



Dynamic Information Retrieval: Theoretical Framework and Application

Marc Sloan and Jun Wang
Department of Computer Science
University College London

M.Sloan@cs.ucl.ac.uk, J.Wang@cs.ucl.ac.uk

ABSTRACT

Theoretical frameworks like the *Probability Ranking Principle* and its more recent *Interactive Information Retrieval* variant have guided the development of ranking and retrieval algorithms for decades, yet they are not capable of helping us model problems in *Dynamic Information Retrieval* which exhibit the following three properties; an observable user signal, retrieval over multiple stages and an overall search intent. In this paper a new theoretical framework for retrieval in these scenarios is proposed. We derive a general dynamic utility function for optimizing over these types of tasks, that takes into account the utility of each stage and the probability of observing user feedback. We apply our framework to experiments over TREC data in the dynamic multi page search scenario as a practical demonstration of its effectiveness and to frame the discussion of its use, its limitations and to compare it against the existing frameworks.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval]: Relevance feedback; Retrieval Models; Search process;

Keywords

Dynamic IR, Interactive IR, Ranking and Retrieval Theory

1. INTRODUCTION

The theoretical frameworks that underpin research in Information Retrieval (IR) are based on abstract models of user benefit. For instance, the loss function defined in the classic *Probability Ranking Principle* (PRP) [26] leads to justification for the simplest and most powerful ranking rule in IR; ranking documents in decreasing order of their probability of relevance. A recent counterpart is the *Probability Ranking Principle for Interactive Information Retrieval* (IIR-PRP) [12], which relaxes the independence assumption in the PRP's model to take into account non-linear decision making. For example, document dependence is a key element in IR diversification that is not handled by the PRP [32]. These models deal with traditional ad hoc query ranking and retrieval. Yet search tasks are complex

and often exploratory, being comprised of multiple stages of retrieval with information needs specialized or generalized over time [35]. Throughout, a user may broadcast signals of search intent that can help an IR system to improve retrieval. Examples include query reformulation in session search [29] and item ratings in collaborative filtering [15]. The described models are not capable of representing search tasks that operate over multiple stages nor can they incorporate user feedback.

These types of problems belong to the area of IR research known as *Dynamic Information Retrieval* (DIR), which we define as exhibiting three characteristics: user feedback, temporal dependency and an overall goal. In this paper we present DIR as a natural progression in IR research complexity; where early research concerned *static* problems such as ad hoc retrieval, which gave way to *interactive* tasks such as those incorporating relevance feedback [27], finally leading to *dynamic* systems where tasks such as ranking for session search are optimized [24].

From this progression we mathematically formulate a generalized framework that models the expected benefit to a user of completing a DIR task. This benefit is represented as a recursive utility function that is goal oriented and adaptive over time. The components of this utility function represent the three DIR features: the likelihood of user feedback, a probability of relevance model conditioned on this feedback and an individual stage utility function. The optimization of this recursive utility leads to an optimal policy of actions dependent on user interactions in the dynamic setting.

This utility is shown to be a form of Bellman equation [3], the framework an instantiation of a *Partially Observable Markov Decision Process* (POMDP) [31] and also a generalization over existing research in DIR [24, 19]. The components of the utility can also be linked to the cost-benefit parameters of the IIR-PRP and the discount-gain functions found in session-based metrics such as *sDCG* [21]. The structure of the utility function and its links to these areas of research give us interesting insights into the behavior of the function in DIR problems. These insights, such as how the quality and diversification of rankings vary over multiple stages, are supported by our experiments performed using a specific application of the DIR utility function over TREC data. Our experimental setting is the multi-page search scenario of choosing optimal rankings to display over several search pages for a fixed query [22, 19], a simplified DIR problem. As well as being a demonstration of the implementation of each of the components that make up the utility,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

ICTIR '15, September 27–30, Northampton, MA, USA.

© 2015 ACM. ISBN 978-1-4503-3833-2/15/09 ...\$15.00.

DOI: <http://dx.doi.org/10.1145/2808194.2809457>.

practical aspects such as the computational complexity are also explored.

Thus, through its supporting theory (Section 2) and application (Section 4), we establish our dynamic utility function (Section 3) as a new theoretical framework for the modeling of dynamic information retrieval problems.

2. COMPARISON OF IR FRAMEWORKS

Before setting up our framework for dynamic information retrieval, we consider *DIR* in the context of existing static and interactive theoretical IR frameworks in order to mathematically identify those features that distinguish it.

2.1 Static IR Framework

Definition: A static IR framework is one which models single user interactions, or else multiple *independent* interactions of different search intents. A typical application would be an ad hoc ranking and retrieval system.

The objective of a static system is to choose an **action** a (or sequence of actions $\vec{a} = \langle a_1, a_2, \dots \rangle$), each of which has an associated **probability of relevance** r (or \vec{r} for a sequence of actions) that maximizes some **static utility function** $U_S(a, r)$. The action represents a choice that can be made by the system and belongs to some action space \mathcal{A} . For example, a may be a query suggestion to display to a user, or \vec{a} the ranking order of a set of documents for retrieval. The utility function gives value to the action based on its probability of relevance by modeling the benefit of the action to the user. Utilities such as expected DCG and MAP [33] are examples from document retrieval, rewarding the ranking of relevant documents at high ranking positions.

2.1.1 Probability Ranking Principle (PRP)

The *PRP* defines $U_S(a, r)$ as a loss minimizing function across pairs of documents, which is optimized when ranking documents in decreasing order of probability of relevance (under document independence assumptions) [26]. Nonetheless, in instances where result diversity is important, it can be shown that *PRP* is no longer optimal [32]. We illustrate this with our example in Fig. 1a. Here, we represent a simplified vector space model for ranking and retrieval using a graph over two term axes. In this case, the query is **apple**, an ambiguous term that can describe three subtopic search intents. Those documents within the ranking \vec{a}_{PRP} for the query are retrieved (analogous to ranking under the *PRP*), and as we can see, in this case only two subtopic preferences are captured. Over a population of users, those seeking information on the **apple logo** subtopic would be dissatisfied.

We can capture this probabilistically by supposing that we have two classes of users, **user₁** and **user₂**, where **user₁** has twice as many members as **user₂**. Users in the **user₁** class are satisfied with the **apple logo** and **apple computer** subtopics, but not **apple fruit**, while those in the **user₂** class are only satisfied with the **apple fruit** subtopic. Our action space here is the set of subtopics which we denote $\{a_1 = \text{apple logo}, a_2 = \text{apple computer}, a_3 = \text{apple fruit}\}$ and our goal is to choose the best ranking of subtopics.

