Inf Retrieval (2011) 14:572–592
DOI 10.1007/s10791-011-9167-7

Intent-based diversification of web search results: metrics and algorithms

Olivier Chapelle · Shihao Ji · Ciya Liao · Emre Velipasaoglu · Larry Lai · Su-Lin Wu

Received: 18 October 2010 / Accepted: 28 April 2011 / Published online: 20 May 2011

Abstract We study the problem of web search result diversification in the case where intent based relevance scores are available. A diversified search result will hopefully satisfy the information need of user-L.s who may have different intents. In this context, we first analyze the properties of an intent-based metric, ERR-IA, to measure relevance and diversity altogether. We argue that this is a better metric than some previously proposed intent aware metrics and show that it has a better correlation with abandonment rate. We then propose an algorithm to rerank web search results based on optimizing an objective function corresponding to this metric and evaluate it on shopping related queries.

Keywords Web search · Ranking · Relevance · Diversification

S. Ji and C. Liao contributed to the paper while working at Yahoo! Labs.

O. Chapelle (✉) · E. Velipasaoglu · L. Lai · S.-L. Wu
Yahoo! Labs, Sunnyvale, CA, USA
e-mail: chap@yahoo-inc.com

E. Velipasaoglu
e-mail: emrev@yahoo-inc.com

L. Lai
e-mail: larrylai@yahoo-inc.com

S.-L. Wu
e-mail: sulin@yahoo-inc.com

S. Ji
Microsoft Bing, Bellevue, WA, USA
e-mail: shihaoji@microsoft.com

C. Liao
Microsoft Bing, Mountain View, CA, USA
e-mail: cliao@microsoft.com

1 Introduction

When they interact with a search engine, users typically enter short queries and there is often ambiguity in the intent of the user. In the case of a common query such as ‘garmin gps’ a user might be interested in a variety of things ranging from places to purchase one, reviews about it, instructions on how to use it, help with repairing the device, or recent product news. In many cases it is not *a priori* clear what the intent of the user might be and it is necessary for search engines to display a good variety of results covering different intents to ensure that all users are satisfied, with the hope that the user will find at least one relevant document for his information need. The problem of the user not finding any relevant document, indicated by no clicks, is defined as query abandonment. Current research on result set diversification mainly aims at minimizing query abandonment by providing diversified search results in top positions (Clarke et al. 2008; Sarma et al. 2008; Agrawal et al. 2009).

Carbonell and Goldstein (1998) proposed Maximal Marginal Relevance (MMR) for increasing diversity. MMR uses an objective function that explicitly trades-off relevance against novelty of search results. The novelty is measured through a similarity function between documents. The level of diversity is controlled explicitly by the trade-off parameter and implicitly by the similarity measure between the documents. The similarity measure is the knob for controlling the saturation levels of different types of documents (i.e. how many documents for each user intent are admitted into the result set). The similarity measure used in MMR is query and intent independent, which renders the control over the match between relative saturation levels and user intents difficult at best.

Agrawal et al. (2009) and Clarke et al. (2008) circumvented this difficulty by using document and query categories. Aligning the categories with the user intents can facilitate finer control over the representation of different topics in the diversified result sets. In particular, Clarke et al. (2008) explicitly partitions the information needs in so-called *nuggets* which can be mapped to intents in our framework. The recent paper of Agrawal et al. (2009) can be seen as an extension of that work and is closely related to ours. They propose an algorithm that minimizes the risk of dissatisfaction of the average user. The work explicitly considers user intent or query topics with different importance. However, they do not have specific intent based relevance estimates.

We approach the diversification problem in the framework of intent-based ranking, in which each intent has a specialized ranking function for evaluating document intent-based relevance, and query intent distribution is estimated directly from click logs. Within that framework, a novel diversification metric—intent-aware expected reciprocal rank (ERR-IA)—is presented. This metric has first been introduced by Chapelle et al. (2009, Section 7.4), but only briefly. We analyze here thoroughly its properties for diversification and argue that it is a better metric than previously proposed metrics, such as DCG-IA and MAP-IA (Agrawal et al. 2009). We also show that ERR-IA is a generalization of α-NDCG (Clarke et al. 2008). It is noteworthy that ERR-IA was one the metrics used to evaluate the diversity task in the Web track of TREC 2010.[1] We finally explore different ways to rerank the result set by optimizing ERR-IA directly, and both a greedy algorithm and an optimal one are presented.

The rest of the paper is organized as follows. In Sect. 2, we review the related work in result set diversification, draw the connection among different approaches, and connect them with our proposed method. Section 3 introduces the intent-based rankings which are at the core of our diversification framework. Section 4 proposes the ERR-IA metric and analyzes its properties as compared to other metrics. Section 5 explores a greedy algorithm

[1] See http://plg.uwaterloo.ca/~trecweb/2010.html

and a non-greedy algorithm (i.e., Branch and Bound) to directly optimize ERR-IA. Experiments results are provided in Sect. 6. Finally, as a complement to the intent-based ranking framework for diversification, Sect. 7 proposes an alternative framework for content based diversification that can be explored in future work.

2 Related work

As mentioned in the introduction, MMR is a seminal work for results diversification and it led to several follow-ups. Gollapudi and Sharma (2009) propose a set of natural axioms for result diversification that aid in the choice of MMR-like objective functions; Wang and Zhu (2009); Rafiei et al. (2010) use a portfolio approach to increase relevance measured by return while decrease similarity measured by risk. Zhai et al. (2003) study both novelty and relevancy in the language modeling framework. They propose an evaluation framework for subtopic retrieval, based on the metrics of subtopic recall and subtopic precision. They also propose a cost based approach to combine relevance and novelty in the same spirit as MMR. All these works share the shortcoming of MMR mentioned in the introduction, namely that the similarity function is query and intent independent.

Chen and Karger (2006) use standard IR techniques to improve diversity for ambiguous queries. In this work, documents are selected sequentially according to relevance. The relevance is conditioned on documents having been already selected. Words in previous documents are associated with a negative weight to improve novelty. Our work differs in that it explicitly considers different query intents, so that the result set covers multiple query intents.

Radlinski et al. (2008) directly learn a diverse ranking of results based on users' clicking behavior through online exploration. They maximize the probability that a relevant document is found in the top *k* positions of a ranking. Since users tend not to click similar documents, online learning produces a diverse set of documents naturally. This is very appealing approach, but it cannot readily be used for tail queries since it requires user feedback.

We have already reviewed Clarke et al. (2008) and Agrawal et al. (2009) in the introduction. The latter extends the former and we mainly differ from them by the incorporation of specific intent based relevance functions. Both papers also felt the urge to develop new metrics, as the literature on effective metrics that take into account both diversity and relevance is thin. In this work, we are extending the metrics that they have introduced and argue that ERR-IA is well suited for this purpose.

