: Intrinsic Diversity

The goal behind this study is to identify **intrinsic diversity**. *Intrinsic diversity* in queries is when there are multiple queries about different aspects of the same information need.

For example, suppose I wanted to learn about *"Why Kelly Clarkson is popular?"*, I can issue queries about her, her participation at American Idol, her performances, the awards she has won and so on.. Such a set of queries would be considered intrinsically diverse.

Another example of intrinsic diversity is suppose you wanted to learn about *"snow leopards (the animals)"*. Here you could issue queries like: *"what habitats are snow leopards found in", "what do snow leopards eat", "how long do snow leopards live"*

To determine if a set of queries (or a topic) is intrinsically diverse, follow this general rule of thumb: *If the required information can be obtained more efficiently by issuing multiple queries (about different aspects of the same topic) instead of any single query, then the set of queries/topic is considered intrinsically diverse.*

Note: If there is more than one aspect/requires more than one query then it is considered intrinsically diverse.

Initiator Queries

A secondary goal of this study is to identify *initiator queries* for the intrinsically diverse sessions.

Given a set of intrinsically diverse queries, the query among them that is most general and likely to have been the first among these set of queries is called the initiator query. If multiple such queries exist, then the first among them from the actual sequence (issued by the user) is considered the initiator.

Labeling Guidelines

You will be given a set of queries from a single user session (in the order they were queried). The session will also contain information about the underlying topic behind these queries, namely the title, description and narration of the topic.

Based on this information you will be asked to label each of these sessions with the following information:

(a) **Is the topic intrinsically diverse?:** Based only on the topic (i.e., title, description and narration) please label if you believe the session is intrinsically diverse or not **(Yes/No/Maybe)**.
(b) **Are the queries intrinsically diverse?:** Now taking into account the queries (along with the topic), please label if you believe the set of queries is intrinsically diverse or not **(Yes/No/Maybe)**. **NOTE:** Even if the topic is intrinsically diverse, the queries need not be intrinsically diverse. For example, when the queries are simply spelling corrections or reformulations, they are not intrinsically diverse regardless of the topic being intrinsically diverse or not.
(c) **Best Initiator Query:** For all sessions (including those marked as **No** for **b)** please indicate which of the queries you would consider to be the best initiator query.
(d) **Comment:** If you have any additional comments for any of the sessions, then please enter it in the comment field.

ACKNOWLEDGMENTS

The authors would like to thank Dan Liebling for his constant support in data collection and filtering.

REFERENCES

Eugene Agichtein, Ryen W. White, Susan T. Dumais, and Paul N. Bennett. 2012. Search, interrupted: Understanding and predicting search task continuation. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*. ACM, New York, NY, 315–324. DOI: http://dx.doi.org/10.1145/2348283.2348328

Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying search results. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining (WSDM'09)*. ACM, New York, NY, 5–14. DOI: http://dx.doi.org/10.1145/1498759.1498766

Peter Bailey, Liwei Chen, Scott Grosenick, Li Jiang, Yan Li, Paul Reinholdtsen, Charles Salada, Haidong Wang, and Sandy Wong. 2012. User task understanding: A web search engine perspective. http://research.microsoft.com/apps/-pubs/default.aspx?id=180594.

Peter Bailey, Nick Craswell, Ryen W. White, Liwei Chen, Ashwin Satyanarayana, and S. M. M. Tahaghoghi. 2010. Evaluating whole-page relevance. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. ACM, New York, NY, 767–768. DOI: http://dx.doi.org/10.1145/1835449.1835606

Peter L. Bartlett, Michael I. Jordan, and Jon M. Mcauliffe. 2004. Large margin classifiers: Convex loss, low noise, and convergence rates. In *Advances in Neural Information Processing Systems 16*. MIT Press, 1173–1180. DOI: http://papers.nips.cc/paper/2416-large-margin-classifiers-convex-loss-low-noise-and-convergence-rates.pdf.

Paul N. Bennett, Krysta Svore, and Susan T. Dumais. 2010. Classification-enhanced ranking. In *Proceedings of the 19th International Conference on World Wide Web (WWW'10)*. ACM, New York, NY, 111–120. DOI: http://dx.doi.org/10.1145/1772690.1772703

Christina Brandt, Thorsten Joachims, Yisong Yue, and Jacob Bank. 2011. Dynamic ranked retrieval. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM'11)*. ACM, New York, NY, 247–256. DOI: http://dx.doi.org/10.1145/1935826.1935872

Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. 2011. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.* 40, 6 (2011), 1740–1766.

