Fidelity, Soundness, and Efficiency of Interleaved Comparison Methods

KATJA HOFMANN, University of Amsterdam
SHIMON WHITESON, University of Amsterdam
MAARTEN DE RIJKE, University of Amsterdam

Ranker evaluation is central to the research into search engines, be it to compare rankers or to provide feedback for learning to rank. Traditional evaluation approaches do not scale well because they require explicit relevance judgments of document-query pairs, which are expensive to obtain. A promising alternative is the use of *interleaved comparison* methods, which compare rankers using click data obtained when interleaving their rankings.

In this article, we propose a framework for analyzing interleaved comparison methods. An interleaved comparison method has *fidelity* if the expected outcome of ranker comparisons properly corresponds to the true relevance of the ranked documents. It is *sound* if its estimates of that expected outcome are unbiased and consistent. It is *efficient* if those estimates are accurate with only little data.

We analyze existing interleaved comparison methods and find that, while sound, none meet our criteria for fidelity. We propose a *probabilistic interleave* method, which is sound and has fidelity. We show empirically that, by marginalizing out variables that are known, it is more efficient than existing interleaved comparison methods. Using importance sampling we derive a sound extension that is able to reuse historical data collected in previous comparisons of other ranker pairs.

Categories and Subject Descriptors: H.3 [**Information Storage and Retrieval**]: H.3.3 Information Search and Retrieval

General Terms: Algorithms, Evaluation

Additional Key Words and Phrases: Information retrieval, interleaved comparison, interleaving, clicks, online evaluation, importance sampling

ACM Reference Format:
Hofmann, K., Whiteson, S. A., and de Rijke, M. 2013. Probabilistic Interleaving. ACM 31, 4, Article 17 (November 2013), 39 pages.
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

1. INTRODUCTION

Evaluating the effectiveness of search result rankings is a central problem in the field of information retrieval (IR). Traditionally, evaluation using a TREC-like setting requires expert annotators to manually provide relevance judgments, i.e., to annotate whether or in how far a document is considered relevant for a given query [Voorhees and Harman 2005]. *Interleaved comparison methods* [Chapelle et al. 2012; Hofmann et al. 2011; Radlinski and Craswell 2010; Radlinski et al. 2008b], which compare rankers using naturally occurring user interactions such as clicks, are quickly gaining interest as a complement to TREC-style evaluations. Compared to evaluations based on manual relevance judgments, interleaved comparison methods rely only on data that can be collected cheaply and

This paper extends work previously published in [Hofmann et al. 2011] and [Hofmann et al. 2012b]. We extend our earlier work by deriving formal criteria for analyzing interleaved comparison methods. We add two original proofs of the unbiasedness of probabilistic interleaving under live comparisons and probabilistic interleaving with importance sampling under historical data. We also add detailed experimental evaluations of interleaved comparisons under historical data and various levels of noise in user feedback.

Author's addresses: K. Hofmann (corresponding author) and S. A. Whiteson and Maarten de Rijke, ISLA, University of Amsterdam.

DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

unobtrusively. Furthermore, since this data is based on the behavior of real users, it more accurately reflects how well their actual information needs are met.

Previous work demonstrated that two rankers can be successfully compared using click data in practice [Chapelle et al. 2012]. However, the field is largely lacking theoretical foundations for defining and analyzing properties of interleaved comparison methods. In this article, we propose to characterize these methods in terms of fidelity, soundness, and efficiency. An interleaved comparison method has *fidelity* if it measures the right quantity, i.e., if the outcome of each ranker comparison is defined such that the expected outcome properly corresponds to the true relevance of the ranked documents. It is *sound* if the estimates it computes of that expected outcome have two desirable statistical properties: namely they are unbiased and consistent. It is *efficient* if the accuracy of those estimates improves quickly as more comparisons are added.

We use the proposed framework to analyze several existing interleaved comparison methods: balanced interleave (BI) [Joachims 2003], team draft (TD) [Radlinski et al. 2008b], and document constraints (DC) [He et al. 2009]. We find that, although sound, none of these methods meet our criteria for fidelity. To overcome this limitation, we propose a new interleaved comparison method, *probabilistic interleave* (PI), and show that it is sound and has fidelity.

However, because the probabilistic approach can introduce more noise than existing interleaving methods, PI in its most naive form can be inefficient. Therefore, we derive an extension to PI that exploits the insight that probability distributions are known for some of the variables in the graphical model that describes its interleaving process. This allows us to derive a variant of PI whose estimator marginalizes out these known variables, instead of relying on noisy samples of them. We prove that the resulting estimator preserves fidelity and soundness.

We also derive a second extension to PI that broadens the applicability of interleaved comparison methods by enabling them to reuse previously observed, *historical*, interaction data. Current interleaved comparison methods are limited to settings with access to *live* data, i.e., where data is gathered during the evaluation itself. Without the ability to estimate comparison outcomes using historical data, the practical utility of interleaved comparison methods is limited. If all comparisons are done with live data, then applications such as learning to rank [Hofmann et al. 2013], which perform many comparisons, need prohibitive amounts of data. Since interleaving result lists may affect the users' experience of a search engine, the collection of live data is complicated by the need to first control the quality of the compared rankers using alternative evaluation setups. Unlike existing methods, the probabilistic nature of PI enables the use of *importance sampling* to properly incorporate historical data. Consequently, as we show, fidelity and soundness are maintained.

We evaluate previously proposed interleaved comparison methods and our PI method using experimental framework that simulates user interactions based on annotated learning to rank data sets and click models. First, we empirically validate the results of our theoretical analysis of fidelity and soundness. Then, we empirically evaluate the efficiency of our PI method and compare it to existing methods. The results show that PI with marginalization is more efficient than all existing interleaved comparison methods in the live data setting. When using only historical data, the results show that only PI can accurately distinguish between rankers.

This article makes the following contributions:

— A framework for analyzing interleaved comparison methods in terms of fidelity, soundness, and efficiency;
— A new interleaved comparison method, PI, that exhibits fidelity and soundness;
— A method that increases the efficiency of PI using marginalization, as well as a proof that this extension preserves fidelity and soundness;
— A method for applying PI to historical interaction data, as well as a proof that this extension preserves fidelity and soundness;
— An empirical validation of the theoretical results on the fidelity and soundness of interleaved comparison methods;

— A detailed experimental comparison of all interleaved comparison methods under live data and with perfect and noisy user feedback, demonstrating that PI with marginalization can infer interleaved comparison outcomes significantly more efficiently than existing methods; and
— A first experimental evaluation of interleaved comparison methods using historical data, showing that PI makes data reuse possible and effective.

Taken together, these contributions make interleaved comparisons a feasible option for large-scale evaluation and learning to rank settings.

This article is organized as follows. We present related work in §2 and background in §3. We detail our criteria for analyzing interleaved comparison methods and analyze existing methods in §4. In §5 we detail our proposed method, PI, and two extensions to make PI more efficient (marginalization and historical data reuse). Our experimental setup is presented in §6. We detail and discuss our results in §7 and conclude in §8.

2. RELATED WORK

In this section, we first discuss IR literature that is related to the use of clicks for IR evaluation in general, and interleaved comparison methods in particular (§2.1). We then give an overview of *off-policy evaluation* approaches, which allow historical data reuse in reinforcement learning (§2.2).

2.1. Click-based evaluation in IR

Click data is a promising source of information for IR systems as it can be collected practically for free, is abundant in frequently-used search applications, and (to some degree) reflects user behavior and preferences. Naturally, then, there are ongoing efforts to incorporate click data in retrieval algorithms, e.g., for pseudo-relevance feedback [Jung et al. 2007], and in learning to rank or re-rank [Ji et al. 2009; Joachims 2002].

Using click data to evaluate retrieval systems has long been a promising alternative or complement to expensive explicit judgments (also called editorial data). However, the reliability of click-based evaluation has been found to be problematic. Jung et al. [2007] found that click data does contain useful information, but that variance is high. They propose aggregating clicks over search sessions and show that focusing on clicks towards the end of sessions can improve relevance predictions. Similarly, Scholer et al. [2008] found that click behavior varies substantially across users and topics, and that click data is too noisy to serve as a measure of absolute relevance. Fox et al. [2005] found that combining several implicit indicators can improve accuracy, though it remains well below that of explicit feedback. In particular, evaluation methods that interpret clicks as absolute relevance judgments in more broadly used settings such as literature search, web search, or search on Wikipedia, were found to be rather unreliable, due to large differences in click behavior between users and search topics [Kamps et al. 2009; Radlinski et al. 2008b].

Nonetheless, in some applications, click data has proven reliable. In searches of expert users who are familiar with the search system and document collection, clicks can be as reliable as purchase decisions [Hofmann et al. 2010; Zhang and Kamps 2010]. Methods for optimizing the click-through rates in ad placement [Langford et al. 2008] and for diversifying web search results for frequent queries [Radlinski et al. 2008a] have also learned effectively from click data.

Methods that use implicit feedback to infer the relevance of specific document-query pairs have also proven effective. Shen et al. [2005] show that integrating click-through information for query-document pairs into a content-based retrieval system can improve retrieval performance substantially. Agichtein et al. [2006] demonstrate dramatic performance improvements by re-ranking search results based on a combination of implicit feedback sources, including click-based and link-based features.