Finally, the recent works of Santos and colleagues (2010a, b) as well as Carterette and Chandar (2009) have an objective function for diversification similar to the one of Agrawal et al. (2009) and ours. Two of the main differences are the query intent distribution, that we estimate from click logs, and the intent based relevance scores, that are generated using a learning to rank approach instead of language models or other standard retrieval models. Also one of our main contribution is the link between a well funded metric for diversity, ERR-IA, and the objective function to be optimized. In the aforementioned works, the evaluation metrics and the objective function were not directly connected.

3 The intent-based ranking framework

In this study, we focus on shopping related queries. There is a clear business incentive for a web search engine to improve this class of queries because they generate a lot of revenues. We first describe the five intents that have been identified for this type of queries.

3.1 Intents

The five intents that we consider for shopping queries are described as follows:

Buying guide Intent is to retrieve a document that explains how to buy a particular category of products. It might contain main features and considerations, important terminology, where and how to purchase.

Reviews intent is to retrieve documents for evaluations of products that could include ratings, recommendations, and explanations of the author's point of view of the product.

Support intent is to retrieve documents containing technical detail about a product that helps a user in using the item. For example, manuals, trouble shooting pages, tutorials and warranties.

Official product homepage Intent is to retrieve a document that describes the specific product at the manufacturer's domain.

Shopping site/purchase Intent is to retrieve documents from sites that give the user an opportunity to purchase the product online.

Finally there is also a "general" intent defined as a catch all for all user needs.

These intents were selected to correspond with an idealization of the shopping process. In this abstract view, the consumer first examines buying guides to understand how to shop for a desired product, then consults reviews and goes to the official product homepage. Finally the consumer makes their purchase at a shopping site, and uses support pages for post-purchase information. In practice, these intents are not orthogonal; consumers can be in multiple stages and documents can satisfy multiple intents at the same time.

In our experiments, however, these shopping intents are actually fairly separable as illustrated by the graphs in Fig. 1. To create these plots, we scored the same set of documents with each of our 5 intent-specific models; the intent-specific models were trained to render a higher score for documents that are relevant to a given query and a specified intent (described in more detail in Sect. 3.3). For example, an Amazon review page for a particular dvd player would get a higher score from the Reviews model than from the Support model for the same query. Each graph in Fig. 1 is a scatterplot comparing the scores from one intent-specific model with another; there is one graph for each possible pair of our five intents. For example, in the bottom-most plot, the scores of the Support model are plotted against the scores from the Homepage model. From the mass of points that align roughly to the axes, we can see that many documents receive high scores from only one of the two intent models. We take this as an indication that the intent-specific models can and do discriminate between intents in these shopping-related documents. Some intents such as Homepage and Review show almost orthogonal scores, while other intents show a lesser degree of differentiation, such as Homepage and Purchase.

3.2 Intent mining

The query intent classification method of Agrawal et al. (2009) is based on smoothing class labels of a seed set over the bipartite graph of query-document click logs by random walk. However, users most often click on a top level page in a site, and navigate to a deeper page better revealing their true intent. In such cases, post-search browsing activity becomes crucial for query intent determination. Our intent mining method takes this into account by joining the data from toolbar and query logs. To better capture user's web activity, we construct search trees where tree nodes represented user's search events (i.e., queries, clicks, and search result links presented to the user) and tree edges denote the referral

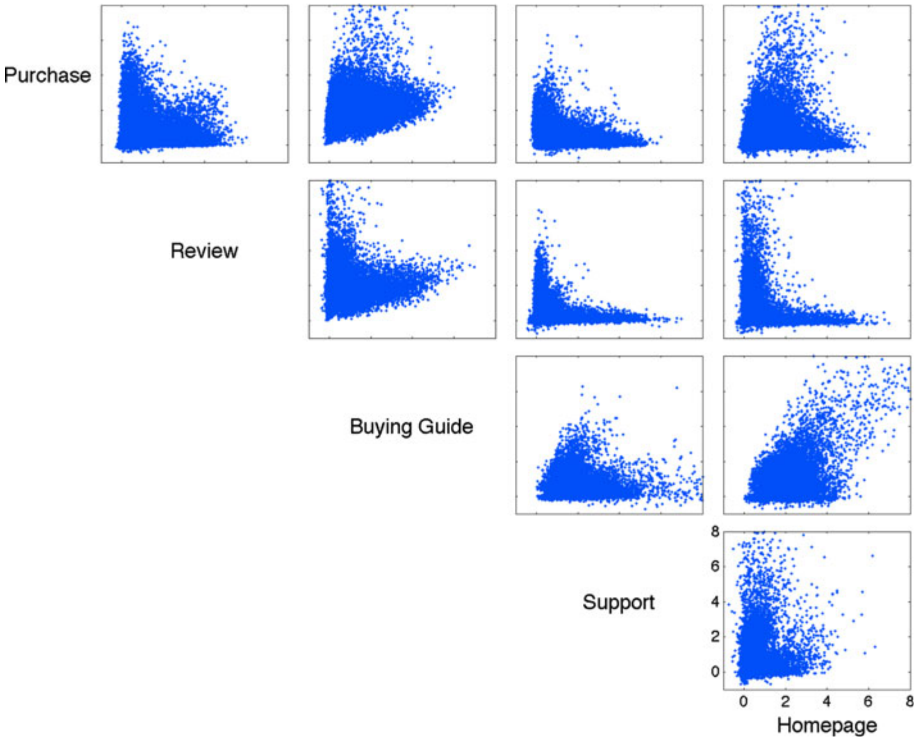

Fig. 1 Scatter plots of the intent based relevance scores on the set of documents used for the experimental evaluation (see Sect. 6)

relationship between nodes. In our search log, we track the causal (i.e., referral) relationship among search events. By using the referral information between events, we can disambiguate interleaved search sessions conducted by the same user (e.g., users who conduct multiple searches simultaneously using multiple tabs/windows) and put them in different branches of the resulting trees. In the tree construction process, we basically use the referral information to connect the search events. The resulting trees not only respect the temporal relationship but also respect causal relationship between user's search events. Also, we do not merge web events visiting the same url into a single node, therefore, there is no cycle in the constructed tree. The resulting data structure becomes disjoint trees (i.e., forest) rather than a graph. There are several advantages with this representation. First, all interleaving query sessions (e.g., users with multiple search tabs/windows opened simultaneously) are well separated. Second, the related clicks are grouped in the same branch of search tree.

We used five page category classifiers (i.e., for the five shopping intents) trained beforehand and to estimate the strength of a web page belonging to a specific category (i.e., intent). For example, an iPad review page on amazon.com would receive high scores on both review and shopping categories but low scores on other categories. The classifiers used all the document ranking features available to our search engine. These features include those features extracted from html structure of a page, page content, linking structures of the page (i.e., inlinks and outlinks), and click/view data on the page if available. There were about a thousand of features for those classifiers. We collected our

Table 1 Intent distribution averaged over a set of 406 queries

Intent	Average prob. (%)
Purchase	17.5
Review	7.4
Buying guide	1.1
Support	8.6
Home page	22.1
General	43.2

training examples editorially and improve our training set using active learning technique. We then train our page category classifiers using those training examples.