Gabriele Capannini, Franco Maria Nardini, Raffaele Perego, and Fabrizio Silvestri. 2011. Efficient diversification of web search results. *Proc. VLDB Endow.* 4, 7 (2011), 451–459. http://dl.acm.org/citation.cfm?id=1988776.1988781.

Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*. ACM, New York, NY, 335–336. DOI: http://dx.doi.org/10.1145/290941.291025

Harr Chen and David R. Karger. 2006. Less is more: Probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*. ACM, New York, NY, 429–436. DOI: http://dx.doi.org/10.1145/1148170.1148245

Charles L. A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*. ACM, New York, NY, 659–666. DOI: http://dx.doi.org/10.1145/1390334.1390446

Charles L. Clarke, Maheedhar Kolla, and Olga Vechtomova. 2009. An effectiveness measure for ambiguous and underspecified queries. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory (ICTIR'09)*. Springer-Verlag, Berlin, Heidelberg, 188–199. DOI: http://dx.doi.org/10.1007/978-3-642-04417-5_17

Wisam Dakka, Rishabh Dayal, and Panagiotis G. Ipeirotis. 2006. Automatic discovery of useful facet terms. In *Proceedings of the SIGIR Workshop on Faceted Search*.

Van Dang, Michael Bendersky, and W. Bruce Croft. 2010. Learning to rank query reformulations. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. ACM, New York, NY, 807–808. DOI: http://dx.doi.org/10.1145/1835449.1835626

Van Dang and Bruce W. Croft. 2013. Term level search result diversification. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 603–612. DOI: http://dx.doi.org/10.1145/2484028.2484095

Van Dang and W. Bruce Croft. 2012. Diversity by proportionality: An election-based approach to search result diversification. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*. ACM, New York, NY, 65–74. DOI: http://dx.doi.org/10.1145/2348283.2348296

Fabio De Bona, Stefan Riezler, Keith Hall, Massimiliano Ciaramita, Amaç Herdağdelen, and Maria Holmqvist. 2010. Learning dense models of query similarity from user click logs. In *Proceedings of the NAACL Conference on Human Language Technologies (HLT'10)*. Association for Computational Linguistics, Stroudsburg, PA, 474–482. http://dl.acm.org/citation.cfm?id=1857999.1858070

Georges Dupret, Ricardo Zilleruelo-Ramos, and Sumio Fujita. 2010. Using related queries to improve Web search results ranking. In *Proceedings of the 17th International Symposium on String Processing and Information Retrieval (SPIRE)*. Lecture Notes in Computer Science, Vol. 6393, Springer, Berlin, 213–224.

Henry Feild and James Allan. 2013. Task-aware query recommendation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 83–92. DOI: http://dx.doi.org/10.1145/2484028.2484069

M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey. 1978. An analysis of approximations for maximizing submodular set functionsII. In *Polyhedral Combinatorics*, M. L. Balinski and A. J. Hoffman (Eds.), Mathematical Programming Studies, Vol. 8, Springer, Berlin Heidelberg, 73–87. DOI: http://dx.doi.org/10.1007/BFb0121195

Steve Fox, Kuldeep Karnawat, Mark Mydland, Susan Dumais, and Thomas White. 2005. Evaluating implicit measures to improve web search. *ACM Trans. Inf. Syst.* 23, 2 (2005), 147–168. DOI: http://dx.doi.org/10.1145/1059981.1059982

Sumio Fujita, Georges Dupret, and Ricardo A. Baeza-Yates. 2012. Learning to rank query recommendations by semantic similarities. *CoRR* abs/1204.2712 (2012).

Jianfeng Gao, Wei Yuan, Xiao Li, Kefeng Deng, and Jian-Yun Nie. 2009. Smoothing clickthrough data for web search ranking. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'09)*. ACM, New York, NY, 355–362. DOI: http://dx.doi.org/10.1145/1571941.1572003

Sreenivas Gollapudi, Samuel Ieong, Alexandros Ntoulas, and Stelios Paparizos. 2011. Efficient query rewrite for structured web queries. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM'11)*. ACM, New York, NY, 2417–2420. DOI: http://dx.doi.org/10.1145/2063576.2063981

Dongyi Guan, Sicong Zhang, and Hui Yang. 2013. Utilizing query change for session search. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 453–462. DOI: http://dx.doi.org/10.1145/2484028.2484055