The quickly growing area of click modeling develops and investigates models of users' click behavior [Chapelle and Zhang 2009; Dupret and Liao 2010; Dupret et al. 2007]. These models are trained *per query* to predict clicks and/or relevance of documents that have not been presented to users at a particular rank, or that have not been presented at all for the given query. An advantage of click models is that they directly model absolute relevance grades of individual documents. However,

it is not yet clear to what degree click models can complement or replace editorial judgments for evaluation. Extensions of click models combine inferred relevance with editorial judgments. These extensions have been found to effectively leverage click data to allow more accurate evaluations with relatively few explicit judgments [Carterette and Jones 2008; Ozertem et al. 2011]. Recently developed evaluation metrics that incorporate insights gained from click models [Chapelle et al. 2009; Moffat and Zobel 2008] provide new possibilities for combining click data and editorial judgments, further bridging the gap between click-based and traditional retrieval evaluation. The click models mentioned above can be reused to some degree but, unlike our method, do not generalize across queries.

Since implicit feedback varies so much across queries, it is difficult to use it to learn models that generalize across queries. To address this problem, so-called *interleaved comparison methods* have been developed that use implicit feedback, not to infer absolute judgments, but to compare two rankers by observing clicks on an interleaved result list [Radlinski et al. 2008b]. They work by combining pairs of document rankings into interleaved document lists, which are then presented to the user, instead of the original lists. User clicks on the interleaved list are observed and projected back to the original lists to infer which list would be preferred by users. Repeating this interleaving over many queries leads to reliable comparisons [Chapelle et al. 2012; Radlinski and Craswell 2010]. The existing interleaved comparison methods are introduced in detail in the next section (§3).

In addition to the interleaved comparison approaches detailed below, alternative methods for interpreting user actions have been investigated as a means of improving the efficiency of interleaved comparison methods [Chapelle et al. 2012; Radlinski and Craswell 2010; Yue et al. 2010]. Most recently, Radlinski and Craswell [2013] build on ideas from the conference version of this article [Hofmann et al. 2011] and propose to formulate interleaving as an optimization problem that is solved to obtain the interleaved lists that maximize the expected information gain from user clicks. Also related is work on bias in user clicks. While most work on interleaved comparison methods makes simplifying assumptions about users' click behavior and the factors that may affect users' click decisions, initial work on relaxing these assumptions is found in [Hofmann et al. 2012a]. This direction of research is complementary to this article.

2.2. Off-policy evaluation

The problem of estimating interleaved comparison outcomes using historical data is closely related to the problem of *off-policy evaluation* [Sutton and Barto 1998] in *reinforcement learning* (RL), a branch of machine learning in which agents learn from interactions with an environment by taking actions and receiving rewards [Sutton and Barto 1998]. Solving RL problems requires being able to evaluate a *policy* that specifies what actions the agent should take in each context. The challenge in off-policy evaluation is to use data gathered with one policy to evaluate another one. Doing so is difficult because the two policies may specify different actions for a given context.

Algorithms for off-policy evaluation have been developed for tasks similar to IR, namely news recommendation [Dudík et al. 2011; Li et al. 2011] and ad placement [Langford et al. 2008; Strehl et al. 2010]. In both settings, the goal is to evaluate the policy of an agent (recommendation engine, or ad selector) that is presented with a context (e.g., a user profile, or website for which an ad is sought), and selects from a set of available actions (news stories, ads). Off-policy learning in this context is hard because the data is sparse, i.e., not all possible actions were observed in all possible contexts. Solutions to this problem are based on randomization during data collection [Li et al. 2011], approximations for cases where exploration is non-random [Langford et al. 2008; Strehl et al. 2010], and combining biased and high-variance estimators to obtain more robust results [Dudík et al. 2011].

Though sparse data is also a problem in IR, existing solutions to off-policy evaluation are not directly applicable. These methods assume reward can be directly observed (e.g., in the form of clicks on ads). Since clicks are too noisy to be treated as absolute reward in IR [Kamps et al. 2009; Radlinski et al. 2008b], only relative feedback can be inferred. In §5.3, we consider how to reuse historical data for interleaved comparison methods that work with implicit, relative feedback.

However, one tool employed by existing off-policy methods that is applicable to our setting is a statistical technique called *importance sampling* [MacKay 1998; Precup et al. 2000]. Importance sampling can be used to estimate the expected value $E_T[f(X)]$ under a *target distribution* P_T when data was collected under a different *source distribution* P_S. The importance sampling estimator is:

$$E_T[f(X)] \approx \frac{1}{n} \sum_{i=1}^{n} f(x_i) \frac{P_T(x_i)}{P_S(x_i)}, \tag{1}$$

where f is a function of X, and the x_i are samples of X collected under P_S. These are then reweighted according to the ratio of their probability of occurring under P_T and P_S. This estimator can be proven to be statistically *sound* (i.e., unbiased and consistent, cf., Definition 4.3 in §4) as long as the source distribution is non-zero at all points at which the target distribution is non-zero [MacKay 1998].

Importance sampling can be more or less efficient than using the target distribution directly, depending on how well the source distribution focuses on regions important for estimating the target value. In §5.3, we use importance sampling to derive an unbiased and consistent estimator of interleaved comparison outcomes using historical data.

3. BACKGROUND

In this section, we introduce the three existing interleaved comparison methods. All three methods are designed to compare pairs of rankers $(\mathbf{l}_1(q), \mathbf{l}_2(q))$. Rankers are deterministic functions that, given a query q, produce a ranked list of documents d.[1] Given $\mathbf{l}_1$ and $\mathbf{l}_2$, interleaved comparison methods produce outcomes $o \in \{-1, 0, 1\}$ that indicate whether the quality of $\mathbf{l}_1$ is judged to be lower, equal to, or higher than that of $\mathbf{l}_2$, respectively. For reliable comparisons, these methods are typically applied over a large number of queries and the individual outcomes are aggregated. However, in this section we focus on how interleaved comparison methods compute individual outcomes. Table I gives an overview of the notation used in this section and the remainder of the article.

Table I. Notation used throughout this article. Uppercase letters indicate random variables and lowercase letters indicate the values they take on. Letters in **bold** designate vectors.

Symbol	*Description*
q	query
d	document
$\mathbf{l}$	document result list, possibly created by interleaving (special cases are $\mathbf{l}_1$ and $\mathbf{l}_2$, which are generated by two competing rankers for a given query)
r	rank of a document in a document list
$\mathbf{a}$	assignment, a vector of length $len(\mathbf{l})$ where each element $\mathbf{a}[r] \in \{1, 2\}$ indicates whether the document at rank r of an interleaved document result list $\mathbf{l}$, $\mathbf{l}[r]$ was contributed by $\mathbf{l}_1$ or $\mathbf{l}_2$ (or by softmax functions $s(\mathbf{l}_1)$ or $s(\mathbf{l}_2)$, respectively)
$\mathbf{c}$	a vector of user clicks observed on a document result list $\mathbf{l}$
$s(\mathbf{l})$	softmax function over a given list, cf., §5.1, Eq. 3
o	$\in \{-1, 0, +1\}$, outcome of an interleaved comparison

The *balanced interleave* (BI) method [Joachims 2003] generates an interleaved result list $\mathbf{l}$ as follows (see Algorithm 1, lines 3–12). First, one of the result lists is randomly selected as the starting list and its first document is placed at the top of $\mathbf{l}$. Then, the non-starting list contributes its highest-ranked document that is not already part of the list. These steps repeat until all documents have been added to $\mathbf{l}$, or until it has the desired length. Next, the constructed interleaved list $\mathbf{l}$ is displayed to the user, and the user's clicks on result documents are recorded. The clicks $\mathbf{c}$ that are observed are then attributed to each list as follows (lines 13–17). For each original list, the rank of the lowest-ranked document that received a click is determined, and the minimum of these values is denoted as k. Then,

[1] If it is clear from the context which q is referred to, we simplify our notation to $\mathbf{l}_1$ and $\mathbf{l}_2$.

ALGORITHM 1: Balanced Interleaving, following [Chapelle et al. 2012].

Input: $\mathbf{l}_1, \mathbf{l}_2$
$\mathbf{l} = []$; $i_1 = 0$; $i_2 = 0$
$first\_1 = random\_bit()$
while $(i_1 < len(\mathbf{l}_1)) \wedge (i_2 < len(\mathbf{l}_2))$ **do**
  if $(i_1 < i_2) \vee ((i_1 == i_2) \wedge (first\_1 == 1))$ **then**
    if $\mathbf{l}_1[i_1] \notin \mathbf{l}$ **then**
      $append(\mathbf{l}, \mathbf{l}_1[i_1])$
    $i_1 = i_1 + 1$
  else
    if $\mathbf{l}_2[i_2] \notin \mathbf{l}$ **then**
      $append(\mathbf{l}, \mathbf{l}_2[i_2])$
    $i_2 = i_2 + 1$
// present $\mathbf{r}$ *to user and observe clicks* $\mathbf{c}$*, then infer outcome (if at least one click was observed)*
$d_{max} =$ lowest-ranked clicked document in $\mathbf{l}$
$k = min\{j : (d_{max} = \mathbf{l}_1[j]) \vee (d_{max} = \mathbf{l}_2[j])\}$
$c_1 = len\{i : c[i] = true \wedge \mathbf{l}[i] \in \mathbf{l}_1[1..k]\}$
$c_2 = len\{i : c[i] = true \wedge \mathbf{l}[i] \in \mathbf{l}_2[1..k]\}$
return -1 **if** $c_1 > c_2$ **else** 1 **if** $c_1 < c_2$ **else** 0

ALGORITHM 2: Team Draft Interleaving, following [Chapelle et al. 2012].