To determine the query intent for a given search tree, we first filter out those pages whose dwell times are less than one second. This is because we believe that users could not spend less than one second on pages that interest them. For the remaining pages under the query search tree, we pull out their five category scores and then compute the average scores weighted by their page dwell times across pages. Here, we give higher weights to those page clicks with higher dwell times since page dwell time is a good implicit indicator of its page utility (Claypool et al. 2001; Liu et al. 2006). The resulting five aggregated scores give us the estimation of intent strength of the query. We then give a binary label to each intent by comparing its score against a threshold tuned using the editorially judged data set described in Sect. 6 For the cases where all five scores are below the thresholds, we say it has the "general" intent (i.e. catch-all intent). As a result, each processed query is labeled with a binary vector of length 6 representing the five shopping intents together with the general intent. To compute the intent distribution for a specific query, we simply mine one year of query logs, extract all search trees containing that query, and compute the average of these binary vectors across all search trees. We finally normalize the resulting six intent probabilities so that they sum to 1.

The intent distribution, averaged over 406 queries that will be used in the evaluation section, is shown in Table 1.

3.3 Intent-specific ranking

We also improve the intent dependent relevance estimate of (Agrawal et al. 2009). In that paper, it is somewhat naively assumed that the intent-dependent relevance of a query-document pair can be approximated by intent-independent relevance weighted by the intent likelihood of the document. In other words, $P(rel|d, q, i) = P(rel|d, q)\ P(i|d)$, where q, d, and i stand for query, document and intent, respectively. It is easy to see the shortcoming of this approximation with an example. Take a document d answering perfectly a rare intent i. Then $P(rel|d, q, i)$ should be large, but $P(rel|d, q)$ will be small because $P(i|q)$ is small (since the intent is rare). In our work, instead of the approximation above, we develop a dedicated machine learned ranking model for each intent that directly estimates $P(rel|d, q, i)$.

These intent-specific functions are built using stochastic gradient boosted decisions trees for regression. This is similar to (Zheng et al. 2008) except that we use a pointwise loss instead of a pairwise one. For each intent, a small ($\sim$6,000 queries) specialized training set is built by obtaining intent-conditioned editorial judgments for sample query-document pairs. For the general intent, a large ($\sim$30,000 queries) training set is used where the editorial judgments are not conditioned on intent. To provide robustness to non-targeted intents in the intent-specific functions, the small intent-based training sets are combined with the large general training set, and the performance is optimized over a held-out set of intent-based judgments.

The judgments were provided by in-house professional editors trained to carefully designed guidelines. For non-intent-specific relevance judgments, the editors are instructed to estimate, based on ad hoc research or their own experience, the most likely user intentions for a given query and grade the relevance of documents based on their holistic assessment. For intent-specific editorial judgments, the user intention is given; the editors were instructed to imagine themselves in a particular stage of the shopping process while judging a query and document pair, and to ignore the value of the document for other stages.

4 Intent aware metrics

We review in this section several metrics which have been recently proposed to take into account the diversification of search results and we also provide some theoretical analysis of these metrics. But first, we need to introduce standard information retrieval metrics.

4.1 Standard information retrieval metrics

We are given an ordered list of n documents and the relevance label for the k-th document is r_k. In the binary case, $r_k \in \{0, 1\}$, but r_k can belong to an ordinal set in the case of graded relevance: $r_k \in \{0, 1, 2, 3, 4\}$ for instance.

4.1.1 Average precision

Average precision (AP) is a commonly used metric for binary relevance judgments. It is defined as the average of the precisions at the positions where there is a relevant document:

$$\frac{1}{R}\sum_k r_k \frac{1}{k}\sum_{j \le k} r_j,$$

where $R = \sum r_k$ is the number of relevant documents. We assume R to be constant. We refer below to this metric as AP instead of MAP (mean average precision) because we consider a single query.

4.1.2 Discounted cumulative gain

The Discounted Cumulative Gain (DCG) score (Jarvelin and Kekalainen 2002) is a popular measure for multi-level relevance judgments. In its basic form it has a logarithmic position discount: the benefit of seeing a relevant document at position k is $1/\log_2(k+1)$. Following (Burges et al. 2005), it became usual to assign exponentially high weight $2^{r_k} - 1$ to highly rated documents. Thus the DCG is defined as:

$$\sum_{k=1}^{n} \frac{\mathcal{R}(r_k)}{\log_2(k+1)},$$

with

$$\mathcal{R}(r) = \frac{2^r - 1}{2^{r_{\max}}}, \tag{1}$$

Algorithm 1 The cascade user model

```
Require: $R_1, \ldots, R_{10}$ the relevance of the 10 documents on the result page.
$i = 1$
User examines position $i$.
if random(0,1) $\leq R_i$ then
    User is satisfied with the $i$-th document and stops.
else
    $i \leftarrow i + 1$; go to 2
end if
```

$r_{\max}$ being the highest relevance level. The normalization by $2^{r_{\max}}$ is inconsequential in the definition of the DCG, but we introduce it because the same function $\mathcal{R}$ is used in the definition of the following metric.

4.1.3 Cascade-based metrics

Cascade-based metrics have been introduced in (Chapelle et al. 2009) and are based on the *cascade* user model first proposed in (Craswell et al. 2008) and described in Algorithm 1.

The idea of a cascade-based metric is to use the relevance labels to estimate the probability that the user will be satisfied[2] by the document in position i. In particular, it has been suggested in (Chapelle et al. 2009) to estimate R_i as $\mathcal{R}(r_i)$.

Given a decreasing utility function φ, the metric is defined as the expectation of $\varphi(k)$, where k is the position where the user finds the document that satisfies him. This quantity turns out to be:

$$\sum_k \varphi(k)\mathcal{R}(r_k)\prod_{j=1}^{k-1}(1-\mathcal{R}(r_j)). \tag{2}$$

The *Expected Reciprocal Rank* (ERR) metric (Chapelle et al. 2009) is an instantiation of (2) with $\varphi(k) = 1/k$.

4.2 Intent aware metrics

In the case of multiple intents, let r_k^i be the relevance label of the k-th document with respect to the i-th intent. Also let p_i be the probability that a user would be interested in the i-th intent for that query.

Given an information retrieval metric, it has been suggested in (Agrawal et al. 2009) to define its *intent aware* version as the expectation of the metric over the intents. For instance, DCG-IA is defined as:

$$\sum_i p_i \sum_j \frac{\mathcal{R}(r_j^i)}{\log_2(j+1)}. \tag{3}$$

[2] We refer, in the rest of the paper, to this probability as a *satisfaction* probability because of the underlying cascade user model. It can also be understood as a *relevance* probability.