If we set $R_{a_k} = 1$ if a_k is relevant, and $r_{a_k} = P(R_{a_k} = 1)$, then we have $r_{a_1} = \frac{2}{3}$, $r_{a_2} = \frac{2}{3}$ and $r_{a_3} = \frac{1}{3}$. According to the *PRP*, we should rank in decreasing order of the probability of relevance, giving us the ranking sequence $\vec{a}_{PRP} = \langle a_1, a_2, a_3 \rangle$. However, intuitively this is not optimal because users belonging to **user₂** have to reject two subtopics before

reaching their preference [8]. We can explain this mathematically by studying the optimization of the diversity-encouraging metric Expected Search Length. One can also derive the same conclusion analogously using the equivalent Expected Reciprocal Rank or k -call at n measures [5]. In this scenario, $U_S(\vec{a}, \vec{r}) = E[L]_{\vec{a}}$ which is the summation of all possible search lengths L weighted by their respective probabilities, given as

$$E[L]_{\vec{a}} = \sum_i ((i-1)P(R_1 = 0, \dots, R_{i-1} = 0, R_i = 1))$$

where R_i is the relevance of the subtopic at rank position i . When assuming subtopics are independent, i.e. $P(R_1 = 0, \dots, R_i = 1) = P(R_1 = 0) \dots P(R_{i-1} = 0)P(R_i = 1)$ the expected search length for ranking \vec{a}_{PRP} is

$$\begin{aligned} E[L]_{\vec{a}_{PRP}} &= 0 \cdot r_{a_1} + 1 \cdot r_{a_2}(1-r_{a_1}) + 2 \cdot r_{a_3}(1-r_{a_2})(1-r_{a_1}) \\ &= 0 \cdot (2/3) + 1 \cdot (2/3)(1/3) + 2 \cdot (1/3)(1/3)(1/3) = 8/27 \end{aligned}$$

and for a diversified ranking $\vec{a}_{DIV} = \langle a_1, a_3, a_2 \rangle$ the expected search length is

$$E[L]_{\vec{a}_{DIV}} = 0 \cdot (2/3) + 1 \cdot (1/3)(1/3) + 2 \cdot (2/3)^2(1/3) = 11/27$$

Thus, in this case the *PRP* ranked documents have a shorter expected search path than the diversified ranking. Here, the *PRP* does lead to the optimal ranking under the independence assumption, but when we remove it this is no longer the case. To see this, we recalculate the expected search length for \vec{a}_{PRP} and \vec{a}_{DIV} but this time without the independence assumption:

$$\begin{aligned} E[L]_{\vec{a}_{PRP}} &= 0 \cdot r_{a_1} + 1 \cdot P(R_{a_2} = 1, R_{a_1} = 0) \\ &\quad + 2 \cdot P(R_{a_3} = 1, R_{a_2} = 0, R_{a_1} = 0) \\ &= 0 \cdot (2/3) + 1 \cdot 0 + 2 \cdot (1/3) = 2/3 \\ E[L]_{\vec{a}_{DIV}} &= 0 \cdot (2/3) + 1 \cdot (1/3) + 2 \cdot 0 = 1/3 \end{aligned}$$

Now we find that the diversified ranking \vec{a}_{DIV} has the shorter expected search length and is thus the optimal ranking, despite the lower probability of relevance for a_3 .

2.2 Interactive IR Framework

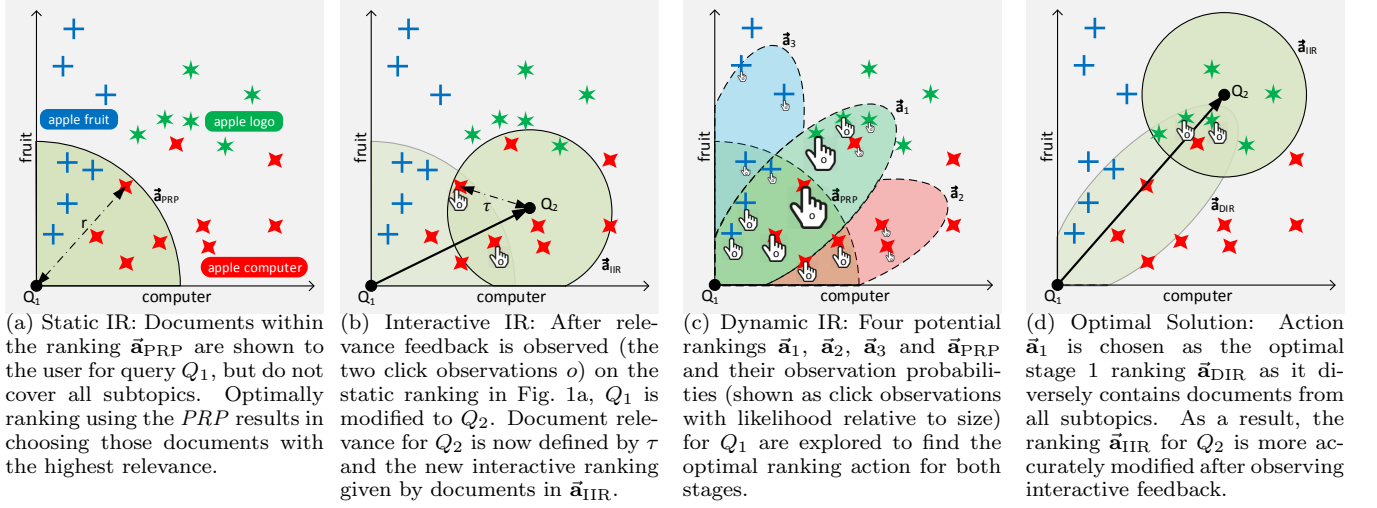
Definition: An interactive IR framework extends a static framework to cover multiple stages of IR. It is *responsive* to feedback from a previous stage but does not anticipate future feedback.

A **stage** represents an interaction with the search system that is distinct from other interactions but belongs to the same search task, for example a sequence of impressions in session search. Generally, an IR system will operate over $1 \leq t \leq T$ stages with T being potentially infinite.

Further to this, an interactive IR framework incorporates user feedback. Feedback is an **observation** signal o (or a sequence of observations \vec{o}) in the space \mathcal{O} , that is measurable by the search system. These signals may be *explicit* declarations of the relevance of search items (such as a movie rating), or *implicit* interpretations of user actions (such as document clicks).

The final element of this framework is the **relevance update function** τ where $r_{t+1} = \tau(a_t, r_t, o_t)$. Thus, the objective function for interactive IR at stage $t+1$ can now be represented as $\arg\max_{a_{t+1} \in \mathcal{A}} U_S(a_{t+1}, \tau(a_t, r_t, o_t))$. The relevance update function τ introduces temporal dependency into the framework, without it the objective simply devolves to optimization over the static utility $U_S(a, r)$. This is also

Figure 1: An example illustration of document ranking and relevance feedback using the vector space model for query $Q_1 = \text{apple}$. Documents are given as points over two term frequency axes, **computer** and **fruit**, and can belong to one of three subtopics **apple fruit**, **apple logo** and **apple computer**. The distance between Q and each document is inversely proportional to its relevance r . The documents ranked for Q_1 or its reformulation Q_2 are contained in each circular shape \bar{a} , whose area could be thought of as the static utility $U_S(\bar{a}, \bar{r})$, or U_D the combined area of actions across stages 1 and 2.



the case when finding the optimal first stage action a_1 i.e. when there are not yet any observations. In interactive IR, the optimal action is chosen at each stage as a reaction to the feedback observed in the previous stage and there is no consideration for future utility.

With these features in mind, we extend the vector space example to the interactive scenario in Fig. 1b by introducing the Rocchio relevance feedback algorithm [27] for interactively re-ranking documents. Here, clicked documents in the *PRP* ranked first stage are used as implicit signals of relevance to modify the user's original query Q_1 to Q_2 . Document re-retrieval occurs using Q_2 , returning documents using updated relevance scores given by τ , which is a function of Q_2 and thus the original ranking \bar{a}_{PRP} , document relevancies r and observations o . Nonetheless, even in this interactive framework, a user interested in the **apple logo** subtopic would be dissatisfied with both the \bar{a}_{PRP} and \bar{a}_{IIR} rankings due to a lack of documents for the relevant subtopic.

2.2.1 Interactive Information Retrieval (IIR)

For clarification, the area of research traditionally known as *Interactive Information Retrieval (IIR)* has an alternative definition to the interactive IR *framework* discussed in this paper, despite the similarity in name. *IIR* research explores the complex sequence of interactions a user may have with a search ranking within the static framework [28], largely motivated by the contradictory results found from conventional Cranfield style evaluation [7] and observational user studies [14]. With the exception of the *IIR-PRP* framework which we cover in more detail in Section 3.3.1, for the remainder of this paper any reference to interactive IR instead reflects the framework defined in this paper.