1: **Input**: $\mathbf{l}_1, \mathbf{l}_2$
2: $\mathbf{l} = []$; $\mathbf{a} = []$
3: **while** $(\exists i : \mathbf{l}_1[i] \notin \mathbf{l}) \vee (\exists i : \mathbf{l}_2[i] \notin \mathbf{l})$ **do**
4: **if** $count(\mathbf{a}, 1) < count(\mathbf{a}, 2) \vee (rand\_bit() == 1)$ **then**
5: $k = min\{i : \mathbf{l}_1[i] \notin \mathbf{l}\}$
6: $append(\mathbf{l}, \mathbf{l}_1[k])$
7: $append(\mathbf{a}, 1)$
8: **else**
9: $k = min\{i : \mathbf{l}_2[i] \notin \mathbf{l}\}$
10: $append(\mathbf{l}, \mathbf{l}_2[k])$
11: $append(\mathbf{a}, 2)$
// present $\mathbf{l}$ *to user and observe clicks* $\mathbf{c}$*, then infer outcome*
12: $c_1 = len\{i : c[i] = true \wedge \mathbf{a}[i] == 1\}$
13: $c_2 = len\{i : c[i] = true \wedge \mathbf{a}[i] == 2\}$
14: **return** -1 **if** $c_1 > c_2$ **else** 1 **if** $c_1 < c_2$ **else** 0

the clicked documents ranked at or above k are counted for each original list. The list with more clicks in its top k is deemed superior. The lists tie if they obtain the same number of clicks.

The alternative *team draft* (TD) method [Radlinski et al. 2008b] creates an interleaved list following the model of "team captains" selecting their team from a set of players (see Algorithm 2). For each pair of documents to be placed on the interleaved list, a coin flip determines which list gets to select a document first (line 4). It then contributes its highest-ranked document that is not yet part of the interleaved list. The method also records which list contributed which document in an *assignment* $\mathbf{a}$ (lines 7, 11). To compare the lists, only clicks on documents that were contributed by each list (as recorded in the assignment) are counted towards that list (lines 12–14), which ensures that each list has an equal chance of being assigned clicks. Again, the list that obtains more clicks wins the comparison. Recent work demonstrates that the team draft method can reliably identify the better of two rankers in practice [Chapelle et al. 2012; Radlinski and Craswell 2010].

Neither the balanced interleave nor the team draft method takes relations between documents explicitly into account. To address this, He et al. [2009] propose an approach that we refer to as the *document constraint* method (see Algorithm 3). Result lists are interleaved and clicks observed as for

ALGORITHM 3: Interleaving with Document Constraints, following [He et al. 2009].

```
1:  Input: l_1, l_2
2:  l = []; i_1 = 0; i_2 = 0
3:  first_1 = random_bit()
4:  while (i_1 < len(l_1)) ∧ (i_2 < len(l_2)) do
5:    if (i_1 < i_2) ∨ ((i_1 == i_2) ∧ (first_1 == 1)) then
6:      if l_1[i_1] ∉ l then
7:        append(l, l_1[i_1])
8:      i_1 = i_1 + 1
9:    else
10:     if l_2[i_2] ∉ l then
11:       append(l, l_2[i_2])
12:     i_2 = i_2 + 1
    // present l to user and observe clicks c, then infer outcome
13: v_1 = violated(l, c, l_1)   // count constraints inferred from l and c that are violated by l_1
14: v_2 = violated(l, c, l_2)   // count constraints inferred from l and c that are violated by l_2
15: return −1 if v_1 < v_2 else 1 if v_1 > v_2 else 0
```

the balanced interleave method (lines 3–12). Then, following [Joachims 2002], the method infers constraints on pairs of individual documents, based on their clicks and ranks. Two types of constraints are defined: (1) for each pair of a clicked document and a higher-ranked non-clicked document, a constraint is inferred that requires the former to be ranked higher than the latter; (2) a clicked document is inferred to be preferred over the next unclicked document.[2] The method compares the inferred constraints to the original result lists and counts how many constraints are violated by each. The list that violates fewer constraints is deemed superior. Though more computationally expensive, this method proved more reliable than either balanced interleave or team draft on synthetic data [He et al. 2009].

4. ANALYSIS

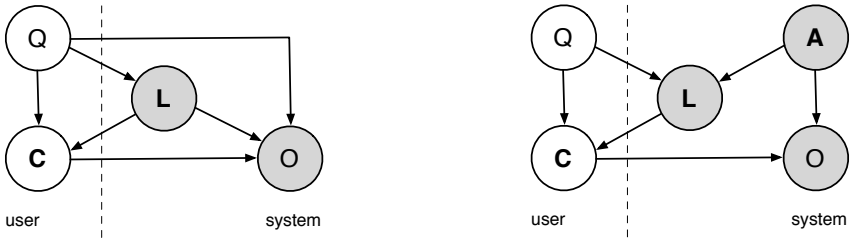
We analyze interleaved comparison methods using a probabilistic framework, and three criteria – fidelity, soundness, and efficiency – that are formulated on the basis of this framework. In this section, we first introduce our probabilistic framework and show how it relates to existing interleaved comparison methods (§4.1). Next, we formally define our criteria for analyzing interleaved comparison methods (§4.2). Finally, we use these criteria to analyze the existing interleaved comparison methods (§4.3–§4.5).

4.1. Framework

The framework we propose in this section is designed to allow systematic assessment of interleaved comparison methods. In our framework, interleaved comparison methods are described probabilistically using graphical models, as shown in Figure 1. These models specify how a retrieval system interacts with users and how observations from such interactions are used to compare rankers. Generally, an interleaved comparison method is completely specified by the components shown in gray, in the "system" part of the model. Figure 1(a)) shows one variant of the model, used for BI and DC, and Figure 1(b) shows another, used for TD and PI. (PI is introduced in §5.)

Both variants include the four random variables Q, $\mathbf{L}$, $\mathbf{C}$, and O. The interaction begins when the user submits a query $q \sim P(Q)$ to the system. We assume that $P(Q)$, though unknown to the system, is static and independent of its actions. Based on q, a result list $\mathbf{l} \sim P(\mathbf{L})$ is generated and presented

[2]Variants of this method can be derived by using only the constraints of type (1), or by using an alternative constraint (2) where only unclicked documents are considered that are ranked immediately below the clicked document. In preliminary experiments, we evaluated all three variants and found the one using constraints (1) and (2) as stated above to be the most reliable. Note that only constraints of type (1) were used in earlier work, reported on in [Hofmann et al. 2011, 2012b].



(a) Graphical model for BI and DC (b) Graphical model for TD and PI

Fig. 1. Probabilistic model for comparing rankers (a) using BI and DC, and (b) using TD and PI. Conditional probability tables are known only for variables in gray.

to the user. Because we deal with interleaving methods, we assume that $\mathbf{l}$ is an interleaved list that combines documents obtained from the two (deterministic) rankers $\mathbf{l}_1(q)$ and $\mathbf{l}_2(q)$. Thus, given q, an interleaving method completely defines $P(\mathbf{L})$ (e.g., Algorithm 1, lines 1–12). The interleaved list $\mathbf{l}$ is returned to the user, who examines it and clicks on documents that may be relevant for the given q, resulting in an observation $\mathbf{c} \sim P(\mathbf{C})$ that is returned to the system. The system then uses $\mathbf{c}$, and possibly additional information, to infer a comparison outcome $o \sim P(O)$. O, which is specified by the comparison step of the method (e.g., Algorithm 1, lines 13–15), is a deterministic function of the other variables but is modeled as a random variable to simplify our analysis.

The optional components defined in the model are the dependencies of O on Q and $\mathbf{L}$ for BI and DC (cf., Figure 1(a)), and the assignments $\mathbf{A}$ for TD and PI (cf., Figure 1(b)). As shown in Algorithms 1 and 3, BI and DC compute outcomes using the observed $\mathbf{c}$, $\mathbf{l}$, and q (specifically, the $\mathbf{l}_1$ and $\mathbf{l}_2$ generated for that q). In contrast, the comparison function of TD (and of PI, as we will see in §5) does not require $\mathbf{l}$ and q, but rather uses assignments $\mathbf{a} \sim P(\mathbf{A})$ that indicate to which original ranking function the documents in $\mathbf{l}$ are assigned (cf., Algorithm 2).

The random variables in the model have the following sample spaces. For Q, it is the (possibly infinite) universe of queries, e.g., $q =$ '*facebook*'. For $\mathbf{L}$ it is all permutations of documents, e.g., $\mathbf{l} = [d_1, d_2, d_3, d_4]$. For $\mathbf{C}$ it is all possible click vectors, such that $\mathbf{c}[i]$ is a binary value that indicates whether the document $\mathbf{l}[i]$ was clicked, e.g., $\mathbf{c} = [1, 0, 0, 0]$. For $\mathbf{A}$ it is all possible assignment vectors, such that $\mathbf{a}[i]$ is a binary value that indicates which ranker contributed $\mathbf{l}[i]$, e.g., $\mathbf{a} = [1, 2, 1, 2]$.

Within this framework, we are particularly interested in the sign of the expected outcome $E[O]$. However, $E[O]$ cannot be determined directly because it depends on the unknown Q and $\mathbf{C}$. Instead, it is estimated from sample data, using an estimator $\hat{E}[O]$. The sign of $\hat{E}[O]$ is then interpreted as follows. An $\hat{E}[O] < 0$ corresponds to inferring a preference for ranker $\mathbf{l}_1$, $\hat{E}[O] = 0$ is interpreted as a tie, and $\hat{E}[O] > 0$ is interpreted as a preference for ranker $\mathbf{l}_2$.