Similarly, ERR-IA is defined in (Chapelle et al. 2009) as the expectation of ERR over the different intents.

Another metric of interest for diversity has been proposed in (Clarke et al. 2008) and is named α-NDCG. It is defined for binary relevance and depends on a parameter $\alpha \in [0, 1]$. For simplicity, we define here the unnormalized version, α-DCG:

$$\sum_k \frac{1}{\log_2(k+1)} \sum_i r_k^i (1-\alpha)^{s_{i,k-1}} \quad \text{with} \quad s_{i,k-1} := \sum_{j=1}^{k-1} r_j^i. \tag{4}$$

α-DCG can be rewritten as:

$$\sum_i \sum_k \frac{1}{\log_2(k+1)} r_k^i \prod_{j=1}^{k-1} (1-\alpha)^{r_j^i} = \frac{1}{\alpha} \sum_i \sum_k \varphi(k) \mathcal{R}(r_k^i) \prod_{j=1}^{k-1} (1 - \mathcal{R}(r_j^i)), \tag{5}$$

with $\varphi(k) = 1/\log_2(k+1)$ and $\mathcal{R}(r) = \alpha r$.

Note that there is a semantic difference between the *intents* of an ambiguous query (Agrawal et al. 2009) and the information *nuggets* for an underspecified query (Clarke et al. 2008). In fact, each intent of a query can have several relevant nuggets and metrics for diversity should thus have a double summation over both intents and nuggets as noted by Clarke et al. (2009). Equations (3) and (5) can formally handle that scenario by letting the index i goes over (intent, nugget) pairs. In the rest of this paper, an intent should thus be understood as an (intent, nugget) pair.

The expression of α-DCG as equation (5) leads us to our first result:

Proposition 1 *The α-DCG metric* (Clarke et al. 2008) *is, up to a constant, an instantiation of an intent aware cascade based metric (CBM-IA)* (*see equation* 2) *in which all the intents are supposed to be of equal importance.*

In fact, the authors of (Clarke et al. 2008) made the simplifying assumption that all intents are equally probable (with a probability γ in their paper), but it is not too difficult to generalize their derivation to the case where each intent has a probability p_i. The resulting metric would be the same as in (5) but with a p_i inside the sum.

4.3 Properties of intent-aware metrics

We argue in this section that DCG-IA and AP-IA are not well suited metrics for diversity because they do not particularly reward rankings covering various intents. On the contrary, CBM-IA does. Key to this analysis is the notion of *submodularity* that we review below.

In this section, we view an information retrieval metric as a function of the relevance of the documents. For instance, if we have n documents, we can write AP as a function defined on $\{0,1\}^n$: $f(x_1, \ldots, x_n)$.

4.3.1 Submodularity

This is a very short introduction to submodular functions. More details can be found in (Simchi-Levi et al. 2005, Chap. 2).

Definition 1 A function f is *submodular* if for any two elements x and y for which f is defined, we have:

$$f(x) + f(y) \geq f(x \vee y) + f(x \wedge y), \tag{6}$$

where $x \vee y$ (resp. $x \wedge y$) denotes the componentwise maximum (resp. minimum) of x and y.

If $-f$ is submodular, f is said to be *supermodular*. If f is both supermodular and submodular, it is said to be *modular*. The following proposition gives two properties of submodular functions.

Proposition 2 *When f is twice differentiable, the following 3 assertions are equivalent:*

1. *f is submodular*
2. $\forall i \neq j$, define $\psi_{ij}(t,u) := f(x_1,\ldots,x_{i-1},t,x_{i+1},\ldots,x_{j-1},u,x_{j+1},\ldots)$. *Then* $\forall a<b$, $t \rightarrow \psi_{ij}(t,a) - \psi_{ij}(t,b)$ *is a non-decreasing function.*
3. $\forall i \neq j$, $\frac{\partial^2 f}{\partial x_i \partial x_j} \leq 0$.

4.3.2 Submodularity for IR metrics

Let us illustrate submodularity in the context of metrics. Let f be a submodular metric, r a vector of binary relevance labels, i and j two rank positions and let us define r^{ab} the relevance vector r where r_i is replaced by a and r_j by b.

Let us apply the definition of a submodular function with $x = r^{01}$ and $y = r^{10}$:

$$f(r^{01}) + f(r^{10}) \geq f(r^{11}) + f(r^{00}).$$

This can be rewritten as:

$$f(r^{10}) - f(r^{00}) \geq f(r^{11}) - f(r^{01}).$$

It means that the added value in switching the relevance of the i-th document from non-relevant to relevant is smaller when the j-th document is relevant than when it is not. This is a *diminishing return* property which seems highly desirable for most IR tasks: if we have already shown a lot of relevant documents, there should be less added value in showing more relevant documents.

Proposition 3 *The metrics described in Sect.* 4.1 *have the following properties: CBM is a submodular; DCG is modular; and AP is supermodular.*

The proof is in the appendix and relies on the 3rd assertion of proposition 2. Note that the properties of proposition 3 extend to the intent-aware versions of the metric since a positive linear combination of submodular (resp. supermodular) functions is also submodular (resp. supermodular).

4.3.3 Consequences for intent aware metrics

The submodularity of CBM implies that CBM-IA (of which ERR-IA and α-DCG are instantiations) will favor rankings covering diverse intents. Indeed, once several relevant documents have already been selected for an intent, there is little gain in adding more relevant documents from that topic (diminishing return property). It is then more beneficial to add more relevant documents from another topic.

On the other hand, there is no diminishing return property for DCG. In fact it is easy to see that, by swapping the order of the summations in (3), the optimal ranking under DCG-IA is simply obtained by sorting the documents according to their expected gain, $\sum p_i \mathcal{R}(r_j^i)$. As

Table 2 Synthetic example with 9 documents and 3 intents

Intent	A	B	C
Prob.	0.4	0.3	0.3
d1	✔		
d2	✔		
d3	✔		
d4		✔	
d5		✔	
d6		✔	
d7			✔
d8			✔
d9			✔

List 1	List 2	
d1	d1	
d2	d4	
d3	d7	
5.97	5.17	DCG-IA
0.243	0.284	ERR-IA

The labels are either Excellent (tick) or Bad. The first rank list is not diverse (only covering intent A) but is preferred by DCG-IA, while the second list returns one relevant document per intent and is preferred by ERR-IA

a result, the choice of a document in a position p does not influence the choice of another document in position $q > p$. This is not desirable from a diversity point of view. A comparison between DCG-IA and ERR-IA on two rankings is illustrated in Table 2.

AP-IA is, in some sense, even worse than DCG-IA because AP is supermodular. This metric will favor rankings where all the relevant documents come from the same intent. The proposition below is a proof of this fact in a special case.