3. DYNAMIC IR THEORY

A dynamic system is one which is goal-directed and adaptive to its environment. From this definition we can specify three elements that determine whether an IR system is a dynamic one:

Feedback An observation signal from the user.

Temporal Dependency Operation across multiple stages where each stage depends on the previous stage.

Overall Goal An objective across all stages.

3.1 Dynamic IR Framework

Definition: A dynamic IR framework extends an interactive framework by being responsive to user feedback and optimizing for it in advance.

We previously defined systems in the interactive IR framework as exhibiting both feedback and temporal dependency features, but they are only capable of locally optimizing for a single stage at a time. In contrast, the optimization of a dynamic system will find the optimal sequence of actions for all future interactions. A result of this is that the utility of an individual stage may be reduced so that gains can be made in the utility at a future stage.

Unlike the interactive IR framework, the observation o is unknown when evaluating the utility of future stages in the dynamic IR framework. Instead, the *expected* utility can be found by marginalizing the utility function over the space of observations \mathcal{O} . When doing this the **observation likelihood function** $P(o|a, r)$ must be specified. This gives the expected utility

$$E[U_S(a, r)] = \sum_{o \in \mathcal{O}} P(o|a, r) U_S(a, r) \quad (1)$$

The observation likelihood function is represented visually in Fig. 1c. In dynamic IR, the expected utility of potential first stage rankings (given here as $\bar{a}_1, \bar{a}_2, \bar{a}_3$ and the optimal static ranking \bar{a}_{PRP}) are calculated by estimating which documents are likely to receive clicks and the effect this has on the utility of future stages. The *PRP* ranking is simply one among many rank actions that can be considered.

The final component of the *DIR* framework is the **path discount** function $\omega(t)$. When optimizing over a potentially infinite number of future stages, this helps ensure that a solution exists and also gives greater weight to earlier stage utilities.

By bringing together all of the components described so far, we can define the **utility function for dynamic information retrieval** as

$$U_D(r_t, t) = \max_{a_t \in \mathcal{A}} [U_S(a_t, r_t) + \omega(t) \sum_{o \in \mathcal{O}} P(o|a_t, r_t) U_D(\tau(a_t, r_t, o), t + 1)] \quad (2)$$

where $U_D(r_T, T) = \max_{a_T \in \mathcal{A}} [U_S(a_T, r_T)]$ is the static optimization of the final stage. Thus, our objective is to find, through backwards induction, the optimal sequence of actions $\bar{\mathbf{a}}^* = \langle a_1, \dots, a_T \rangle$ that maximizes the **dynamic utility** U_D given in Eq. (2). To derive this utility we have simply recursively applied the dynamic utility to the expected utility from Eq. (1).

Through the maximization of the dynamic utility, in our example in Fig. 1d we find the optimal action for stage 1, which is to diversify the initial ranking so that it retrieves documents belonging to all three subtopics. While this may harm the immediate retrieval utility score, overall the system improves because it can more accurately re-rank results over the subtopic preferences for all users in the next stage.

We observe that the eight elements of the *DIR* framework: a , r , U_S , t , o , τ , $P(o|a, r)$ and $\omega(t)$, are also the elements that define a *POMDP* [31], and that the dynamic utility function is its corresponding Bellman equation [3]. Intuitively this makes sense, like a *POMDP* the dynamic IR framework finds an optimal Markovian sequence of actions to maximize a reward (here the static utility) subject to discounting (with ω). The state of the system (the underlying document relevance) is unknown but a belief state (the probability of relevance) is updated according to observations. The key difference from a *POMDP* is that for dynamic IR we do not define a transition probability between states as we assume that the hidden relevance of each document does not change throughout the search task.

3.2 Framework Analysis

So far we have described the general framework for dynamic IR but have not addressed the setting of its parameters. Here we analyze each component within the context of dynamic information retrieval.

Relevance. As with any framework in information retrieval, the overall aim is to retrieve relevant information items and present them to the user. The intrinsic ‘relevance’ of an information item is an unknown quality and the subject of most of the research in IR. In the *DIR* framework any document relevance scoring method can be used. For instance, in our application in the next section we make use of five well established relevance scoring techniques.

The relevance update function τ is more difficult to define as it depends specifically on the action and observation space of the *DIR* task, for instance in Fig. 1b it depends on the distance of the documents from Q_2 , which itself depends on Q_1 , $\bar{\mathbf{a}}_{\text{PRP}}$ and its clicked documents. This dependence allows τ to adapt to the hidden relevance preferences of the user over the course of the search process.

It may not always be clear how to update the relevance score based on a given observation, the most straightforward setting for τ can simply be to set $r_a = 0$ for actions already chosen by the IR system. Because τ enforces the temporal

dependency, it is the most important aspect in the dynamic utility because without it the utility is static.

Actions. The action space is what distinguishes search tasks from one another and it is the size of this space that dictates the complexity of optimizing over the dynamic utility function. For example, the action space in query suggestion or document ranking is potentially infinite whereas the space of available advertisements in an ad selection problem may be small and finite. In our example the action space is any potential grouping of the documents in the 2D term space (four such groupings are shown in Fig. 1c). Along with τ , the setting of the static utility U_S is important for determining the desirable features of the optimal action sequence $\bar{\mathbf{a}}^*$, such as results diversification.

Observations. The observation space is dependent on the action space, its elements representing the user’s response to system actions. Each observation must contain some signal of relevance or search intent, otherwise we would have $\tau(a, r, o) = \tau(a, r)$ and lose the temporal dependency. In some cases the value of the observation likelihood is simply $P(o|a, r) = r$, for instance in search tasks where accurate explicit relevance feedback is guaranteed. Otherwise, in most situations the observations will be click-related and thus the observation probability is the probability of click, as is the case in our example (Fig. 1).

Stages. Typically, the stages in a *DIR* task will represent distinct interactions occurring in a linear time order. In these cases $\omega(t)$ may take a value between 0 and 1 or be set to a monotonically decreasing function that favorably weights the utility scores of immediate stages. Setting $\omega(1) = 1$ and $\omega(t) = 0$ for $t > 1$ gives us the static and interactive scenarios.

Alternatively, a non-linear sequence of interactions (or *search path*) can be modeled as Yang and Lad did with their session-based utility function [37]. For instance in session search, a search path represents a particular sequence of documents examined by the user and the query reformulations made. For our framework, the stage t may instead represent a specific search path, and so $\omega(t)$ could be interpreted as the likelihood of this path rather than an explicit discount, penalizing improbable search paths and rewarding likely ones.

The time horizon T dictates the number of advance stages to optimize for. A large time horizon will lead to explorative action strategies that benefit later stages. In our experiments, we set $T = 2$ so that we only consider exploitative optimizations for the immediate next stage.

Dynamic Utility. Through the recursive evaluation of the utility function we not only learn the optimal sequence of actions to make in the dynamic system, but we also learn the optimal action for each possible observation at each stage. If we were to store these in a lookup table ahead of deployment, then the dynamic system would be immediately responsive to user feedback and able to cater to a population of users. Nonetheless, the construction and storage complexity of such a table may prove intractable. We also note that the static utility U_S may be set as the dynamic utility function U_D of a nested subproblem in the search task. For example, the utility of choosing an optimal *ranking of documents* may be embedded in the utility for determining an

optimal *sequence of rankings* for a user in a session, which itself may be defined within the context of modeling a user's *topic preference* from search sessions in their search history.

3.3 Links to Existing Work

Building on our analysis of the *DIR* framework, here we identify links between its components and other related work in *IIR* and session search.