The simplest estimator of an expected value is the mean computed from a sample of i.i.d. observations of that value. Thus, the expected outcome can be estimated by the mean of observed outcomes $\hat{E}[O] = \frac{1}{n}\sum_{i=0}^{n} o_i$. Previous work did not formulate estimated interleaved comparison outcomes in terms of a probabilistic framework as done here. However, we show below that a commonly used previous estimator is equivalent to the sample mean. In [Chapelle et al. 2012] the following estimator is formulated:

$$\hat{E}_{wins} = \frac{wins(\mathbf{l}_2) + \frac{1}{2}ties(\mathbf{l}_{1,2})}{wins(\mathbf{l}_2) + wins(\mathbf{l}_1) + ties(\mathbf{l}_{1,2})} - 0.5. \tag{2}$$

Here, $wins(\mathbf{l}_i)$ denotes the number of samples for which $\mathbf{l}_i$ won the comparison, and $ties(\cdot)$ denotes the number of samples for which the two competing rankers tied. The following theorem states that this estimator is equal to the rescaled sampled mean.

THEOREM 4.1. *The estimator in Eq. 2 is equal to two times the sample mean.*

PROOF. See Appendix A. □

Clearly, this theorem implies that Eq. 2 always has the same sign as the sample mean, and thus the same preferences will be inferred.

As described in §2, alternative estimators have been proposed and investigated in [Chapelle et al. 2012; Radlinski and Craswell 2010; Yue et al. 2010]. Typically, these alternatives are designed to converge faster at the expense of obtaining biased estimates. This introduces a bias-variance trade-off; a formal analysis of these is beyond the scope of this article.

4.2. Definitions of Fidelity, Soundness, and Efficiency

Based on the probabilistic framework introduced in the previous subsection, we define our criteria for analyzing interleaved comparison methods: *fidelity*, *soundness*, and *efficiency*. These criteria reflect what interleaved comparison outcomes measure, whether an estimator of that outcome is statistically sound, and how efficiently it uses data samples. These assessment criteria are not intended to be complete, but are considered minimal requirements. Nevertheless, they enable a more systematic analysis of interleaved comparison methods than was previously attempted.

Our first criterion, fidelity, concerns whether the interleaved comparison method measures the right quantity, i.e., if $E[O|q]$ properly corresponds to the true quality difference between $\mathbf{l}_1$ and $\mathbf{l}_2$ in terms of how they rank relevant documents for a given q. Our definition uses the following concepts:

— $random\_clicks$ indicates that, for a given query, clicks are uniformly random, i.e., all documents at all ranks are equally likely to be clicked:

$$random\_clicks(q) \Leftrightarrow \forall d_{i,j} \in \mathbf{l}, P(c[r(d_i, \mathbf{l})]|q) = P(c[r(d_j, \mathbf{l})]|q),$$

where $r(d_i, \mathbf{l})$ is the rank of a given document d_i in result list $\mathbf{l}$ and $P(c[r(d_i, \mathbf{l})]|q)$ is the probability of a click at the rank at which document d_i is displayed.

— $correlated\_clicks(q)$ indicates positive correlation between clicks and document relevance:

$$correlated\_clicks(q) \Leftrightarrow \forall r \in ranks(\mathbf{l}), P(c[r]|rel(\mathbf{l}[r], q)) > P(c[r]|\neg rel(\mathbf{l}[r], q)),$$

where r is a rank in the interleaved list $\mathbf{l}$, $P(c[r]|rel(\mathbf{l}[r], q)$ is the probability of a click at r given that the document at r is relevant for the query. This means that, for a given query and at equal ranks, a relevant document is more likely to be clicked than a non-relevant one.

— $pareto\_dominates$ indicates that ranker $\mathbf{l}_1$ pareto dominates $\mathbf{l}_2$ for query q:

$$pareto\_dominates(\mathbf{l}_1, \mathbf{l}_2, q) \Leftrightarrow$$
$$\forall d \in rel(\mathbf{l}_1 \cup \mathbf{l}_2), r(d, \mathbf{l}_1) \geq r(d, \mathbf{l}_2) \wedge \exists d \in rel(\mathbf{l}_1 \cup \mathbf{l}_2), r(d, \mathbf{l}_1) > r(d, \mathbf{l}_2).$$

Here, $rel(\cdot)$ denotes the set of relevant documents in a given document set, and $r(d, \mathbf{l}_i)$ denotes the rank of document d according to ranker $\mathbf{l}_i$. Thus, one ranker Pareto dominates another in terms of how it ranks relevant documents if and only if it ranks all relevant documents at least as high as, and at least one relevant document higher than, the other ranker.

Definition 4.2 (*Fidelity*). An interleaved comparison method exhibits *fidelity* if,

(1) under random clicks, the rankers tie in expectation over clicks, i.e.,
$$\forall q(random\_clicks(q) \Rightarrow E[O|q] = 0),$$
(2) under correlated clicks, ranker $\mathbf{l}_2$ is preferred if it Pareto dominates $\mathbf{l}_1$:
$$\forall q(pareto\_dominates(\mathbf{l}_2, \mathbf{l}_1, q) \Rightarrow E[O|q] > 0).$$

We formulate fidelity in terms of the expected outcome for a given q because, in practice, a ranking function can be preferred for some rankers and not for others. We consider the expectation over some

population of queries in our definition of soundness below. In addition, we formulate condition (2) in terms of detecting a preference for $\mathbf{l}_2$. This is without loss of generality, as switching $\mathbf{l}_1$ and $\mathbf{l}_2$ results in a sign change of $E[O|q]$.

The first condition of our definition of fidelity has been previously proposed in [Radlinski et al. 2008b] and [Chapelle et al. 2012], and was used to analyze BI. A method that violates (1) is problematic because noise in click feedback can affect the outcome inferred by such a method. However, this condition is not sufficient for assessing interleaved comparison methods because a method that picks a preferred ranker at random would satisfy it, but cannot effectively infer preferences between rankers.

We add the second condition to require that an interleaved comparison method prefers a ranker that ranks relevant documents higher than its competitor. A method that violates (2) is problematic because it may fail to detect quality differences between rankers. This condition includes the assumption that clicks are positively correlated with relevance and rank. This assumption, which is implicit in previous definitions of interleaved comparison methods, is a minimal requirement for using clicks for evaluation.

Our definition of fidelity is stated in terms of binary relevance, as opposed to graded relevance, because requirements about how ranks of documents with different relevance grades should be weighted depend on the context in which an IR system is used (e.g., is a ranking with one highly relevant document better than one with three moderately relevant documents?). In addition, our definition imposes no preferences on rankings for which none dominates the other (e.g., one ranking placed relevant documents at ranks 1 and 7, the other places the same documents at ranks 3 and 4–which is better again depends on the search setting).

Because it is based on Pareto dominance, the second condition of our definition imposes only a partial ordering on ranked lists. This does not mean that interleaved comparison methods cannot infer preferences in cases where neither ranker dominates the other. It only requires that the correct direction of preference is detected when we know what that correct direction should be. This partial ordering is stronger than the explicit requirements posed in previous work, with a minimal set of additional assumptions. Note that in past and present experimental evaluations, stronger assumptions are implicitly made by using NDCG as a performance measure. Evaluation against NDCG differences implies that Pareto dominant rankers are preferred and also makes assumptions on how changes at different ranks and for documents of different relevance grades should be traded off. We address this relationship between our assumptions and NDCG in more detail at the end of §7.2.

In contrast to fidelity, which focused on outcomes for individual observations, our second criterion focuses on the characteristics of interleaved comparison methods when estimating comparison outcomes from sample data (of size n). Soundness concerns whether an interleaved comparison method's estimates of $E[O]$ are statistically sound.

Definition 4.3 (*Soundness*). An interleaved comparison method exhibits *soundness* for a given definition of O if its corresponding $\hat{E}[O]$ computed from sample data is an unbiased and consistent estimator of $E[O]$.

An estimator is *unbiased* if its expected value is equal to $E[O]$ [Halmos 1946]. It is *consistent* if it converges with probability 1 to $E[O]$ in the limit as $n \to \infty$ [Lehmann 1999]. A trivial example of an unbiased and consistent estimator of the expected value of a random variable X distributed according to some distribution $P(X)$ is the mean of samples drawn i.i.d. from $P(X)$.

Soundness has not been explicitly addressed in previous work on interleaved comparison methods. However, as shown above (§4.1, Theorem 4.1) a typical estimator proposed in previous work can be reduced to the sample mean, which is trivially sound. Soundness is more difficult to establish for some variants of our PI method introduced in §5, because they ignore parts of observed samples, marginalizing over known parts of the distribution in order to reduce variance. We prove in §5 that these variants preserve soundness.

Note that methods can perform well in practice in many cases even if they are biased, because there usually is a trade-off between bias and variance. However, all else being equal, an unbiased estimator provides more accurate estimates.

The third criterion, efficiency, concerns the amount of sample data a method requires to make reliable preference decisions.

Definition 4.4 (*Efficiency*). Let $\hat{E}_1[O]$, $\hat{E}_2[O]$ be two estimators of expected interleaved comparison outcomes $E[O]$. $\hat{E}_1[O]$ is a more *efficient* estimator of $E[O]$ than $\hat{E}_2[O]$ if $\hat{E}_1[O]$ Pareto dominates $\hat{E}_2[O]$ in terms of accuracy for a given sample size, i.e., $\hat{E}_1[O]$ is more efficient than $\hat{E}_2[O]$ if and only if

$$\forall n(P(sign(\hat{E}_1^n[O]) = sign(E[O])) \geq P(sign(\hat{E}_2^n[O]) = sign(E[O])))\wedge$$
$$\exists n(P(sign(\hat{E}_1^n[O]) = sign(E[O])) > P(sign(\hat{E}_2^n[O]) = sign(E[O]))),$$

where $\hat{E}_i^n[O]$ is the outcome estimated by $\hat{E}_i$ given sample data of size n.