Proposition 4 *Assume that we want to retrieve n documents and that each document is relevant to at most one intent. Then, if there are at least n relevant documents for the most likely intent, the optimal ranking under AP-IA is obtained by selecting n relevant documents from that intent.*

4.4 Correlation with click logs

After this theoretical analysis, we now turn to an empirical validation to check whether ERR-IA can indeed better predict user behavior—as observed in click logs—than DCG-IA. For this purpose, we follow the same experimental protocol as the one outlined in Chapelle et al. (2009). As discussed later in Sect. 6, we have a set of 406 queries and about 50 documents per query. Each triplet (query, document, intent) has been editorially judged on a 5-point relevance scale. The intents and the method for estimating the intent distributions have been described in Sect. 3.

The sessions corresponding to these 406 queries have been extracted from the clicks logs of Yahoo! search over a 5-month period. We only kept the sessions for which the top 5 documents were editorially judged and only considered the clicks on one of the top 5 documents. The reason for not considering the top 10 positions is because the intersection between clicks logs and editorial judgments would be much smaller. As in (Chapelle et al. 2009), we call a *configuration* a given query and list of 5 documents. Because of variations in the search engine, a query may have several different configurations. In fact the total

Table 3 Correlation on 406 queries of ERR-IA and DCG-IA with the opposite of abandonment rate

ERR-IA	DCG-IA
0.326	0.276

number of configurations for these 406 queries is 9930. Each configuration has an average of 54 sessions. For each configuration, we compute two intent aware metrics, DCG-IA and ERR-IA, as well as UCTR which is the proportion of sessions for which there is at least one click. The reason we consider UCTR is that it is the opposite of the abandonment rate, which is a typical quantity of interest when optimizing for diversity (Clarke et al. 2008; Sarma et al. 2008; Agrawal et al. 2009). The correlations are shown in Table 3. The difference between ERR-IA and DCG-IA is statistically significant according to a bootstrap test (p value $= 2 \times 10^{-7}$).

5 Algorithms for diversity

5.1 Objective function

As explained in the previous section, ERR-IA is defined as:

$$\sum_i p_i \sum_k \frac{s_{ik}}{k} \prod_{j<k} (1 - s_{ij}), \tag{7}$$

with $s_{ik} := \mathcal{R}(r_k^i)$ and r_k^i the relevance label of the k-th document with respect to the i-th intent. This is the expected reciprocal rank at which the user will stop its search under the cascade model, where the expectation is taken over the intents and the users.

But instead of using (7) for evaluation, it can be *optimized* to retrieve a set of diverse and relevant results. The relevance judgments are of course not available for most queries, so the s_{ik} are to be estimated from the intent based relevance functions presented in Sect. 3.

The objective function (7) is related to the one used in (Agrawal et al. 2009). Indeed, if the $1/k$ decay factor is ignored, the expected reciprocal rank turns out to be the probability of a clicking on any result, that is 1 - the probability of abandonment. Mathematically, this can be seen with:

$$\sum_i p_i \sum_k s_{ik} \prod_{j<k} (1 - s_{ij}) = \sum_i p_i \left[1 - \prod_k (1 - s_{ik}) \right]. \tag{8}$$

The right hand side of this equation is indeed the quantity that is optimized in (Agrawal et al. 2009). But note that it is independent of the order to the documents; it only depends on the *set* of documents. On the other hand, the objective function (7) is better suited for ranking purposes as its value depends on the order of the documents.

5.2 Greedy optimization

The objective function (8) has been proved to be submodular.[3] This implies that a greedy algorithm for optimizing it is not too far from the optimal solution. Submodularity is

[3] The notion of submodularity here is slightly different than the one presented in Sect. 4.3.1: this one is defined on sets.

Algorithm 2 Greedy optimization of (7)

```
Require: $s_{ik}$, probabilities of satisfaction for all intents $i$ and all the documents $k$.
      $p_i$ probability of intent $i$.
$R = \emptyset$                                        Documents selected in the rank list
for $j = 1, \ldots, 10$ do                             Loop over the positions
    $k^* \leftarrow \arg\max_{k \notin R} \sum_i p_i s_{ik}$           Greedy selection
    $R \leftarrow R \cup k^*$
    $p_i \leftarrow p_i(1 - s_{ik^*})$                  Posterior probabilities
end for
```

defined in terms of sets and that is indeed how the objective function of (Agrawal et al. 2009) is defined. The situation is a bit more complex in our case because our objective function is not defined on sets, but on ordered lists. However by considering an ordered list as a set of pairs (document, rank), we can also prove that the objective function 7 is submodular. We are not including here a proof because it is out of the scope of this paper. The bottom line is that the submodularity implies a theoretical guarantee about the greedy algorithm presented below: the objective function value that it will reach will never be less than half of the optimal one. The greedy optimization algorithm turns out to be the same as the IA-SELECT algorithm of (Agrawal et al. 2009) and is described in Algorithm 2.

5.3 Branch-and-bound for exact optimization

As already discussed in (Agrawal et al. 2009), optimizing (7) is NP hard. Even though one can use submodularity to prove theoretical guarantee of the greedy Algorithm 2, a natural question arises: how far, in practice, is this greedy solution from the optimal one?

To answer this question, we devised a branch-and-bound algorithm to find the global optimal solution of (7). Branch and bound algorithms are methods for global optimization in combinatorial problems (Papadimitriou and Steiglitz 1998). It generally involves the following two steps:

Branching This consists in growing a tree in which each leaf corresponds to a possible solution and an inner node represents a set of solutions. In our case, the tree has depth $N + 1$ (N is the number of documents) and a node at the j-th level corresponds to an assignment from positions to 1 to j.

Bounding This is a procedure that computes upper and lower bounds for the value of the objective function within a given subset.

The key idea of a branch-and-bound algorithm is: if the upper bound for some tree node (set of candidates) A is lower than the lower bound for some other node B, then A may be safely discarded from the search. This step is called pruning and reduces the number of possible branches. We made the following design choices in our implementation:

- We use a depth first search algorithm and the children of a node are explored according to the greedy heuristic of Algorithm 2. In particular, this means that the first leaf to to be reached corresponds to the solution of the greedy algorithm.
- The lower bound is the best objective function value so far and the upper bound is explained below.

The maximum of the objective function (7) over a set of permutations $\pi \in \mathcal{P}$ can be upper bounded as:

$$\max\nolimits_{\pi\in\mathcal{P}} \sum_i p_i \sum_k \frac{s_{i,\pi(k)}}{k} \prod_{j<k} (1 - s_{i,\pi(j)}) \leq \sum_i p_i \max\nolimits_{\pi\in\mathcal{P}} \sum_k \frac{s_{i,\pi(k)}}{k} \prod_{j<k} (1 - s_{i,\pi(j)}), \qquad (9)$$

where $\pi(k)$ corresponds to the index of the k-th document in the ranking induced by π. For a given node in the tree, the first documents in the ranking are already chosen, so we use the above upper bound where the set of permutations $\mathcal{P}$ is restricted to those correctly placing these documents. An intuitive explanation of this upper bound is the following: for a given intent, maximizing ERR is straightforward because this only involves sorting according to the scores s_{ik}; the upper bound is simply the weighted average of these maximum ERR values.