3.3.1 IIR-PRP

The *IIR-PRP* [12] is a framework designed for interactive IR in the traditional sense. The objective function in *IIR-PRP* balances the costs and benefits of choosing actions within a sequence and bears some similarities to our dynamic utility function. Nonetheless, by lacking any form of user feedback or temporal dependency, we do not describe the model as interactive or dynamic, and as already discussed, within the terminology used in this paper this means that it is actually a static method.

By mapping our notation onto the *IIR-PRP* objective function, we get

$$U_{IIR}(\vec{a}, \vec{r}) = \sum_{i=1}^M \left[\prod_{j=1}^{i-1} (1 - r_j) \right] \times \left(\omega(i) + r_i \sum_{o \in \vec{a}} P(o|a_i, r_i) U_S(a_i, r_i) \right) \quad (3)$$

evaluated over a sequence of M actions (usually a ranking of documents). Eq. (3) is a generalization of the formula originally defined, where we recognize that the cost and benefit parameters are simply a utility value, that the probability of whether a user continues searching or not is the observation likelihood, and that the cost of reaching a specific action is the path discount.

Further to this, the *IIR-PRP* defines a simple ranking rule according to this utility model, ranking documents in order of decreasing value of function $\varrho(a, r)$, which balances the costs and benefits of choosing an action as well as its probability of relevance. We can define ϱ in our setting as

$$\varrho(\vec{a}_{1...i}, \vec{r}_{1...i}, \omega) = U_S(\vec{a}_{1...i}, \vec{r}_{1...i}) - \frac{\omega}{r_{a_i}} \prod_{j=1}^{i-1} (1 - r_{a_j}) \quad (4)$$

where the utility of adding an action a_i to an existing sequence is countered by the path discount and the probability of not finding previous actions relevant. We implement this in our algorithm for *IIR-PRP* in Algorithm 2 in the next section.

3.3.2 Session-Based Utility

There have been recent advances in the modeling of user benefit across queries in search sessions. This is in recognition of the fact that ad-hoc retrieval often occurs over multiple queries in a session [35], with one study finding that 32% of search sessions consisted of at least three queries [17]. A simple approach has been to extend discount-gain metrics such as DCG and Average Precision, typically associated with static retrieval, across multiple stages. Using our terminology, a discount-gain function for a single stage has the form $\sum_{i=1}^M \omega(i) U_S(a_i, r_i)$. For example, in the DCG metric the setting would be $\omega(i) = \frac{1}{\log_2(i+1)}$ and $U_S(a_i, r_i) = 2^{r_{a_i}} - 1$. For the session-DCG (sDCG) [18] metric, a single

layer of recursion is introduced, where

$$sDCG = \sum_{t=1}^T \omega(t) U(\vec{a}_t, \vec{r}_t) \quad (5)$$

and the discount and gain functions are set as $\omega(t) = \frac{1}{\log_2(t+1)}$ and

$$U(\vec{a}_t, \vec{r}_t) = DCG(\vec{a}_t, \vec{r}_t) = \sum_{i=1}^M \underbrace{\frac{1}{\log_2(i+1)}}_{\omega(i)} \times \underbrace{(2^{r_{a_{ti}}} - 1)}_{U_S(a_{ti}, r_{ti})}$$

Here, the stages operate across a linear sequence of search rankings. In the session-Average Precision (sAP) metric, the path of interaction taken by the user is unknown and so the metric function marginalizes over the space of all such paths to find the expected sAP [21].

4. APPLICATION

So far we have formulated a theoretical framework for dynamic IR and derived the dynamic utility function U_D given in Eq. (2). In this section we apply this framework to the multi page search problem in *DIR*. In doing so we demonstrate functional settings for the elements in the framework and their implementation in a workable algorithm, which gives us useful insight into the practical limitations of optimizing over U_D . We compare our algorithm against *PRP* and *IIR-PRP* based approaches in experiments using TREC data and also investigate static and interactive variants of our objective function. Through this we gain understanding of the effect that dynamic utility optimization has on the quality and diversity of rankings in multi page search.

4.1 Multi Page Search Problem

The *Multi Page Search* (MPS) scenario concerns the ranking of documents over multiple pages of search results [22, 19]. MPS typically models exploratory search queries which are more likely to lead to multi-query sessions and multi-page searches [35] (with one study finding that 27% of all searches occur over multiple pages [16]). In this scenario documents are retrieved for a single query, ranked and then segregated into pages of M documents. On each page, a user may examine and click on documents. We assume that the user will return to the results page and move onto the next page and we define a threshold of T pages which the user will search over. The goal in MPS is to create rankings of relevant documents across T pages. For the pages following the first, document clicks can be used to personalize search rankings, a situation analogous to our example in Fig. 1.

We chose this particular problem to apply our dynamic IR framework to as it exhibits the following beneficial features: 1) it is a *DIR* problem that is familiar and easy to define, 2) it is a simple IR scenario where we only have to consider a single query and a single set of documents, 3) we can use existing ad hoc ranking and retrieval research to find suitable implementations for the *DIR* framework components, 4) we can readily use TREC data collections and relevance judgments to evaluate our algorithms, and 5) it is naturally translatable to the *PRP* and *IIR-PRP* frameworks. A similar analysis of *DIR* in the session search scenario was also conducted by Luo et al. [24]

In this scenario, each page of search represents a stage in our framework, with T the threshold number of pages. We

nominally set $T = 2$ although a larger number of pages is feasible and has been studied by Jin et al. [19]. The action sequence $\vec{a}_t = \langle a_{t1}, \dots, a_{tM} \rangle$ represents the ranking of documents for ranks 1 to M on page t . Before we can fully implement Eq. (2), we must first define each of its functional components in the context of MPS.

4.1.1 Expected DCG

The static utility in multi page search is a measure of the quality of the ranking of documents on each page. As with ad hoc ranking and retrieval, we can evaluate this using a metric such as DCG, MAP or ERR. In the absence of relevance judgments, we can instead find the *expected* metric value which uses probabilities of relevance instead [33]. In our application we set the static utility as the expected DCG function, given by

$$U_S(\vec{a}_t, \vec{r}_t) = \sum_{i=1}^M \frac{2^{r_{a_{ti}}} - 1 + 2^{r_{a_{ti}}-1} \log^2(2) \text{Var}[r_{a_{ti}}]}{\log(i+1)} \quad (6)$$

This utility also takes into consideration the variance of the document's probability of relevance.

4.1.2 Examination Hypothesis

In multi page search our observations are document clicks, which we regard as an implicit signal of the relevance of a document to the user. Thus, we can utilize the clicks from previous search pages to update our probability of relevance model and personalize the document rankings for future pages.

For a ranking of M documents, the observation space \mathcal{O} in MPS for a particular page is the combination of binary click events for each document in the ranking. We denote this as the observation vector $\vec{o} = \langle o_1, \dots, o_M \rangle$ where $o \in \{0, 1\}$. We could naively set the observation likelihood to the uniform distribution $P(o|a, r) = \frac{1}{|\mathcal{O}|}$ but eye-tracking studies tell us that this is not the case. Instead, the probability of a click occurring on a ranked document is dependent on not only its probability of relevance but also its rank position, amongst other variables [20]. The probabilistic modeling of user clicks is an extensive area of IR research and in our application we use the simplest model, the *Examination Hypothesis* model [10].

This model supports the eye tracking research by inferring that the probability of a click on a document in a ranked list is equal to the product of its probability of relevance and the bias of its rank position. Thus, the probability of a sequence of clicks is given by

$$P(\vec{o}|\vec{a}_t, \vec{r}_t) = \prod_{i=1}^M (b_i r_{a_{ti}})^{o_i} (1 - b_i r_{a_{ti}})^{1-o_i} \quad (7)$$

where b_i is a rank bias parameter. In our implementation we set Eq. (7) as our observation likelihood function and set $b_i = \frac{1}{\log(i+1)}$ which is the discount value used in our expected DCG utility.