Some interleaving methods may be more efficient than others in specific scenarios (e.g., known-item search [He et al. 2009]). However, more generally, efficiency is affected by the variance of comparison outcomes under a comparison method, and trends in efficiency can be observed when applying these methods to a large number of ranker comparisons. Here, we assess efficiency of interleaved comparison methods experimentally, on a large number of ranker comparisons under various conditions (e.g., noise in user feedback) in §7.

Efficiency (also called *cost* in [He et al. 2009]), has been previously proposed as an assessment criterion, and has been investigated experimentally on synthetic data [He et al. 2009] and on large-scale comparisons of individual ranker pairs in real-life web search traffic [Chapelle et al. 2012].

In addition to improving efficiency by reducing variance, subsequent interleaved comparisons can be made more efficient by reusing historical data. For methods that do not reuse historical data, the required amount of live data is necessarily linear in the number of ranker pairs to be compared. A key result of this article is that this requirement can be made sub-linear by reusing historical data. In the rest of this section, we include an analysis in terms of whether historical data reuse and the resulting increase in efficiency is possible for existing methods. Throughout this article, we assume that historical data is collected using an interleaved comparison method for earlier comparisons of other ranker pairs. Given such data, historical data reuse is most beneficial when a historical estimator exhibits fidelity, soundness and efficiency.

Below, we analyze the fidelity, soundness, and efficiency of all existing interleaved comparison methods, balanced interleave §4.3, team draft §4.4, and document constraints §4.5.

4.3. Balanced Interleave

Fidelity. BI was previously analyzed in [Radlinski et al. 2008b] and [Chapelle et al. 2012]. The method was shown to violate requirement (1) of fidelity. Here, we extend this argument, and provide example cases in which this violation of requirement (1) is particularly problematic. The identified problem is illustrated in Figure 2. Given $\mathbf{l}_1$ and $\mathbf{l}_2$ as shown, two interleaved lists can result from interleaving. The first is identical to $\mathbf{l}_1$, the second switches documents d_1 and d_2. Consider a user that randomly clicks on one of the result documents, so that each document is equally likely to be clicked. Because d_1 is ranked higher by $\mathbf{l}_1$ than by $\mathbf{l}_2$, $\mathbf{l}_1$ wins the comparison for clicks on d_1. However, $\mathbf{l}_2$ wins in all other cases, which means that it wins in expectation over possible interleaved lists and clicks. This argument can easily be extended to all possible click configurations using truth tables.

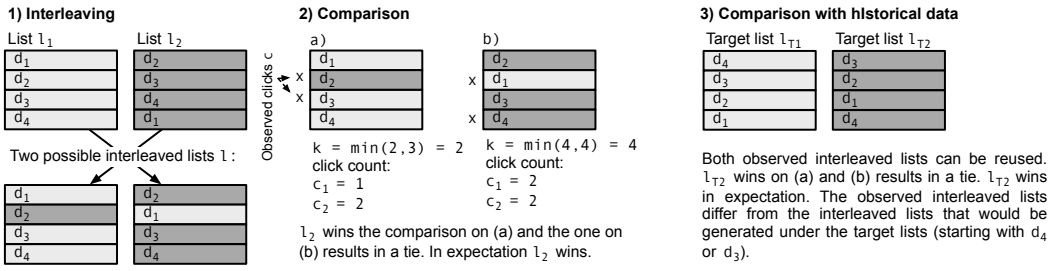


Fig. 2. Interleaving (1) and comparison with *balanced interleave* using live data (2) and historical data (3).

The demonstrated violation of fidelity condition (1) occurs whenever one original list ranks more documents higher than the other.[3] In practice, it is possible that the direction of such ranking changes can be approximately balanced between rankers when a large number of queries are considered. However, this is unlikely in settings where the compared lists are systematically similar to each other. For example, re-ranking approaches such as [Xue et al. 2004] combine two or more ranking features. Imagine two instances of such an algorithm, where one places a slightly higher weight on one of the features than the other instance. The two rankings will be similar, except for individual documents with specific feature values, which will be boosted to higher ranks. If users were to only click a single document, the new ranker would win BI comparisons for clicks on all boosted documents (as it ranks them higher), and lose for clicks on all other documents below the first boosted document (as these are in the original order and necessarily ranked lower by the new ranker). Thus, under random clicks, the direction of preference would be determined solely by the number and absolute rank differences of boosted documents. A similar effect (in the opposite direction) would be observed for algorithms that remove or demote documents, e.g., in (near-)duplicate detection [Radlinski et al. 2011].

In addition, BI violates condition (2) of fidelity when more than one document is relevant. The reason is that only the lowest-ranked clicked document (k) is taken into account to calculate click score differences. If for both original lists the lowest-ranked clicked document has the same rank, the comparison results in a tie, even if large ranking differences exist for higher-ranked documents. Condition (2) is not violated when only one relevant document is present.

Soundness. Soundness of BI has not been explicitly investigated in previous work. However, as we showed in the previous sections, it is trivially sound because its estimator can be reduced to the sample mean (§4.1).

Efficiency. The efficiency of BI was found to be sufficient for practical applications in [Chapelle et al. 2012]. For example, to detect preferences with high confidence for ranker changes that are typical for incremental improvements at commercial search engines, several thousand impressions were required.

Data Reuse. Reusing historical data to compare new target rankers using BI is possible in principle. Given historical result lists and clicks, and a new pair of target rankers, comparison outcomes can be computed as under live data, following Algorithm 1, lines 13–17. This means that observed clicks would be projected onto the new target lists to determine k, the rank at which the lowest click would occur for the target rankers. Then, the number of clicks on the top$k$ results can be counted for the target rankers as if they had been used in a live comparison. However, such straightforward data reuse would severely bias the inferred comparison outcomes. In particular, the target ranker that is more similar to those under which the historical data was originally collected will be likely to be preferred when data is reused. It is not clear whether and how the differences between observed interleaved lists and "correct" interleaved lists for the new target rankers could be compensated for.

[3]This occurs frequently. For a simple example, consider rankers that produce identical rankings, except that one ranker moves a single document up or down by more than one rank.

4.4. Team-draft

Fidelity. TD was designed to address fidelity requirement (1) [Radlinski et al. 2008b]. This is achieved by using assignments as described in the previous section (cf., §3). That the requirement is fulfilled can be seen as follows. Each ranker is assigned the same number of documents in the interleaved result list in expectation (by design of the interleaving process). Rankers get credit for clicks if and only if they are assigned to them. Thus, if clicks are randomly distributed, each ranker is credited with the same number of clicks in expectation.

However, TD violates fidelity requirement (2) when the original lists are similar. Figure 3 illustrates such a case. Consider the original lists $\mathbf{l}_1$ and $\mathbf{l}_2$. Also, assume that d_3 is the only relevant document, and is therefore more likely to be clicked than other documents. We can see that $\mathbf{l}_2$ ranks d_3 higher than $\mathbf{l}_1$ (i.e., $pareto\_dominates(\mathbf{l}_2, \mathbf{l}_1, q) = true$; cf. §4.2), and therefore $\mathbf{l}_2$ should win the comparison. When TD is applied, four possible interleaved lists can be generated, as in the figure. All these possible interleaved lists place document d_3 at the same rank. In two interleaved lists, d_3 is contributed by $\mathbf{l}_1$, and in two cases it is contributed by $\mathbf{l}_2$. Thus, in expectation, both lists obtain the same number of clicks for this document, yielding a tie. Thus, we can see the method fails to detect the preference for $\mathbf{l}_2$. Note that in the example shown, the lists would also tie if d_4 was the only relevant document, while in cases where only d_2 is relevant, a preference for $\mathbf{l}_2$ would be detected.

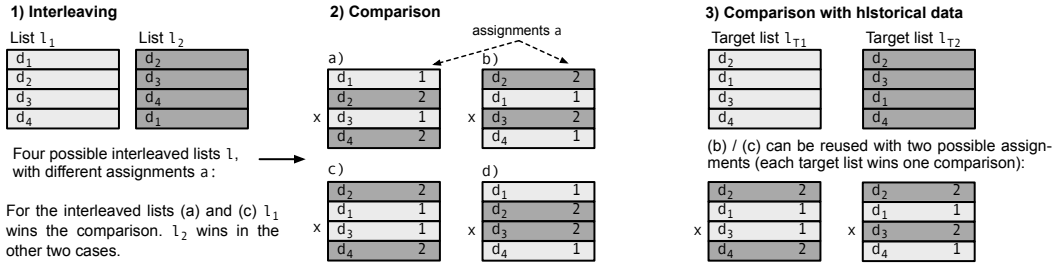


Fig. 3. Interleaving (1) and comparison with *team draft* using live data (2) and historical data (3).

In practice, TD's violation of requirement (2) can result in insensitivity to some small ranking changes. As shown above, some changes by one rank may result in a difference being detected while others are not detected. This is expected to be problematic in cases where a new ranking-function affects a large number of queries by a small amount, i.e., documents are moved up or down by one rank, as only some of these changes would be detected. In addition, it can result in a loss of efficiency, because, when some ranking differences are not detected, more data is required to reliably detect differences between rankers.

Soundness. As with BI, the soundness of TD has not been analyzed in practice. However, as above, typical estimators produce estimates that can easily be rescaled to the sample mean, which is consistent and unbiased (cf., Theorem 4.1).

Efficiency. As with BI, the efficiency of TD was found to be sufficient for practical applications in web and literature search [Chapelle et al. 2012]. The amount of sample data required was within the same order of magnitude as for BI, with TD requiring slightly fewer samples in some cases and vice versa in others. In an analysis based on synthetic data, TD was found to be less efficient than BI on simulated known-item search task (i.e., searches with only one relevant document) [He et al. 2009]. This result is likely due to TD's lack of sensitivity under small ranking changes.