Evaluation of the greedy algorithm In order to quantify the sub-optimality (if any) of the greedy Algorithm 2, we took a set of 50 queries from the dataset to be described in Sect. 6 and diversified the search results using the greedy algorithm and the branch-and-bound algorithm.

The differences in objective function values obtained by these algorithms are reported in Table 4. The greedy algorithm finds the optimal solution for most queries and when it does not find it, the gap is very small. This is very reassuring since the branch-and-bound algorithm is too slow in practice; instead, the near-optimal greedy algorithm can safely be used (at least on this dataset).

5.4 Learning the transfer function

A very important component of our algorithm is a reliable estimation of s_{ik}, the satisfaction probability of the k-th document for the i-th intent. This crucial aspect has not been addressed in (Agrawal et al. 2009). As explained in Sect. 3, we have at our disposal a scoring function for each intent and we can thus get a relevance score t_{ik} for each document and each intent. Since these scoring functions have been trained by regression with targets between 0 and 10, a first heuristic which comes in mind is to rescale these scores between 0 and 1:

$$s_{ik} := \frac{t_{ik}}{10}. \qquad (10)$$

But as will see in the next section, this heuristic yields only a small improvement.

Instead we propose to learn a transfer function ϕ_i such that:

$$s_{ik} := \phi_i(t_{ik}) \qquad (11)$$

Table 4 Relative difference, over 50 queries, in the objective function value (7) between the solution found by the greedy algorithm 2 and the optimal one found by the branch-and-bound algorithm

Relative difference	Number of queries
None	39
10^{-5} to 10^{-4}	6
10^{-4} to 10^{-3}	5

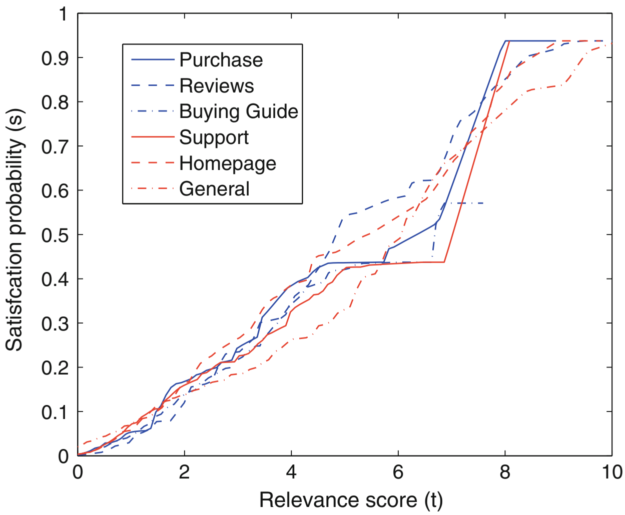

Fig. 2 The transfer functions ϕ_i (11) learned with isotonic regression

For this purpose, we used, for each intent, an independent set of editorial judgments and computed for each document its score t according to the intent based relevance function as well as its satisfaction probability, as estimated with $\mathcal{R}(r)$, r being the editorial judgment and $\mathcal{R}$ the function defined in (1). These input-outputs pairs are then fed to an isotonic regression algorithm (Barlow et al. 1972) which produces a non-parametric function minimizing the mean squared error under a monotonicity constraint. The reason for having this constraint is that we expect the probability of satisfaction to be an increasing function of the relevance score.

The learned transfer functions ϕ_i are plotted in Fig. 2.

6 Evaluation

In this section, we first evaluate the intent mining method described in Sect. 3.2, then the quality of intent based relevance functions of Sect. 3.3 and we finally assess the diversification performance of Algorithm 2.

6.1 Intent mining

To evaluate our intent mining method, we randomly sampled 2,492 user sessions from the logs of Yahoo! search from December, 2008 to March, 2009. There are 3,658 queries and 18,296 documents in this data set. For each user session, a professional editor examined the user activity in that session and judged the user intents. There were about thirty editors involved in the intent judgment process. We minimized the inter-editor disagreements by having pilot judgment runs, workshops, and clear editorial guidelines. Editors were instructed to examine the whole search session that includes user's queries, clicks, and seen search result links before they made any intent judgment. As explained in Sect. 3.2, part of this dataset (70%) has been used for our per-intent score threshold determination. The remaining 30% of the queries were used as a test set for our query intent classifier. Table 5

Table 5 Precision and recall of each query intent classifier

Intent	Precision (%)	Recall (%)
Homepage	84.1	52.6
Reviews	78.7	32
Purchase	71.7	29
Support	44	10.6
Buying guide	0.8	25

shows the precision and recall of our query intent classifiers. For homepage, review, shopping, we have good classification performance. However, for support and buying guide intent, their classifier performances are not satisfactory. This is because the support intent itself covers wide variety of sub-intents such as “product recall”, “product manual”, “product warranty”, “product repair”, “tutorial” etc. The support intent classifier could not cover all those sub-intents. Even though the support classifier does not have high accuracy, the intent-specific ranking function still improve relevancy drastically as shown in the next section. For buying guide intent, it has low performance due to data scarcity. Only very few users had buying guide intents and issued a query for that. Furthermore, many product categories do not have any good buying guide pages available over the web except a few popular product categories. Even though the user would submit a query for a buying guide, there is little chance the search engine would find a good page for that.

We also asked the editors the relevance of each clicked page with respect to the determined intent. The goal was to evaluate wether *post-search* clicks—mined from the toolbar logs—are more informative or not than the clicks from the *search* page. Among the sessions for which there was at least one click relevant to the determined intent, 49.8% had a post-search relevant click, 34% had a relevant click on the search page and 16.2% had both. This confirms that post-search clicks are very helpful in determining the user’s intent.

6.2 Intent-specific ranking

We also evaluated the performance of the intent-specifc learned ranking functions by comparing them to a general purpose ranking function that does not use intent-conditioned training labels. As explained in Sect. 3.3, these functions are trained using Gradient Boosted Decision Trees (GBDT) (Friedman 2001) on a set of thousands of editorially labeled (query,urls) pairs. The relevance labels are on a 5 point scale, ranging from *Bad* to *Perfect*. For the comparison of each these functions, we used a held-out portion of its intent-conditioned training set. Table 6 shows the relevance gain ranges

Table 6 Relevance gain of the intent-specific ranking functions over the general purpose ranking function

Intent	DCG5 gain (%)
Homepage	15.5
Reviews	126.5
Purchase	37.0
Support	104.2
Buying guide	22.6

from 15.5% to 126.5%. It is also interesting to observe that for a relatively rare intent category such as Support, the large relevance difference indicates that the general purpose ranking function fails to return good documents for that intent (i.e. it strongly underestimates the relevance of these documents). This is exactly why the approximation in (Agrawal et al. 2009) that we mentioned in Sect. 3.3 is poor. Another interesting observation is that, our method can also underestimate the relevance of good documents when there are too few examples to learn from, as in the case of Buying Guide category.