4.1.3 Conditional Multivariate Gaussian Distribution

Once we have a sequence of click observations for a ranking of documents, we can update the probability of relevance distribution for the remaining documents. Here, we achieve this by defining the distribution of all the probabilities of relevance for all documents in the collection as a multivariate

Gaussian distribution $R \sim \mathcal{N}(\vec{r}, \Sigma)$, where R is their collective random variable, \vec{r} the vector of mean relevance scores and Σ the covariance matrix over the documents. \vec{r} may be set as any relevance score and Σ may be set using document similarity or other correlation scores [19]. If \vec{r} represents a probability of relevance, then we can set the distribution as a *truncated* multivariate Gaussian bounded between 0 and 1. If it is not possible to define the distribution of a relevance score, then the distribution of the mean of multiple relevance scoring techniques can be derived, resulting in an approximately Gaussian distribution that incorporates multiple signals of relevance. It is this approach we take in our experiment, where we set \vec{r} and Σ as the means and variances of the retrieval scores from five well known techniques, with the diagonal elements from Σ used as variance values for our utility calculation in Eq. (6).

Modeling the relevance distribution in this way allows us to conditionally update the probabilities of relevance \vec{r} based on our click observations. For a given rank action \vec{a}_t (which includes both clicked and non-clicked documents in the ranking), we denote the remaining non-ranked documents as $\backslash \vec{a}_t$ and partition our distribution parameters as

$$\vec{r} = \begin{bmatrix} \vec{r}_{\backslash \vec{a}_t} \\ \vec{r}_{\vec{a}_t} \end{bmatrix} \quad \Sigma = \begin{bmatrix} \Sigma_{\backslash \vec{a}_t \backslash \vec{a}_t} & \Sigma_{\backslash \vec{a}_t \vec{a}_t} \\ \Sigma_{\vec{a}_t \backslash \vec{a}_t} & \Sigma_{\vec{a}_t \vec{a}_t} \end{bmatrix}$$

We can then update the mean relevance scores and covariance matrix for non ranked documents using the formulae

$$\begin{aligned} \vec{r}_{\backslash \vec{a}_t} &= \vec{r}_{\backslash \vec{a}_t} + \Sigma_{\backslash \vec{a}_t \vec{a}_t} \Sigma_{\vec{a}_t \vec{a}_t}^{-1} (\vec{o} - \vec{r}_{\vec{a}_t}) \\ \Sigma_{\backslash \vec{a}_t \backslash \vec{a}_t} &= \Sigma_{\backslash \vec{a}_t \backslash \vec{a}_t} - \Sigma_{\backslash \vec{a}_t \vec{a}_t} \Sigma_{\vec{a}_t \vec{a}_t}^{-1} \Sigma_{\vec{a}_t \backslash \vec{a}_t} \end{aligned} \quad (8)$$

and observations \vec{o} . Thus, for given actions and observations, we can use the functions above to define a new conditional multivariate Gaussian distribution of the probability of relevance of the remaining documents, given as $R_{t+1} \sim \mathcal{N}(\vec{r}_{\backslash \vec{a}_t}, \Sigma_{\backslash \vec{a}_t \backslash \vec{a}_t} | \vec{a}_t, \vec{o})$. For the multi page search setting, we define $\tau(a, r, o)$ as the relevance update function in Eq. (8).

4.1.4 Geometric Discount

The final component required for our application is the discount function $\omega(t)$. In the multi page scenario, we measure the utility of a linear sequence of document rankings rather than the path-based behavior of users. As such, we adopt the simple discount used in a POMDP, setting $\omega = \lambda$ (which is effectively setting it as the geometric discount $\omega(t) = \lambda^{t-1}$ due to the recursion of the dynamic utility).

Here, we can consider $\omega(t)$ as the probability of the user visiting page t . When $\lambda = 0$, we assume only the first page will be visited, and when $\lambda = 1$ all pages are equally likely and given equal weight. The optimal setting for λ will vary depending on the type of searches being performed as well as the corpus and quality of results.

4.2 DIR-MPS

Now that we have defined each of the functional components of the dynamic IR framework for the multi page search scenario, we present the DIR-MPS algorithm in Algorithm 1. This algorithm is a direct implementation of the recursive utility function U_D in Eq. (2) that determines the optimal sequence of document rankings to display for each page.

It is worth noting that Algorithm 1 and the described settings for the *DIR* framework elements are one such instantiation of the framework in the multi page search scenario.

Algorithm 1 The DIR-MPS Algorithm

```

function DIR-MPS( $t, \vec{r}, \mathcal{A}$ )
  if  $t = T + 1$  then return  $[0, \langle \rangle]$ 
  end if
   $\vec{a}_t^* = \langle \rangle; \vec{a}_{t+1}^* = \langle \rangle$ 
  loop  $i \leftarrow 1$  to  $M$   $\triangleright$  Sequential Ranking Decision
     $\vec{a} = \vec{a}_t^*; u^* = 0$ 
    for all  $a \in \mathcal{A} \setminus \vec{a}$  do
       $\vec{a}_t = \langle \vec{a}, a \rangle$ 
       $u_t = U_S(\vec{a}_t, \vec{r}_t)$   $\triangleright$  Eq. (6)
      for all  $\vec{o} \in \mathcal{O}$  do
         $\vec{r}_{t+1} = \tau(\vec{a}_t, \vec{r}, o)$   $\triangleright$  Eq. (8)
         $[u_{t+1}, \vec{a}_{t+1}] = \text{DIR-MPS}(t + 1, \vec{r}_{t+1}, \mathcal{A} \setminus \vec{a}_t)$ 
         $u_t = u_t + \lambda \cdot P(\vec{o} | \vec{a}_t, \vec{r}_t) \cdot u_{t+1}$   $\triangleright$  Eq. (7)
      end for
      if  $u_t > u^*$  then
         $u^* = u_t; \vec{a}_t^* = \vec{a}_t; \vec{a}_{t+1}^* = \vec{a}_{t+1}$ 
      end if
    end for
  end loop
  return  $[u^*, \langle \vec{a}_t^*, \vec{a}_{t+1}^* \rangle]$ 
end function

```

Our motive in this section is not to develop a state of the art new ranking technique but rather to demonstrate the application of the framework to a *DIR* problem.

4.2.1 Dynamic Utility Approximation

The DIR-MPS algorithm features a number of approximation techniques that increase its computational efficiency as a way to counteract the inherent complexity of the *DIR* framework (discussed further in Section 4.3). Firstly, we reduce the action space of potential rankings by employing a *Sequential Ranking Decision* policy. That is, for each page we find the optimal document to rank at each position one by one. For example, we set $M = 1$ and find the document a^* that maximizes $U_S(a, r_a)$. Then we fix this document, set $M = 2$ and find the next in the sequence that maximizes $U_S(\langle a^*, a \rangle, \vec{r}_{\langle a^*, a \rangle})$. Continuing in this fashion allows us to find an approximately optimal ranking for a single page, one document at a time, greatly reducing the computational complexity.

A property of the probability distribution given in Eq. (7) also allows us to greatly reduce the observation space. We find that this distribution follows Zipf’s law, with a few of the click combinations contributing towards most of the probability mass. In fact, from our experiments we typically found that around 15% of the combinations contributed to 95% of the aggregated probability. As such, in our implementation of DIR-MPS we restrict the observation space to only the most probable click combinations that cumulatively sum to 0.95, trading off the potential 5% inaccuracy for speed.

Finally, it can be shown that when ranking over a single stage, the expected DCG utility function is maximized when documents are ranked according to the *PRP* [33]. We exploit this to increase the efficiency of our algorithm by ranking the threshold page (where we no longer consider a future temporal dependency) according to the *PRP* over the conditionally updated probabilities of relevance.