Data Reuse. Reusing historical data under TD is difficult due to the use of assignments. One option is to use only observed interleaved lists that could have been constructed under the target rankers for the historical query. If the observed interleaved lists can be generated with the target rankers, the assignment under which this would be possible can be used to compute comparison outcomes.

If several assignments are possible, one can be selected at random, or outcomes for all possible assignments can be averaged. An example is shown in Figure 3. Given the observed interleaved lists shown in step (2), and two target rankers $\mathbf{l}_{T1}$ and $\mathbf{l}_{T2}$, the observed document rankings (b) and (c) could be reused, as they are identical to lists that can be produced under the target rankers. However, this approach is extremely inefficient. If we were to obtain historical data under a ranker that presents uniformly random permutations of candidate documents to users, of the $d!$ possible orderings of d documents that could be observed, only an expected $2^{\frac{d}{2}}$ could actually be used for a particular pair of target rankers. Even for a shallow pool of 10 candidate documents per query, these figures differ by five orders of magnitude. In typical settings, where candidate pools can be large, a prohibitively large amount of data would have to be collected and only a tiny fraction of it could be reused. Thus, the effectiveness of applying the team-draft method to historical data depends on the similarity of the document lists under the original and target rankers, but is generally expected to be very low.

Even in cases where data reuse is possible because ranker pairs are similar, TD may violate requirement (2) of fidelity under historical data. An example that is analogous to that under live data is shown in Figure 3. Here, the lists would tie in the case that document d_3 is relevant, even though $\mathbf{l}_{T2}$ Pareto dominates $\mathbf{l}_{T1}$. In addition, reusing historical data under TD affects soundness because not all interleaved lists that are possible under the target rankers may be found in observed historical data. For example, in Figure 3, only interleaved lists that place d_2 at the top rank match the observed data and not all possible assignments can be observed. In this example, clicks on d_2 would result in wins for $\mathbf{l}_{T2}$, although the target lists place this document at the same rank. This problem can be considered a form of sampling bias, but it is not clear how it could be corrected for.

4.5. Document Constraints

Fidelity. The DC method has not been previously analyzed in terms of fidelity. Here, we find that DC violates both requirements (1) and (2). An example that violates both requirements is provided in Figure 4. The original lists $\mathbf{l}_1$ and $\mathbf{l}_2$, and the possible interleaved lists are shown. In the example, the first condition of fidelity is violated, as $\mathbf{l}_2$ wins in expectation over random clicks. The reason is that $\mathbf{l}_2$ is less similar to the possible interleaved lists and can therefore violate fewer constraints inferred from clicks on these lists. For example, consider the possible constraints that d_1 (ranked higher by $\mathbf{l}_1$) and d_4 (ranked higher by $\mathbf{l}_2$) can be involved in. Clicks on the possible interleaved lists could result in 14 constraints that prefer other documents over d_4, but in 24 constraints that prefer other documents over d_1. As a result, $\mathbf{l}_1$ violates more constraints in expectation, and $\mathbf{l}_2$ wins the comparison in expectation under random clicks.

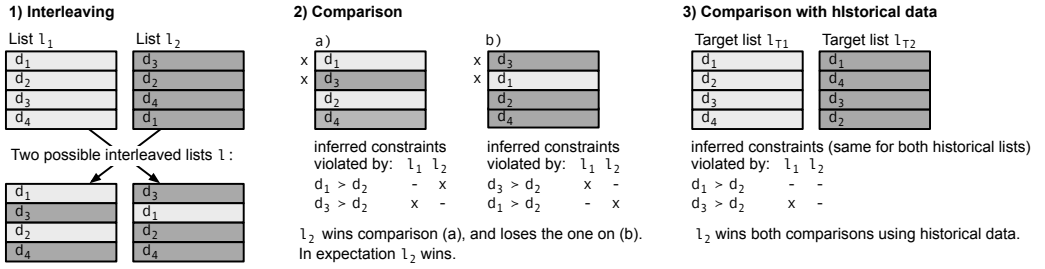


Fig. 4. Interleaving (1) and comparison with *document constraints* using live data (2) and historical data (3).

The example above also violates requirement (2). Consider two relevant documents, d_1 and d_3 are clicked by the user. In this case, $\mathbf{l}_1$ should win the comparison as it Pareto dominates $\mathbf{l}_2$. However, for the interleaved lists generated for this case, each original list violates exactly one constraint, which results in a tie. The reason for the violation of both requirements of fidelity is that the number of

requirements each list and each document is involved in is not controlled for. It is not clear whether and how controlling for the number of constraints is possible when making comparisons using DC.

Soundness. As with BI and TD, soundness of DC estimator can be easily established, as it is based on the sample mean (Theorem 4.1).

Efficiency. The efficiency of DC was previously studied on synthetic data [He et al. 2009]. On the investigated cases (known-item search, easy and hard high-recall tasks with perfect click feedback), DC was demonstrated to be more efficient than BC and TD. DC has not been evaluated in a real-live application.

Data Reuse. Finally, we consider applying DC to historical data. Doing this is in principle possible, because constraints inferred from previously observed lists can easily be compared to new target rankers. However, the fidelity of outcomes cannot be guaranteed (as under live data). An example is shown in part (3) of Figure 4. Two new target lists are compared using the historical data collected in earlier comparisons. Again, two documents are relevant, d_1 and d_3. The target lists place these relevant documents at the same ranks. However, $\mathbf{l}_1$ violates more constraints inferred from the historical data than $\mathbf{l}_2$, so that a preference for $\mathbf{l}_2$ is detected using either historical observation. As with live data, the number of constraints that can be violated by each original list is not controlled for. Depending on how the historical result list was constructed, this can lead to outcomes that are biased similarly or more strongly than under live data.

5. PROBABILISTIC INTERLEAVE METHODS

In this section, we present a new interleaved comparison method called *probabilistic interleave* (PI). We first give an overview of the algorithm and provide a naive estimator of comparison outcomes (§5.1). We show that this approach exhibits fidelity and soundness, but that its efficiency is expected to be low. Then, we introduce two extensions of PI, that increase efficiency while maintaining fidelity and soundness. The first extension, PI-MA, is based on marginalizing over possible comparison outcomes for observed samples (§5.2). The second extension, PI-MA-IS, shows how historical data can be reused to further increase efficiency (§5.3).

5.1. Probabilistic Interleave

We propose a probabilistic form of interleaving in which the interleaved document list $\mathbf{l}$ is constructed, not from fixed lists $\mathbf{l}_1$ and $\mathbf{l}_2$ for a given query q, but from *softmax* functions $s(\mathbf{l}_1)$ and $s(\mathbf{l}_2)$ that transform these lists into probability distributions over documents. The use of softmax functions is key to our approach, as it ensures that every document has a non-zero probability of being selected by each ranker and for each rank of the interleaved result list. As a result, the distribution of credit accumulated for clicks is smoothed, based on the relative rank of the document in the original result lists. If both rankers place a given document at the same rank, then the corresponding softmax functions have the same probability of selecting it and thus they accumulate the same number of clicks in expectation. More importantly, rankers that put a given document at similar ranks receive similar credit in expectation. The difference between these expectations reflects the magnitude of the difference between the two rankings. In this way, the method becomes sensitive to even small differences between rankings and can accurately estimate the magnitude of such differences.

The softmax functions $s(\mathbf{l}_1)$ and $s(\mathbf{l}_2)$ for given ranked lists $\mathbf{l}_1$ and $\mathbf{l}_2$ are generated by applying a monotonically decreasing function over document ranks, so that documents at higher ranks are assigned higher probabilities. Many softmax functions are possible, including the sigmoid or normalized exponential functions typically used in neural networks and reinforcement learning [Lippmann 2002; Sutton and Barto 1998]. Here, we use a function in which the probability of selecting a document is inversely proportional to a power of the rank $r_i(d)$ of a document d in list $\mathbf{l}_i$:

$$s(\mathbf{l}_i) := P_i(d) = \frac{\frac{1}{r_i(d)^\tau}}{\sum_{d' \in D} \frac{1}{r_i(d')^\tau}}, \tag{3}$$

where D is the set of all ranked documents, including d. The denominator applies a normalization to make probabilities sum to 1. Because this softmax function has a steep decay at top ranks, it is suitable for an IR setting in which correctly ranking the top documents is the most important. It also has a slow decay at lower ranks, preventing underflow in calculations. The parameter τ controls how quickly selection probabilities decay as rank decreases, similar to the Boltzmann temperature in the normalized exponential function [Sutton and Barto 1998]. In relation to traditional IR metrics, τ can be interpreted as a discount factor that controls the focus on top ranked documents, similarly to e.g., the rank discount in NDCG [Järvelin and Kekäläinen 2002]. In our experiments, we use a default of $\tau = 3$ and explore possible choices of τ and their relation to traditional evaluation metrics (§7.2).

After constructing $s(\mathbf{l}_1)$ and $s(\mathbf{l}_2)$, $\mathbf{l}$ is generated similarly to the team draft method (cf., Algorithm 4). However, instead of randomizing the ranker to contribute the next document per pair, one of the softmax functions is randomly selected at each rank (line 7). Doing so is mathematically convenient, as the only component that changes at each rank is the distribution over documents. More importantly, this change ensures fidelity, as will be shown shortly. During interleaving, the system records which softmax function was selected to contribute the next document in assignment $\mathbf{a}$ (line 9). Then, a document is randomly sampled without replacement from the selected softmax function (line 10) and added to the interleaved list (line 11). The document is also removed from the non-sampled softmax function, and this softmax function is renormalized (line 12). This process repeats until $\mathbf{l}$ has the desired length.