6.3 Diversification

To assess the effectiveness of our IA-DIV (Intent aware diversification) algorithm, we randomly selected 406 shopping related queries. For each query, the 6 intent based ranking functions are used to obtain the top 10 results retrieved from the index of Yahoo! search; after aggregating, each query has about 50 unique documents. Then each triplet (query, document, intent) was editorially judged on a 5 points relevance scale.

The baseline "undiversified" ranking list is obtained by sorting the documents according to the standard relevance function; and the diversified ranking list is computed via the greedy Algorithm 2 that we call IA-DIV. The corresponding ERR-IA of each ranking list is computed using the editorial judgments. The same query intent distribution is used for optimization and evaluation. We have tried two different transfer functions: the "linear" one (10) and the "monotonic" one (11) learned with isotonic regression. Table 7 reports the performance of different methods.

The difference between IA-DIV-monotonic and the undiversified baseline is statistically significant: the p value of a paired t-test is 0.013 and the one of Wilcoxon test is 1.4×10^{-4}. Even though the DCG gains in table 6 of the intent-specific ranking functions are very large, the gain of the rankings produced by IA-DIV is not as large. This could be due to the fact that for a lot of queries, the probability of the "general" intent has been estimated to be more than 50% and the ranking produced by our algorithm for these queries is then similar to the original ranking.

Finally we also compared to the MMR algorithm (Carbonell and Goldstein, 1998) where the similarity measure is the cosine similarity in the vector space model. It resulted in a drop in performance which can be explained by the fact that this similarity measure is query and intent independent.

Table 7 Values of the ERR-IA editorial metric for different reranking strategies

	Normalized ERR-IA	Improvement (%)
Undiversified	0.8179	–
IA-DIV-linear	0.8238	0.7
IA-DIV-monotonic	0.8325	1.8
MMR	0.8005	−2.1

The two versions of IA-DIV differ by the choice of the transfer function. The scores have been normalized such that the ideal ranking has a score of 1

7 Future work

The diversification scheme proposed in this paper is powerful, but is limited to only a certain class of queries because it requires to explicitly model the relevance of a document with respect to a given intent. As future work, we propose an extension of the proposed algorithms which can be used to diversify the result set of *any* query and does not require intent relevance modeling.

In this proposal, diversification arises from document content analysis which can be done through the *machine learned aboutness* (MLA) algorithm introduced in (Paranjpe 2009). This algorithm extracts from a document the terms which are supposed to be salient. In addition each term is given a score of how "important" it is in the document. More precisely, this score is an estimate of the click probability on the document for a user issuing a query containing that term. This score can thus be thought as a relevance probability conditioned on that term.

We now propose to consider each of these terms as an intent and apply the same diversification algorithm as above. More precisely, we maximize the objective function (7), where s_{ik} is the score given by the MLA algorithm to the i-th term for the k-th document. Instead of considering the set of all possible terms, we can restrict ourselves to the ones that appear at least once in a short list of documents to be reranked.

Let p_i the probability that the user is interested in the i-th term. A crucial step is to infer these probabilities. Indeed, remember that the terms and scores are extracted in a query independent manner; so we now need to know, given a query, how likely is a user to be interested in a given term. For this purpose, we can make use of an estimate of the marginal relevance.

The marginal probability that a user is satisfied by the k-th document is:

$$P(S_k = 1|q) = \sum_i P(S_k = 1|i, q)P(i|q) = \sum_i s_{ik}p_i. \tag{12}$$

The left hand side of equation (12) can either be estimated with standard relevance judgments if they are available– for instance like in ERR, $P(s_k = 1|q) = \mathcal{R}(r_k)$, if document k has relevance label r_k; or through a relevance score $\phi(t_k)$ like in Sect. 5.4.

We thus have a system of linear equations (12) where the unknown are p_i. Once these p_i are found, we can use the same diversification algorithm as in Sect. 5.

8 Conclusion

We have explored in this paper the diversification of search results based on intent modeling. This work can be seen as a follow-up of (Agrawal et al. 2009) with the following contributions. First, we studied metrics for diversity and showed that a diminishing return property is needed in order to favor the coverage of various intents. Second, by comparison with a branch-and-bound algorithm that we devised, we showed that, on our dataset, the greedy algorithm for diversification is near optimal. Third, we explicitly modeled the intent relevance scores by training dedicated intent-specific ranking functions. And finally, we proposed a method to map the relevance scores to probabilities of satisfaction.

Empirically we have first shown that ERR-IA correlates better with user satisfaction—as measured by abandonment rate—than DCG-IA. Then we showed that our framework for diversification leads to better results in terms of ERR-IA. Future work, as explained in the previous section, includes extension of the proposed algorithm to the case where intent based relevant judgments are not explicitly available.

Appendix

Proof of proposition 3 We will use the 3rd property of proposition 2.

For AP,

$$f_{AP}(x_1,\ldots,x_n) = \frac{1}{R}\sum_k x_k \frac{1}{k}\sum_{j\leq k} x_j$$

and $\frac{\partial^2 f_{AP}}{\partial x_i \partial x_j} > 0,\ \forall i \neq j$. Note that we use here the assumption that R is constant.

For DCG, we have:

$$f_{DCG}(x_1,\ldots,x_n) = \sum_{j=1}^{n} \frac{\mathcal{R}(x_j)}{\log_2(j+1)}$$

and $\frac{\partial^2 f_{DCG}}{\partial x_i \partial x_j} = 0$.

For CBM, the proof is a bit more involved, but it is clear intuitively that CBM has a diminishing return property: if a relevant document is placed in position i, the probability of examination in positions $j > i$ will be low and the added value of placing another relevant document in position j is lower than if the the document in position i were not relevant. □

We first need the following lemma:

Lemma 1 $\forall x_i \in [0,1], \forall k \leq n$, *we have:*

$$\sum_{i=k}^{n} x_i \prod_{j=k}^{i-1}(1-x_j) \leq 1.$$

The lemma can easily be proved by induction: it is true for $k = n$ and if it true for a given k, then it also true for $k - 1$:

$$\begin{aligned}
&\sum_{i=k-1}^{n} x_i \prod_{j=k-1}^{i-1}(1-x_j) \\
&\quad = x_{k-1} + (1-x_{k-1})\sum_{i=k}^{n} x_i \prod_{j=k}^{i-1}(1-x_j) \\
&\quad \leq x_{k-1} + (1-x_{k-1}) = 1.
\end{aligned}$$