Algorithm 2 The IIR-PRP-MPS algorithm

```

function IIR-PRP-MPS( $M, T, \lambda, \mathcal{A}, \vec{r}$ )
   $\vec{a}^* = \langle \rangle$ 
  loop  $i \leftarrow 1$  to  $M \times T$ 
     $a^* = \text{argmax}_{a \in \mathcal{A} \setminus \vec{a}^*} \varrho(\langle \vec{a}^*, a \rangle, \vec{r}, \lambda)$   $\triangleright$  Eq. (4)
     $\vec{a}^* = \langle \vec{a}^*, a^* \rangle$ 
  end loop
  return  $\vec{a}^*$ 
end function

```

4.2.2 IIR-PRP-MPS

In our experiments we directly compare DIR-MPS against rankings created from the applied *PRP* and *IIR-PRP* ranking rules. With the *Probability Ranking Principle* we can simply rank documents in decreasing order of the probability of relevance across T pages. However, the *IIR-PRP* has no existing direct application to our scenario. Instead, we use our definition of the ranking function ϱ given in Eq. (4) to create the IIR-PRP-MPS algorithm shown in Algorithm 2.

Here, the sequential ranking rule is also employed to build up an optimal ranking over all pages, one document at a time, by selecting the document that has the highest ϱ value for each rank. Thus, there is some dependency on previously ranked documents, which is not possible in the *PRP*, but like the *PRP* there is no way to take into account user feedback or update the probabilities of relevance.

4.3 Practical Limitations

The general computational complexity of the optimization of U_D can be shown to be PSPACE-Complete (through its connection to *POMDPs*). For small T and observation and action spaces this can be reasonable, but typically these spaces may be impractically large.

For example, an IR task such as information filtering or music recommendation may operate over potentially infinite time steps. In these cases the discount factor and threshold T are important. Further, the observation space may not be as well defined as that in our multi page search scenario where $|\mathcal{O}| = 2^M$, for example, the space of possible reformulations for a query or 2D gaze positions in eye-tracking. Finally, the action space can be difficult to optimize over as is the case with DIR-MPS, where the sequential ranking decision reduces the size of the action space from $O(N!/(N - TM)!)$ to $O(TNM - TM^2)$ for a collection of N documents. Our application serves to demonstrate that such approximations may be needed when working with the *DIR* framework, especially given that the optimization of U_D is not guaranteed to be tractable, and an optimal solution may not exist depending on the particular problem settings.

4.4 Experiment

To gain insight into our application of the dynamic IR framework in the multi page search scenario, and to compare with the other theoretical frameworks, we conducted an experiment using the WT10g, AQUAINT and ClueWeb09 datasets, the details of which are included in Table 1. We chose these collections as they were designed for evaluating ranking and retrieval algorithms and were easily extended to the multi page problem. The WT10g dataset allowed us to test the theoretical frameworks in the standard ad hoc ranking and retrieval environment. The Robust data consists of

Table 1: Overview of the three TREC test collections

Name	Task	# Docs	Topics
WT10g	TREC 9 Web Track	1,692,096	451-500
AQUAINT	Robust 2005	1,033,461	50 difficult Robust 2004 topics
ClueWeb09	Diversity Task 2009/10	503,903,810	1-100 (461 subtopics)

difficult to rank ad hoc queries which we hypothesized would be more likely to require several pages of search results. The diversity track data allowed us to test our hypothesis that dynamic optimization leads to increased diversification in ad hoc retrieval. A drawback to using these datasets is that they lack interaction data, which is not needed when optimizing for probable clicks in the *DIR* framework, but important in the interactive setting.

On each collection we retrieved the top 100 documents for each topic scored using each of the TF-IDF, BM25, Jelinek-Mercer, Dirichlet and Two-Stage language model retrieval methods from the Indri¹ search engine. We pooled the documents and subsequently scored them over all the techniques. This gave us an average of 193 ranked documents per topic each with 5 relevance score values. After min-max normalization we averaged each score to give us our probability of relevance vector $\bar{\mathbf{r}}$ and covariance matrix Σ . The dependencies in this covariance matrix reflect the level of agreement between the different retrieval methods rather than direct correlations between the documents themselves i.e. similarly ranked documents will be positively correlated with one another. Finally, we selected those documents that had the top 30 mean relevance scores. These were then used by our algorithms to create rankings for two pages of search results with ten documents on each.

We ranked these documents according to the baseline *PRP* approach and also the already described *IIR-PRP-MPS* and *DIR-MPS* algorithms. We also investigated an interactive version of *DIR-MPS* (called *IIR-MPS*) that ranks the first page of results according to the *PRP* and then optimizes a ranking for the second page of results by marginalizing over potential clicks using Eq. (1). We also created a static version of *DIR-MPS* (called *S-MPS*) that removes feedback from U_D entirely to give us the objective function $U_S(\bar{\mathbf{a}}_1, \bar{\mathbf{r}}) + \lambda U_S(\bar{\mathbf{a}}_2, \bar{\mathbf{r}})$. We also investigated ‘perfect click’ variants of the dynamic (*DIR-MPS^C*) and interactive (*IIR-MPS^C*) algorithms, where we interpret the hidden relevance labels as clicks on the first page of results, giving us the optimal observation setting and an upper bound on performance for the second page.

To evaluate the quality of the rankings we measured MAP, NDCG and ERR for each page. For the *DIR-MPS* and *IIR-MPS* algorithms we actually generate an optimal 2nd page ranking for every click combination in our observation space, giving us different metrics scores for each. In these cases, the reported page two metric scores are averages over the page two scores for all click combination based rankings. This highlights an open area for research; the definition of evaluation metrics for *DIR* that can take into account all of the potential rankings in a dynamic system. We also measure the session-based metrics *sDCG* (defined in Eq. (5)) and *sAP* to evaluate performance over both pages, although it

Table 3: α -DCG, IA-Precision and ERR-IA scores for page 1 and 2 search results from the diversity track data. The maximum score for each metric on each page is given in boldface. A ² indicates that the result is significantly better than the *IIR-PRP-MPS* baseline score using the Wilcoxon signed-rank test ($p < 0.05$).

Algorithm	Page 1			Page 2		
	α -DCG	IA-Prec	ERR-IA	α -DCG	IA-Prec	ERR-IA
PRP	0.360	0.083	0.239	0.420	0.085	0.294
<i>IIR-PRP-MPS</i>	0.345	0.077	0.230	0.404	0.086	0.269
<i>S-MPS</i>	0.352	0.078	0.233	0.417	0.089	0.280
<i>IIR-MPS</i>				0.377	0.079	0.243
<i>IIR-MPS^C</i>	0.360	0.083	0.239	0.379	0.080	0.236
<i>DIR-MPS</i>				0.400	0.079	0.264
<i>DIR-MPS^C</i>	0.403 ²	0.082	0.270	0.386	0.077	0.254

is worth noting that these metrics were designed for session search rather than multi page search. Finally, we also measure α -NDCG [6], Intent-Aware Precision (IA-Precision) [1] and Intent-Aware ERR (ERR-IA) [4] for scoring the diversity of rankings in the ClueWeb09 collection.

The results of our experiments are shown in Table 2. For the Web Track and Robust collections, we observe that the 1st page losses of the dynamic techniques (when compared to the *PRP* and *IIR-PRP-MPS* baselines) are made up for by gains in the second page, significantly so on the WT10g dataset. Nonetheless, in these ad hoc ranking scenarios it is clear that the static *PRP* and *IIR-PRP* frameworks are still very effective.