ALGORITHM 4: Probabilistic Interleave.

1: **Input**: $\mathbf{l}_1, \mathbf{l}_2, \tau$
2: $\mathbf{l} \leftarrow []$
3: $\mathbf{a} \leftarrow []$
4: **for** $i \in (1, 2)$ **do**
5: initialize $s(\mathbf{l}_i)$ using Eq. 3
6: **while** $(\exists r : \mathbf{l}_1[r] \notin \mathbf{l}) \vee (\exists r : \mathbf{l}_2[r] \notin \mathbf{l})$ **do**
7: $a \leftarrow 1$ **if** $random\_bit()$ **else** 2
8: $\bar{a} \leftarrow 2$ **if** $a = 1$ **else** 1
9: $append(\mathbf{a}, a)$
10: $d_{next} \leftarrow sample\_without\_replacement(s(\mathbf{l}_a))$
11: $append(\mathbf{l}, d_{next})$
12: $remove\_and\_renormalize(s(\mathbf{l}_{\bar{a}}), d_{next})$
// present $\mathbf{l}$ *to user and observe clicks* $\mathbf{c}$
13: compute o, e.g., using Eqs. 6–9
14: **return** o

After generating an interleaved list using the probabilistic interleave process described above, and observing user clicks, comparison outcomes can be computed as under the team draft methods, i.e., by counting the clicks c_1 and c_2 assigned to each softmax function and returning $o = (-1$ **if** $c_1 > c_2$ **else** 1 **if** $c_1 < c_2$ **else** $0)$.

PI exhibits fidelity for the following reasons. To verify condition (1), consider that each softmax function is assigned the same number of documents to each rank in expectation (by design of the interleaving process). Clicks are credited to the assigned softmax function only, which means that in expectation the softmax functions tie under random user clicks. To verify condition (2), consider that each softmax function has a non-zero probability of contributing each document to each rank of the interleaved list. This probability is strictly higher for documents that are ranked higher in the result list underlying the softmax function, because the softmax functions are monotonically decreasing and depend on the document rank only. The softmax function that assigns a higher probability to a particular document d_x has a higher probability of contributing that document to $\mathbf{l}$, which gives it a higher probability of being assigned clicks on d_x. Thus, in expectation, the softmax function that ranks relevant documents higher obtains more clicks, and therefore has higher expected outcomes if

clicks are correlated with relevance. In cases where $\mathbf{l}_1$ and $\mathbf{l}_2$ place d_x at the same rank, the softmax functions assign the same probability to that document, because the softmax functions have the same shape. Thus, for documents placed at the same rank, expected clicks tie in expectation.

An issue related to fidelity that has not been addressed previously is what the magnitude of the differences in outcomes should be for ranking changes at different ranks. For example, consider two ranker pairs that rerank a result list with one relevant document. In one pair, the better ranker moves the relevant document from rank 3 to rank 1. In the other pair, the better ranker moves the relevant document from rank 5 to 7. Should the detected magnitude of the ranker differences be the same, or should the first change have a stronger impact (and how much stronger)? In our definition of fidelity, this question is left open, as it requires additional assumptions about user expectations and behavior. In PI, this magnitude can be determined by the choice of softmax function. For example, when using the formulation in Eq. 3, rank discounts decrease as $\tau \to 0$. Rank discounts increase as $\tau \to \infty$, and probabilistic interleaving with deterministic ranking functions is the limiting case (this case is identical to changing team draft so that rankers are randomized per rank instead of per pair of ranks). Interpreted in this way, we see that PI defines a class of interleaved comparison metrics that can be adapted to different scenarios.

As discussed in §4.2, the simplest estimator of $E[O]$ is the mean of sample outcomes:

$$\hat{E}[O] = \frac{1}{n} \sum_{i=0}^{n} o_i. \tag{4}$$

Since the sample mean is unbiased and consistent, soundness is trivially established. A limitation of this naive estimator is that its efficiency is expected to be low. In comparison to existing interleaved comparison methods, additional noise is introduced by the higher amount of randomization when selecting softmax functions per rank, and by using softmax functions instead of selecting documents from the contributing lists deterministically. In the next sections, we show how probabilistic interleaving allows us to derive more efficient estimators while maintaining fidelity and soundness.

5.2. Probabilistic Comparisons with Marginalization

In the previous subsection, we described PI and showed that it has fidelity and soundness. In this section, we introduce a more efficient estimator, PI-MA, that is derived by exploiting known parts of the probabilistic interleaving process, and show that under this more efficient estimator fidelity and soundness are maintained.

To derive PI-MA, we start by modeling PI using the graphical model in Figure 1(b).[4] This allows us to rewrite Eq. 4 as:

$$\hat{E}[O] = \frac{1}{n} \sum_{i=0}^{n} o_i = \frac{1}{n} \sum_{i=1}^{n} \sum_{o \in O} o P(o | \mathbf{a}_i, \mathbf{c}_i, \mathbf{l}_i, q_i), \tag{5}$$

where $\mathbf{a}_i$, $\mathbf{c}_i$ and $\mathbf{l}_i$, and q_i are the observed assignment, clicks, interleaved list, and query for the i-th sample. This formulation is equivalent because o is deterministic given $\mathbf{a}$ and $\mathbf{c}$.

In Eq. 5, the expected outcome is estimated directly from the observed samples. However, the distributions for $\mathbf{A}$ and $\mathbf{L}$ are known given an observed q. As a result, we need not consider only the observed assignments. Instead, we can consider all possible assignments that could have co-occurred with each observed interleaved list $\mathbf{l}$, i.e., we can marginalize over all possible values of $\mathbf{A}$ for given $\mathbf{l}_i$ and q_i. This method reduces noise resulting from randomized assignments, making it more efficient than methods that directly use observed assignments. Marginalizing over $\mathbf{A}$ leads to the following

[4] In contrast to [Hofmann et al. 2011], we treat the outcome O as a random variable. This leads to an equivalent estimator that is more convenient for the proof below.

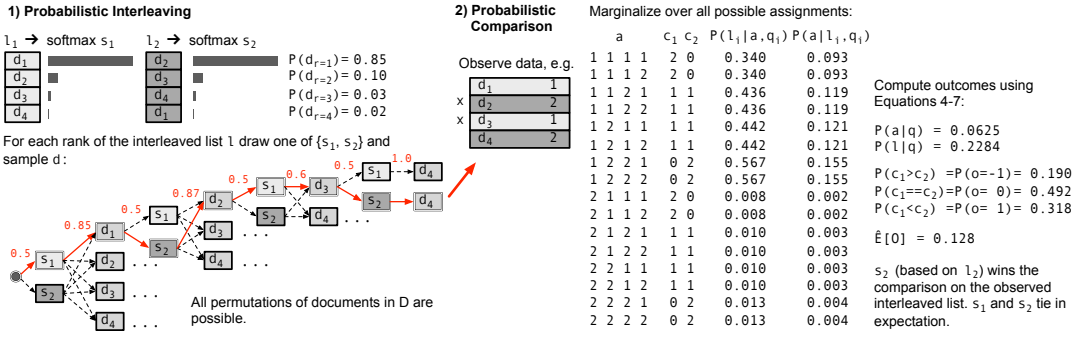


Fig. 5. Example probabilistic interleaving (1) and comparison (2) with marginalization over all possible assignments.

alternative estimator:

$$\hat{E}[O] = \frac{1}{n}\sum_{i=1}^{n}\sum_{\mathbf{a}\in\mathbf{A}}\sum_{o\in O} oP(o|\mathbf{a},\mathbf{c}_i)P(\mathbf{a}|\mathbf{l}_i,q_i). \tag{6}$$

The estimator in Eq. 6 marginalizes over all possible assignments that could have led to observing $\mathbf{l}$ by making use of the fact that this distribution is fully known. The probability of an assignment given observed lists and queries is computed using Bayes' rule:

$$P(\mathbf{a}|\mathbf{l},q) = \frac{P(\mathbf{l}|\mathbf{a},q)P(\mathbf{a}|q)}{P(\mathbf{l}|q)}. \tag{7}$$

Note that $P(\mathbf{a}|q) = P(\mathbf{a}) = \frac{1}{|\mathbf{A}|}$, because $\mathbf{a}$ and q are independent. $P(\mathbf{l}|\mathbf{a},q)$ is fully specified by the probabilistic interleaving process and can be obtained using:

$$P(\mathbf{l}|\mathbf{a},q) = P(\mathbf{l},\mathbf{a}|q)P(\mathbf{a}|q) = \prod_{r=1}^{len(\mathbf{l})} P(\mathbf{l}[r] \mid \mathbf{a}[r],\mathbf{l}[1,r-1],q)P(\mathbf{a}|q). \tag{8}$$

Here, $len(\mathbf{l})$ is the length of the document list, $\mathbf{l}[r]$ denotes the document placed at rank r in the interleaved list $\mathbf{l}$, $\mathbf{l}[1,r-1]$ contains the documents added to the list before rank r, and $\mathbf{a}[r]$ denotes the assignment at rank r, i.e., which list contributed the document at r. Finally, $P(\mathbf{l}|q)$ can be computed as follows:

$$P(\mathbf{l}|q) = \sum_{\mathbf{a}\in\mathbf{A}} P(\mathbf{l}|\mathbf{a},q)P(\mathbf{a}). \tag{9}$$

An example comparison using PI-MA is shown in Figure 5. In it, an interleaved list is generated using the process shown in Algorithm 4, in this case $\mathbf{l} = (d_1, d_2, d_3, d_4)$ (as marked in red). After observing clicks on d_2 and d_3, the naive estimator detects a tie ($o = 0$), as both original lists obtain 1 click. In contrast, the probabilistic comparison shown in step 2 marginalizes over all possible assignments, and detects a preference for $\mathbf{l}_2$.