We have $f_{CBM}(x_1,\ldots,x_n) = \sum_k \varphi(k) x_k \prod_{j=1}^{k-1}(1-x_j)$. We ignore for simplicity the function $\mathcal{R}$: since it is a monotonic function, it will not affect the sign of the second derivatives. Let us fix two ranks $p < q$:

$$\begin{aligned}
\frac{\partial f_{CBM}}{\partial x_p \partial x_q} &= -\varphi(q)\prod_{i=p+1}^{q-1}(1-x_i) + \sum_{j=q+1}^{n}\varphi(j)x_j \prod_{i=p+1,\, i\neq q}^{j-1}(1-x_i). \\
&= \prod_{i=p+1}^{q-1}(1-x_i)\left[-\varphi(q) + \sum_{j=q+1}^{n}\varphi(j)x_j \prod_{i=q+1}^{j-1}(1-x_i)\right]
\end{aligned}$$

$$\le \prod_{i=p+1}^{q-1}(1-x_i)\varphi(q)\left[-1+\sum_{j=q+1}^{n}x_j\prod_{i=q+1}^{j-1}(1-x_i)\right]$$
$$\le 0.$$

The first inequality hold because φ is decreasing while the second comes from applying the lemma with $k = q + 1$.

Proof of proposition 4 Let us assume, without any loss of generality, that the first topic is the most likely one. Then, let us consider the ranking consisting of n relevant documents for that topic. The value for AP will be 1 for that topic and 0 for the other topics (because we assumed that a document is relevant to at most one topic). And thus the value of AP-IA is p_1. We will now show that the value of any other ranking is less or equal to p_1.

Let r_j^i be the relevance values for an arbitrary ranking and let r^i denote the vector $(r_1^i, \ldots, r_n^i)$. Because f_{AP} is supermodular, we can apply several times the reverse inequality from the definition (6) and get:

$$\sum_i f_{AP}(r^i) \le f\left(\bigvee_{i=1}^{n} r^i\right) + \sum_{j=2}^{n} f\left(\left(\bigvee_{i=1}^{j-1} r^i\right)\bigwedge r^j\right).$$

Because of the assumption that a document is at most relevant to one topic, all the terms containing a $\wedge$ in the above equation are 0. And we finally obtain:

$$\sum p_i f_{AP}(r^i) \le p_1 \sum_i f_{AP}(r^i) \le p_1 f\left(\bigvee_{i=1}^{n} r^i\right) \le p_1.$$

□

References

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In *WSDM '09: Proceedings of the 2nd international conference on web search and web data mining* (pp. 5–14). ACM.

Barlow, R. E., Brunk, H. D., Bartholomew, D. J., & Bremner, J. M. (1972). *Statistical inference under order restrictions: The theory and application of isotonic regression*. New York: Wiley.

Burges, C. J., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., et al. (2005). Learning to rank using gradient descent. In *Proceedings of the international conference on machine learning*.

Carbonell, J., & Goldstein, J. (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* (pp. 335–336).

Carterette, B., & Chandar, P. (2009). Probabilistic models of ranking novel documents for faceted topic retrieval. In *CIKM '09: Proceeding of the 18th ACM conference on information and knowledge management*.

Chapelle, O., Metlzer, D., Zhang, Y., & Grinspan, P. (2009). Expected reciprocal rank for graded relevance. In *CIKM '09: Proceedings of the 18th ACM conference on information and knowledge management*.

Chen, H., & Karger, D. R. (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* (pp. 429–436).

Clarke, C., Kolla, M., Cormack, G., Vechtomova, O., Ashkan, A., & Büttcher, S., et al. (2008). Novelty and diversity in information retrieval evaluation. In *SIGIR '08: Proceedings of the 31st annual international conference on research and development in information retrieval* (pp. 659–666). ACM.

Clarke, C. L., Kolla, M., & Vechtomova, O. (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* (pp. 188–199).

Claypool, M., Le, P., Waseda, M., & Brown, D. (2001). Information filtering based on user behavior analysis and best match text retrieval. In *Proceedings of ACM intelligent user interfaces conference (IUI)* (pp. 33–40).

Craswell, N., Zoeter, O., Taylor, M., & Ramsey, B. (2008). An experimental comparison of click position-bias models. In *WSDM '08: proceedings of the international conference on web search and web data mining* (pp. 87–94). ACM.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics, 29*(5), 1189–1232.

Gollapudi, S., & Sharma, A. (2009). An axiomatic approach for result diversification. In *Proceedings of the 18th international conference on World Wide Web* (pp. 381–390).

Jarvelin, K., & Kekalainen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems, 20*(4), 422–446.

Liu, C., White, R. W., & Dumais, S. (2006). Understanding web browsing behaviors through weibull analysis of dwell time. In *Proceedings of the 33rd annual international ACM SIGIR conference on research and development in information retrieval.*

Papadimitriou, C. H., & Steiglitz, K. (1998). *Combinatorial optimization: Algorithms and complexity*. Mineola, NY: Dover.

Paranjpe, D. (2009). Learning document aboutness from implicit user feedback and document structure. In *CIKM '09: Proceeding of the 18th ACM conference on information and knowledge management* (pp. 365–374). ACM.

Radlinski, F., Kleinberg, R., & Joachims, T. (2008). Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th international conference on machine learning* (pp. 784–791).

Rafiei, D., Bharat, K., & Shukla, A. (2010). Diversifying web search results. In *Proceedings of the 19th international conference on World Wide Web* (pp. 781–790).

Santos, R. L. T., Macdonald, C., & Ounis, I. (2010a). Exploiting query reformulations for Web search result diversification. In *Proceedings of the 19th international conference on World Wide Web* (pp. 881–890).

Santos, R. L. T., Peng, J., Macdonald, C., & Ounis, I. (2010b). Explicit search result diversification through sub-queries. In *Proceedings of the 31st European conference on information retrieval* (pp. 87–99).

Sarma, A. D., Gollapudi, S., & Leong, S. (2008). Bypass rates: Reducing query abandonment using negative inferences. In *Proceeding of the 14th ACM SIGKDD international conference on knowledge discovery and data mining (KDD)* (pp. 177–185).

Simchi-Levi, D., Chen, X., & Bramel, J. (2005). *The logic of logistics: theory, algorithms, and applications for logistics and supply chain management*. Berlin: Springer.

Wang, J., & Zhu, J. (2009). Portfolio theory of information retrieval. In *Proceedings of the 32th annual international ACM SIGIR conference on research and development in informaion retrieval* (pp. 115–122).

Zhai, C. X., Cohen, W. W., & Lafferty, J. (2003). Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th snnual international ACM SIGIR conference on research and development in informaion retrieval* (pp. 10–17).

Zheng, Z., Zha, H., Zhang, T., Chapelle, O., Chen, K., & Sun, G. (2008). A general boosting method and its application to learning ranking functions for web search. In J. C. Platt, D. Koller, Y. Singer, & S. Roweis (Eds.), *Advances in Neural Information Processing Systems, 20*, 1697–1704.