We see different results with the diversity track data. The metric scores for this data in Table 2 were calculated using relevance judgments from all subtopics. We see the opposite relationship between page scores here, with *DIR-MPS* having higher scores for the 1st page and losses in the second (except for ERR which is significantly improved across both pages). We see further evidence of this with the diversity metric scores in Table 3, where it is clear that diversification is occurring in the first page and less so in the second. This backs up our intuition (in Fig. 1d) that a dynamic technique will initially diversify results to improve future rankings, and also helps explain the losses in performance of the 1st page in the other datasets (which do not have subtopic relevance judgments). The reduced diversity of the 2nd page indicates that it is more tightly focused on the user’s subtopic preference.

We observe that the diversity task is more suited as an application of the dynamic IR framework. This is evidenced by the optimal settings for λ in each collection. For the ad hoc ranking task in the WT10g and AQUAINT collections, the setting for λ gives greater weight to the 1st page of results, rewarding immediately effective rankings. Whereas in the diversity task, the utility of the 2nd page has a larger effect on the overall utility, encouraging diversity.

Further to this, the interactive variant scored highly with *session based metrics* on the Robust dataset, but otherwise the static techniques were optimal, even for the diversity task. This may partly be due to the application of a session-based metric to the multi page scenario and also the inability of the metric to take into account the user interaction. Finally, we also see that the ‘perfect click’ variants generally outperform their counterparts (except over the diversity

¹<http://www.lemurproject.org/indri.php>

Table 2: NDCG, MAP and ERR scores for pages 1 and 2 of the search results and sAP and sDCG over both pages. Static, interactive and dynamic algorithms are grouped. The results shown are those for the optimal value of λ in each collection, found by repeating the experiment for values in the range $[0, 1]$. The maximum score for each metric on each page is given in boldface. A ¹ or ² indicates that the result is significantly better than the *PRP* or *IIR-PRP-MPS* baseline scores respectively using the Wilcoxon signed-rank test ($p < 0.05$).

Collection	Algorithm	Page 1			Page 2			Both Pages	
		NDCG	MAP	ERR	NDCG	MAP	ERR	sAP	sDCG
Web Track (WT10g) $\lambda = 0.5$	PRP	0.338	0.167	0.169	0.133	0.025	0.053	0.097	1.326
	IIR-PRP-MPS	0.330	0.162	0.162	0.166	0.041	0.078	0.101	1.347
	S-MPS	0.295	0.134	0.130	0.226	0.070	0.242 ¹²	0.095	1.236
	IIR-MPS				0.125	0.025	0.103 ¹	0.097	1.291
	IIR-MPS ^C	0.338	0.167	0.169	0.154	0.040	0.151 ¹	0.102	1.353
	DIR-MPS				0.212 ¹	0.054 ¹	0.289 ¹²	0.069	1.022
	DIR-MPS ^C	0.235	0.091	0.092	0.230 ¹	0.059	0.297 ¹²	0.072	1.027
Robust (AQUAINT) $\lambda = 0.5$	PRP	0.624	0.107	0.398	0.552	0.061	0.294	0.085	5.735
	IIR-PRP-MPS	0.629	0.107	0.398	0.514	0.052	0.288	0.083	5.680
	S-MPS	0.608	0.096	0.388	0.595	0.066	0.887 ¹²	0.083	5.749
	IIR-MPS				0.519	0.050	0.737 ¹²	0.081	5.543
	IIR-MPS ^C	0.624	0.107	0.398	0.554	0.057	0.690 ¹²	0.084	5.729
	DIR-MPS				0.575	0.065	0.656 ¹²	0.065	4.921
	DIR-MPS ^C	0.548	0.063	0.304	0.553	0.058	0.697 ¹²	0.062	4.909
Diversity (ClueWeb09) $\lambda = 0.8$	PRP	0.402 ²	0.049 ²	0.199	0.476	0.052	0.265	0.051 ²	1.883 ²
	IIR-PRP-MPS	0.384	0.046	0.193	0.468	0.051	0.257	0.048	1.808
	S-MPS	0.388	0.041	0.193	0.465	0.054	0.358 ¹²	0.049	1.787
	IIR-MPS				0.431	0.042	0.353 ¹²	0.047	1.783
	IIR-MPS ^C	0.402 ²	0.049 ²	0.199	0.436	0.042	0.345 ¹²	0.047	1.787
	DIR-MPS				0.445	0.042	0.373 ¹²	0.046	1.859
	DIR-MPS ^C	0.451 ²	0.047	0.238 ²	0.426	0.037	0.356 ¹²	0.044	1.839

data), indicating that the 2nd page ranking can be improved when high quality clicks are observed.

In summary, by its nature the *DIR* approach to multi-page search places greater emphasis on different stages of the search task. We find that this may not be suited to all search environments i.e. ad hoc search. In such cases the static approaches can be more effective. Nonetheless, the dynamic IR framework has other desirable properties such as the diversification and personalization of results over time.

5. RELATED WORK

Throughout this paper we have presented the dynamic IR theory within the context of the surrounding literature, so in this section we cover those areas of the related work not already discussed.

For instance, the settings of the components in the *DIR* framework for multi page search cover a wide area of research in IR. Firstly, the examination hypothesis model used is just one of a number of probabilistic click models that could have been employed, including the click-chain model [13] and even a POMDP-based model [34]. Other path-based discount functions have been explored in the literature [36] as well as other multi-stage utilities and metrics such as Time-Based Gain [30]. Related work on using Markov chains to measure the utility of rankings at each time step is a potential method for evaluating *DIR* problems [11]. Further to this, the identification of the dynamic IR framework as a POMDP raises the possibility of using established techniques such as the Witness algorithm [23] to find optimal action policies. Also, the performance of the *PRP* under results diversification is well-reported in the static frameworks quantum-PRP [38] and the portfolio theory of IR [32].

The concept of evaluating for retrieval utility rather than relevance was proposed by Cooper in 1973 [9] and is extended to all the frameworks discussed in this paper, where we aim to maximize some utility function that balances the costs and benefits of an IR system's actions. Other work in this area includes Azzopardi's [2] work on economic models, which is itself an extension of the *IIR-PRP*, and also the work of Mostafa et al. [25] who had a similar motivation to this work, where they defined a framework for running user simulations for information filtering, itself a *DIR* problem.

Other than the *PRP* and *IIR-PRP*, the closest related works to this one are the following: The application of a POMDP to multi page search [19], from which many aspects of our experiments in this paper are derived, including the problem setting and the probability of relevance distribution. This paper extends their formulation to a general one applicable to other *DIR* problems and explores a different setting for the static utility, observation likelihood and discount, while also linking to static and interactive techniques. Our experimental time horizon setting of $T = 2$ is based on the optimal results found in their work. Finally, the work on defining the elements of a POMDP in session search [24] is a close relation to this work, though focusing more on the testing of particular settings of *DIR* components in the session search scenario rather than explicitly gaining an understanding of the framework and components themselves. Nonetheless, their work contains an evaluation of algorithms that fall under the *DIR* framework including one similar to *DIR-MPS*.

This work differs from the literature in that: 1) ours is the first work to define the characteristics that distinguish dynamic IR from the other theoretical IR frameworks, 2) our

utility is the generalization of many existing ranking utilities and incorporates many elements of IR research such as click models, and 3) we confirm the effectiveness of ours and the static frameworks in different scenarios in our experiments.

6. CONCLUSION

In this paper we have established a theoretical framework for *Dynamic Information Retrieval*. By contrasting with *static* and *interactive* frameworks, we found three characteristics that define dynamic IR systems; user feedback, temporal dependency and an overall goal. This motivated the derivation of our dynamic utility function U_D , which has its roots in the POMDP formulation. The components of this utility can be directly implemented using elements from existing research which we apply in the DIR-MPS algorithm, an example instantiation designed for the multi page search problem. Our experiments confirm that in this scenario, one of the effects of optimizing for U_D is the diversification of search results. Otherwise, we also demonstrate that for other scenarios the *PRP* and *IIR-PRP* frameworks are still effective.