Jiafeng Guo, Xueqi Cheng, Gu Xu, and Xiaofei Zhu. 2011. Intent-aware query similarity. In *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'11)*. ACM, New York, NY, 259–268. DOI: http://dx.doi.org/10.1145/2063576.2063619

Ahmed Hassan, Yang Song, and Li-Wei He. 2011. A task level metric for measuring web search satisfaction and its application on improving relevance estimation. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM'11)*. ACM, New York, NY, 125–134. DOI: http://dx.doi.org/10.1145/2063576.2063599

Ahmed Hassan and Ryen W. White. 2012. Task tours: Helping users tackle complex search tasks. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM'12)*. ACM, New York, NY, 1885–1889. DOI: http://dx.doi.org/10.1145/2396761.2398537

Jiyin He, Marc Bron, and Arjen P. de Vries. 2013. Characterizing stages of a multi-session complex search task through direct and indirect query modifications. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 897–900. DOI: http://dx.doi.org/10.1145/2484028.2484178

Jiyin He, Vera Hollink, Corrado Boscarino, Arjen P. de Vries, and Roberto Cornacchia. 2011. CWI at TREC 2011: Session, Web, and Medical. In *Proceedings of the 20th Text Retrieval Conference (TREC'11)*.

Kalervo Järvelin, Susan L. Price, Lois M. L. Delcambre, and Marianne Lykke Nielsen. 2008. Discounted cumulated gain based evaluation of multiple-query IR sessions. In *Proceedings of the 30th European Conference on Advances in Information Retrieval (ECIR'08)*. Lecture Notes in Computer Science, Vol. 4956, Springer-Verlag, Berlin, Heidelberg, 4–15. http://dl.acm.org/citation.cfm?id=1793274.1793280

Thorsten Joachims. 1999. Making large-scale support vector machine learning practical. In *Advances in Kernel Methods*, Bernhard Schölkopf, Christopher J. C. Burges, and Alexander J. Smola (Eds.), MIT Press, 169–184. http://dl.acm.org/citation.cfm?id=299094.299104.

Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, Filip Radlinski, and Geri Gay. 2007. Evaluating the accuracy of implicit feedback from clicks and query reformulations in Web search. *ACM Trans. Inf. Syst.* 25, 2, Article 7 (2007). DOI: http://dx.doi.org/10.1145/1229179.1229181

Rosie Jones and Kristina Lisa Klinkner. 2008. Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM'08)*. ACM, New York, NY, 699–708. DOI: http://dx.doi.org/10.1145/1458082.1458176

Evangelos Kanoulas, Ben Carterette, Paul D. Clough, and Mark Sanderson. 2010. Overview of the TREC 2010 session track. In *Proceedings of the 19th Text Retrieval Conference (TREC)*.

Evangelos Kanoulas, Ben Carterette, Paul D. Clough, and Mark Sanderson. 2011a. Evaluating multi-query sessions. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11)*. ACM, New York, NY, 1053–1062. DOI: http://dx.doi.org/10.1145/2009916.2010056

Evangelos Kanoulas, Mark M. Hall, Paul Clough, Ben Carterette, and Mark Sanderson. 2011b. Overview of the TREC 2011 session track. In *Proceedings of the 20th Text Retrieval Conference (TREC)*.

Christian Kohlschutter, Paul-Alexandru Chirita, and Wolfgang Nejdl. 2006. Using link analysis to identify aspects in faceted web search. In *Proceedings of the International ACM SIGIR Conference on Research and Development Information Retrieval (SIGIR'06)*.

Weize Kong and James Allan. 2013. Extracting query facets from search results. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 93–102. DOI: http://dx.doi.org/10.1145/2484028.2484097

Alexander Kotov, Paul N. Bennett, Ryen W. White, Susan T. Dumais, and Jaime Teevan. 2011. Modeling and analysis of cross-session search tasks. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11)*. ACM, New York, NY, 5–14. DOI: http://dx.doi.org/10.1145/2009916.2009922

Zhen Liao, Yang Song, Li-Wei He, and Yalou Huang. 2012. Evaluating the effectiveness of search task trails. In *Proceedings of the 21st International Conference on World Wide Web (WWW'12)*. ACM, New York, NY, 489–498. DOI: http://dx.doi.org/10.1145/2187836.2187903

Daniel J. Liebling, Paul N. Bennett, and Ryen W. White. 2012. Anticipatory search: Using context to initiate search. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*. ACM, New York, NY, 1035–1036. DOI: http://dx.doi.org/10.1145/2348283.2348456

Chang Liu, Si Sun, Michael J. Cole, and Nicholas J. Belkin. 2011. Rutgers at the TREC 2011 session track. In *Proceedings of the 20th Text Retrieval Conference (TREC)*.

Jingjing Liu and Nicholas J. Belkin. 2010. Personalizing information retrieval for multi-session tasks: The roles of task stage and task type. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. ACM, New York, NY, 26–33. DOI: http://dx.doi.org/10.1145/1835449.1835457

Jingjing Liu, Michael J. Cole, Chang Liu, Ralf Bierig, Jacek Gwizdka, Nicholas J. Belkin, Jun Zhang, and Xiangmin Zhang. 2010. Search behaviors in different task types. In *Proceedings of the 10th Annual Joint Conference on Digital Libraries (JCDL'10)*. ACM, New York, NY, 69–78. DOI: http://dx.doi.org/10.1145/1816123.1816134

Claudio Lucchese, Salvatore Orlando, Raffaele Perego, Fabrizio Silvestri, and Gabriele Tolomei. 2011. Identifying task-based sessions in search engine query logs. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM'11)*. ACM, New York, NY, 277–286. DOI: http://dx.doi.org/10.1145/1935826.1935875

Hao Ma, Michael R. Lyu, and Irwin King. 2010. Diversifying query suggestion results. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence (AAAI'10)*. AAAI Press.

Gary Marchionini. 2006. Exploratory search: From finding to understanding. *Commun. ACM* 49, 4 (April 2006), 41–46. DOI: http://dx.doi.org/10.1145/1121949.1121979

Dan Morris, Meredith Ringel Morris, and Gina Venolia. 2008. SearchBar: A search-centric web history for task resumption and information re-finding. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'08)*. ACM, New York, NY, 1207–1216. DOI: http://dx.doi.org/10.1145/1357054.1357242

Jeffrey Pound, Stelios Paparizos, and Panayiotis Tsaparas. 2011. Facet discovery for structured web search: A query-log mining approach. In *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'11)*. ACM, New York, NY, 169–180. DOI: http://dx.doi.org/10.1145/1989323.1989342

Pernilla Qvarfordt, Gene Golovchinsky, Tony Dunnigan, and Elena Agapie. 2013. Looking ahead: Query preview in exploratory search. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 243–252. DOI: http://dx.doi.org/10.1145/2484028.2484084

Filip Radlinski, Paul N. Bennett, Ben Carterette, and Thorsten Joachims. 2009. Redundancy, diversity and interdependent document relevance. *SIGIR Forum* 43, 2 (2009), 46–52. DOI: http://dx.doi.org/10.1145/1670564.1670572

Filip Radlinski and Susan Dumais. 2006. Improving personalized web search using result diversification. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'06)*. ACM, New York, NY, 691–692. DOI: http://dx.doi.org/10.1145/1148170.1148320

Filip Radlinski and Thorsten Joachims. 2005. Query chains: Learning to rank from implicit feedback. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining (KDD'05)*. ACM, New York, NY, 239–248. DOI: http://dx.doi.org/10.1145/1081870.1081899

Karthik Raman, Paul N. Bennett, and Kevyn Collins-Thompson. 2013. Toward whole-session relevance: Exploring intrinsic diversity in web search. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 463–472. DOI: http://dx.doi.org/10.1145/2484028.2484089

Karthik Raman and Thorsten Joachims. 2013. Learning socially optimal information systems from egoistic users. In *Machine Learning and Knowledge Discovery in Databases*, Hendrik Blockeel, Kristian Kersting, Siegfried Nijssen, and Filip elezn (Eds.), Lecture Notes in Computer Science, Vol. 8189, Springer, Berlin Heidelberg, 128–144. DOI: http://dx.doi.org/10.1007/978-3-642-40991-2_9

Karthik Raman, Thorsten Joachims, and Pannaga Shivaswamy. 2011. Structured learning of two-level dynamic rankings. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM'11)*. ACM, New York, NY, 291–296. DOI: http://dx.doi.org/10.1145/2063576.2063623

Karthik Raman, Pannaga Shivaswamy, and Thorsten Joachims. 2012. Online learning to diversify from implicit feedback. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'12)*. ACM, New York, NY, 705–713. DOI: http://dx.doi.org/10.1145/2339530.2339642

Eldar Sadikov, Jayant Madhavan, Lu Wang, and Alon Halevy. 2010. Clustering query refinements by user intent. In *Proceedings of the 19th International Conference on World Wide Web (WWW'10)*. ACM, New York, NY, 10. DOI: http://dx.doi.org/10.1145/1772690.1772776