Like the *PRP* and *IIR-PRP*, our framework defines certain functional parameters but does not definitively specify how to set them. Instead, this work is a point of reference that can be used for the development of specialized models and algorithms applied to *DIR* problems. Through our application we were able to consider the limitations of the *DIR* framework, a result of which is the approximations used in the DIR-MPS algorithm. Finally, the derivation of a stationary solution to the *DIR* Markovian model is an intended future goal and would be an important result to come from this work.

7. REFERENCES

- [1] AGRAWAL, R., GOLLAPUDI, S., HALVERSON, A., AND IEONG, S. Diversifying search results. WSDM '09, ACM, pp. 5–14.
- [2] AZZOPARDI, L. Modelling interaction with economic models of search. SIGIR '14, ACM, pp. 3–12.
- [3] BELLMAN, R. E. *Dynamic Programming*. Dover Publications, 2003.
- [4] CHAPPELLE, O., METLZER, D., ZHANG, Y., AND GRINSPAN, P. Expected reciprocal rank for graded relevance. CIKM '09, ACM, pp. 621–630.
- [5] CHEN, H., AND KARGER, D. R. Less is more: Probabilistic models for retrieving fewer relevant documents. SIGIR '06, ACM, pp. 429–436.
- [6] CLARKE, C. L., KOLLA, M., CORMACK, G. V., VECHTOMOVA, O., ASHKAN, A., BÜTTCHER, S., AND MACKINNON, I. Novelty and diversity in information retrieval evaluation. SIGIR '08, ACM, pp. 659–666.
- [7] CLEVERDON, C., AND KEAN, M. Factors determining the performance of indexing systems. Aslib Cranfield Research Project, Cranfield, England, 1968.
- [8] COOPER, W. S. The inadequacy of probability of usefulness as a ranking criterion for retrieval system output. *University of California, Berkeley* (1971).
- [9] COOPER, W. S. On selecting a measure of retrieval effectiveness. *Journal of the American Society for Information Science* 24, 2 (1973), 87–100.
- [10] CRASWELL, N., ZOETER, O., TAYLOR, M., AND RAMSEY, B. An experimental comparison of click position-bias models. WSDM '08, ACM, pp. 87–94.
- [11] FERRANTE, M., FERRO, N., AND MAISTRO, M. Injecting user models and time into precision via Markov chains. SIGIR '14, ACM, pp. 597–606.
- [12] FUHR, N. A probability ranking principle for interactive information retrieval. *Inf. Retr.* 11, 3 (2008), 251–265.
- [13] GUO, F., LIU, C., KANNAN, A., MINKA, T., TAYLOR, M., WANG, Y.-M., AND FALOUTSOS, C. Click chain model in web search. WWW '09, ACM, pp. 11–20.
- [14] HERSH, W., TURPIN, A., PRICE, S., CHAN, B., KRAMER, D., SACHEREK, L., AND OLSON, D. Do batch and user evaluations give the same results? SIGIR '00, ACM, pp. 17–24.
- [15] JAMBOR, T., WANG, J., AND LATHIA, N. Using control theory for stable and efficient recommender systems. WWW '12, ACM, pp. 11–20.
- [16] JANSEN, B. J., AND SPINK, A. How are we searching the world wide web?: A comparison of nine search engine transaction logs. *Inf. Process. Manage.* 42, 1 (2006), 248–263.
- [17] JANSEN, B. J., SPINK, A., AND PEDERSEN, J. A temporal comparison of AltaVista web searching: Research articles. *J. Am. Soc. Inf. Sci. Technol.* 56, 6 (2005), 559–570.
- [18] JÄRVELIN, K., PRICE, S. L., DELCAMBRE, L. M. L., AND NIELSEN, M. L. Discounted cumulated gain based evaluation of multiple-query IR sessions. ECIR'08, Springer-Verlag, pp. 4–15.
- [19] JIN, X., SLOAN, M., AND WANG, J. Interactive exploratory search for multi page search results. In *WWW '13* (2013), pp. 655–666.
- [20] JOACHIMS, T., GRANKA, L., PAN, B., HEMBROOKE, H., RADLINSKI, F., AND GAY, G. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. Inf. Syst.* 25, 2 (2007).
- [21] KANOULAS, E., CARTERETTE, B., CLOUGH, P. D., AND SANDERSON, M. Evaluating multi-query sessions. SIGIR '11, ACM, pp. 1053–1062.
- [22] KIM, J. Y., CRAMER, M., TEEVAN, J., AND LAGUN, D. Understanding how people interact with web search results that change in real-time using implicit feedback. CIKM '13, ACM, pp. 2321–2326.
- [23] LITTMAN, M. L. The witness algorithm: Solving partially observable markov decision processes. Tech. rep., 1994.
- [24] LUO, J., ZHANG, S., DONG, X., AND YANG, H. Designing states, actions, and rewards for using POMDP in session search. In *Advances in Information Retrieval*, vol. 9022 of *Lecture Notes in Computer Science*. Springer International Publishing, 2015, pp. 526–537.
- [25] MOSTAFA, J., MUKHOPADHYAY, S., AND PALAKAL, M. Simulation studies of different dimensions of users' interests and their impact on user modeling and information filtering. *Information Retrieval* 6, 2 (2003), 199–223.
- [26] ROBERTSON, S. E. The Probability Ranking Principle in IR. *Journal of Documentation* 33, 4 (1977), 294–304.
- [27] ROCCHIO, J. *Relevance Feedback in Information Retrieval*. 1971, pp. 313–323.
- [28] RUTHVEN, I. Interactive information retrieval. *ARIST* 42, 1 (2008), 43–91.
- [29] SLOAN, M., YANG, H., AND WANG, J. A term-based methodology for query reformulation understanding. *Information Retrieval Journal* 18, 2 (2015), 145–165.
- [30] SMUCKER, M. D., AND CLARKE, C. L. Time-based calibration of effectiveness measures. SIGIR '12, ACM, pp. 95–104.
- [31] SONDIK, E. The optimal control of partially observable markov processes over the infinite horizon: Discounted cost. *Operations Research* 26, 2 (1978), 282–304.
- [32] WANG, J., AND ZHU, J. Portfolio theory of information retrieval. SIGIR' 09, ACM, pp. 115–122.
- [33] WANG, J., AND ZHU, J. On statistical analysis and optimization of information retrieval effectiveness metrics. SIGIR '10, pp. 226–233.
- [34] WANG, K., GLOY, N., AND LI, X. Inferring search behaviors using partially observable markov (POM) model. WSDM '10, ACM, pp. 211–220.
- [35] WHITE, R., WHITE, R., AND ROTH, R. *Exploratory Search: Beyond the Query-Response Paradigm*. Synthesis Lectures on Information Concepts, Retrieval, and Services Series. Morgan & Claypool, 2009.
- [36] WHITE, R. W., RUTHVEN, I., JOSE, J. M., AND RIJSBERGEN, C. J. V. Evaluating implicit feedback models using searcher simulations. *ACM Trans. Inf. Syst.* 23, 3 (2005), 325–361.
- [37] YANG, Y., AND LAD, A. Modeling expected utility of multi-session information distillation. In *Advances in Information Retrieval Theory*, vol. 5766 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2009, pp. 164–175.
- [38] ZUCCON, G., AZZOPARDI, L., AND VAN RIJSBERGEN, K. The quantum probability ranking principle for information retrieval. In *Advances in Information Retrieval Theory*, vol. 5766. Springer Berlin Heidelberg, 2009, pp. 232–240.