Tetsuya Sakai and Zhicheng Dou. 2013. Summaries, ranked retrieval and sessions: A unified framework for information access evaluation. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. ACM, New York, NY, 473–482. DOI: http://dx.doi.org/10.1145/2484028.2484031

Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. 2010a. Exploiting query reformulations for web search result diversification. In *Proceedings of the World Wide Web Conference (WWW'10)*. ACM, New York, NY, 881–890. DOI: http://dx.doi.org/10.1145/1772690.1772780

Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. 2010b. Selectively diversifying web search results. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM'10)*. ACM, New York, NY, 1179–1188. DOI: http://dx.doi.org/10.1145/1871437.1871586

Adish Singla, Ryen White, and Jeff Huang. 2010. Studying trailfinding algorithms for enhanced web search. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. ACM, New York, NY, 443–450. DOI: http://dx.doi.org/10.1145/1835449.1835524

Aleksandrs Slivkins, Filip Radlinski, and Sreenivas Gollapudi. 2010. Learning optimally diverse rankings over large document collections. In *Proceedings of the 27th International Conference on Machine Learning (ICML'10)*. http://www.icml2010.org/papers/446.pdf.

Mark D. Smucker and Charles L. A. Clarke. 2012. Time-based calibration of effectiveness measures. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*. ACM, New York, NY, 95–104. DOI: http://dx.doi.org/10.1145/2348283.2348300

Yang Song, Dengyong Zhou, and Li-Wei He. 2011. Post-ranking query suggestion by diversifying search results. In *Proceedings of the ACM Conference on Research and Development in Information Retrieval (SIGIR'11)*. ACM, New York, NY, 815–824. DOI: http://dx.doi.org/10.1145/2009916.2010025

Ashwin Swaminathan, Cherian V. Mathew, and Darko Kirovski. 2009. Essential pages. In *Proceedings of the IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'09)*, Vol. 1, IEEE Computer Society, 173–182. DOI: http://dx.doi.org/10.1109/WI-IAT.2009.33

Ryen W. White, Paul N. Bennett, and Susan T. Dumais. 2010. Predicting short-term interests using activity-based search context. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM'10)*. ACM, New York, NY, 1009–1018. DOI: http://dx.doi.org/10.1145/1871437.1871565

Ryen W. White and Steven M. Drucker. 2007. Investigating behavioral variability in web search. In *Proceedings of the 16th International Conference on World Wide Web (WWW'07)*. ACM, New York, NY, 21–30. DOI: http://dx.doi.org/10.1145/1242572.1242576

Ryen W. White, Bill Kules, Steven M. Drucker, and M. C. Schraefel. 2006. Supporting exploratory search, introduction. *Commun. ACM* 49, 4 (April 2006), 36–39. DOI: http://eprints.soton.ac.uk/263649/.

Ryen W. White, Gary Marchionini, and Gheorghe Muresan. 2008. Editorial: Evaluating exploratory search systems. *Inf. Process. Manag.* 44, 2 (2008), 433–436. DOI: http://dx.doi.org/10.1016/j.ipm.2007.09.011

Xiaojun Yuan and Ryen White. 2012. Building the trail best traveled: Effects of domain knowledge on web search trailblazing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'12)*. ACM, New York, NY, USA, 1795–1804. DOI: http://dx.doi.org/10.1145/2207676.2208312

Yisong Yue and Thorsten Joachims. 2008. Predicting diverse subsets using structural SVMs. In *Proceedings of the 25th International Conference on Machine Learning (ICML'08)*. ACM, New York, NY, 8. DOI: http://dx.doi.org/10.1145/1390156.1390310

Cheng Xiang Zhai, William W. Cohen, and John Lafferty. 2003. Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'03)*. ACM, New York, NY, 10–17. DOI: http://dx.doi.org/10.1145/860435.860440

Lanbo Zhang and Yi Zhang. 2010. Interactive retrieval based on faceted feedback. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. ACM, New York, NY, 363–370. DOI: http://dx.doi.org/10.1145/1835449.1835511

Qiankun Zhao, Steven C. H. Hoi, Tie-Yan Liu, Sourav S. Bhowmick, Michael R. Lyu, and Wei-Ying Ma. 2006. Time-dependent semantic similarity measure of queries using historical click-through data. In *Proceedings of the 15th International Conference on World Wide Web (WWW'06)*. ACM, New York, NY, 543–552. DOI: http://dx.doi.org/10.1145/1135777.1135858

Received October 2013; revised February 2014; accepted May 2014