Next, we establish the soundness of PI-MA by showing that it is an unbiased and consistent estimator of our target outcome $E[O]$. Because PI exhibits fidelity (cf. §5.1), showing that PI-MA is a consistent and unbiased estimator of the same quantity establishes fidelity as well.

THEOREM 5.1. *The following estimator is unbiased and consistent given samples from an interleaving experiment conducted according to the graphical model in Figure 1(b) (Eq. 6):*

$$\hat{E}[O] = \frac{1}{n}\sum_{i=1}^{n}\sum_{\mathbf{a}\in\mathbf{A}}\sum_{o\in O} oP(o|\mathbf{a},\mathbf{c}_i)P(\mathbf{a}|\mathbf{l}_i,q_i).$$

PROOF. See Appendix B. □

Theorem 5.1 establishes soundness for PI-MA (Eq. 6), which is designed to be more efficient than the naive estimator (Eq. 5). We report on an empirical evaluation of the efficiency of these estimators in §7.2.

5.3. Probabilistic Comparisons with Historical Data

In the previous subsections, we derived two estimators for inferring preferences between rankers using live data. We now turn to the historical data setting, where previously collected data (e.g., from an earlier comparison of different rankers) is used to compare a new ranker pair. As shown above (cf., §4), none of the existing interleaved comparison methods can reuse data while maintaining fidelity and soundness. Here, we show that this is possible for a new estimator, PI-MA-IS, that we derive from PI-MA.

In principle, PI-MA, as defined in Eq. 6 could be directly applied to historical data. Note that, for a ranker pair that re-ranks the same set of candidate documents D as the method used to collect the historical data, $P(\mathbf{a}|\mathbf{l}, q)$ is known and non-zero for all possible assignments. Such an application of the method designed for live data could be efficient because it marginalizes over possible assignments. However, the soundness of the estimator designed for live data would be violated because the use of historical data would introduce bias, i.e., the expected outcome under historical data would not necessarily equal the expected value under live data. Similarly, the estimator would not be consistent.

To see why bias and inconsistency would be introduced, consider two pairs of rankers. Pair S is the source ranker pair, which was compared in a live experiment using interleaved result lists from which the comparison outcome was computed using the resulting clicks. All data from this past experiment were recorded, and we want to compare a new ranker pair T using this historical data. Observations for pair S occur under the original distribution P_S, while observations for pair T occur under the target distribution P_T. The difference between P_S and P_T is that the two ranker pairs result in different distributions over $\mathbf{L}$. For example, interleaved lists that place documents ranked highly by the rankers in S at the top are more likely under P_S, while they may be much less likely under P_T. Bias and inconsistency would be introduced if, e.g., one of the rankers in T would be more likely to win comparisons on lists that are more likely to be observed under P_S than under P_T.

Our goal is to estimate $E_T[O]$, the expected outcome of comparing T, given data from the earlier experiment of comparing S, by compensating for the difference between P_T and P_S. To derive an unbiased and consistent estimator, note that P_T and P_S can be seen as two different instantiations of the graphical model in Figure 1(b). Also note that both instantiations have the same event spaces (i.e., the same queries, lists, click and assignment vectors are possible), and, more importantly, only the distributions over $\mathbf{L}$ change for different ranker pairs. This means that the same result lists can be displayed, but with different probabilities. The distributions over $\mathbf{A}$ are the same under P_T and P_S by design of the interleaving process. Distributions over $\mathbf{C}$ (conditioned on $\mathbf{L}$) and Q are the same for different ranker pairs, because we assume that clicks and queries are drawn from the same static distribution, independently of the ranker pair used to generate the presented list.

A naive estimator of the expected outcome $E_T[O]$ from sample data observed under P_S can be obtained from the definition of the importance sampling estimator in Eq. 1 with $f(\mathbf{a}, \mathbf{c}) = \sum_{o \in O} oP(o|\mathbf{a}, \mathbf{c})$:

$$\hat{E}_T[O] = \frac{1}{n} \sum_{i=1}^{n} \sum_{o \in O} oP(o|\mathbf{a}_i, \mathbf{c}_i) \frac{P_T(\mathbf{a}_i, \mathbf{c}_i)}{P_S(\mathbf{a}_i, \mathbf{c}_i)}. \tag{10}$$

We refer to this estimator as PI-IS. It simply applies importance sampling to reweight observations by the ratio of their probability under the source and target distributions. Importance sampling has been shown to produce unbiased and consistent estimates of the expected outcome under the target distribution, $E_T[O]$, as long as P_S and P_T have the same event space, and P_S is non-zero for all events that have a non-zero probability under P_T (this is given by our definition of probabilistic interleaving, as long as the softmax functions under P_S are non-zero all documents that have non-zero

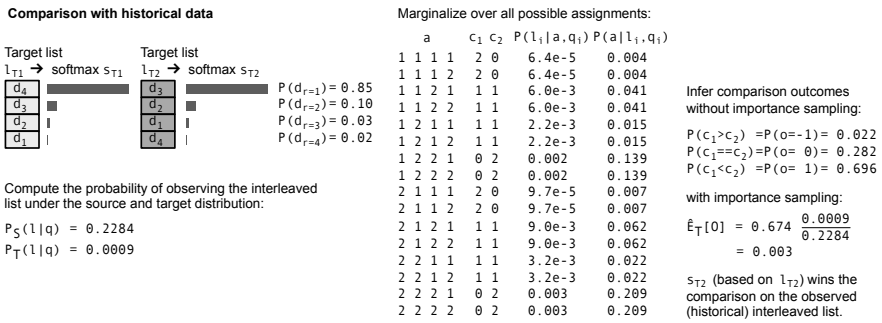


Fig. 6. Example probabilistic comparison with historical data. We assume observed historical data as shown in Figure 5 above.

probabilities under P_T) [MacKay 1998]. Although this estimator is unbiased and consistent, it is expected to be inefficient, because it merely reweights the original, noisy, estimates, which can lead to high overall variance.

To derive an efficient estimator of $E_T[O]$, we need to marginalize over all possible assignments, as in §5.2. Building on Eq. 10, we marginalize over the possible assignments (so the assignments $\mathbf{a}_i$ observed with the sample data are not used) and obtain the estimator PI-MA-IS:

$$\hat{E}_T[O] = \frac{1}{n}\sum_{i=1}^{n}\sum_{\mathbf{a}\in\mathbf{A}}\sum_{o\in O} oP(o|\mathbf{a},\mathbf{c}_i)P(\mathbf{a}|\mathbf{l}_i,q_i)\frac{P_T(\mathbf{l}_i|q_i)}{P_S(\mathbf{l}_i|q_i)}. \tag{11}$$

As in the previous section, $P(\mathbf{a}|\mathbf{l},q)$ is computed using Eq. 8, and $P(\mathbf{l}|q)$ is obtained from Eq. 9. An example is given in Figure 6. In this example, the target lists are very different from the original lists, which is reflected in the low probability of the observed interleaved list under the target distribution ($P_T(\mathbf{l}|q) = 0.0009$). Although $\mathbf{l}_{T2}$ performs much better on the observed list, the small importance weight results in only a small win for this target list.

The following theorem establishes the soundness of PI-MA-IS. By showing that Eq. 11 is an unbiased and consistent estimator of $E_T[O]$ under historical data, we also show that it maintains fidelity.

THEOREM 5.2. *The following estimator is unbiased given samples from an interleaving experiment conducted according to the graphical model in Figure 1(b) under P_S:*

$$\hat{E}_T[O] = \frac{1}{n}\sum_{i=1}^{n}\sum_{a\in A}\sum_{o\in O} oP(o|c_i,a)P(a|l_i,q_i)\frac{P_T(l_i|q_i)}{P_S(l_i|q_i)}.$$

PROOF. See Appendix C. □

The efficiency of PI-MA-IS depends on the similarity between P_S and P_T. It is easy to see that importance weights can become very large when there are large differences between these distributions, leading to high variance. As observed by Chen [2005], this variance can be quantified as the ratio between the variance of outcomes under the source distribution and under the target distribution. We empirically assess the efficiency of the estimator under a wide range of source and target distributions in (§7.3).

Note that PI-MA-IS does not depend on the assignments observed in the original data (cf., Eq. 11). This means that it can be applied not just to historical data collected using probabilistic interleaving, but to data collected under any arbitrary distribution, as long as the distribution over result lists is known and non-zero for all lists that are possible under the target distribution. This makes it possible to develop new sampling algorithms that can make interleaved comparisons even more efficient. For example, data could be sampled in a way that allows optimal comparisons of a set of more than two

rankers, or with the combined goal to maximize both the quality of the lists presented to users, and the reusability of the collected data. While doing so is beyond the scope of the current article, it is an important direction for future research.

6. EXPERIMENTS

We report on three sets of experiments. The first set is designed to verify our analytical results (§4) regarding the fidelity and soundness of interleaved comparison methods (§6.1). The second and third sets are designed to assess the efficiency of interleaved comparison methods under live data (§6.2) and under historical data (§6.3). All our experiments rely on a simulation framework that allows us to evaluate interleaved comparison methods on a large set of ranker pairs in a controlled setting without the risk of affecting users of a production system. In this section, we first give an overview of the simulation framework and its assumptions about user interactions. We then describe our data set and metrics. Finally, we detail the experimental procedures (§6.1–§6.3). Results of all experiments are provided in the next section (§7).

Our experiments are based on the simulation framework introduced in [Hofmann et al. 2011]. It combines learning to rank data sets and click models to simulate users' interactions with a retrieval system. This setup allows us to study interleaving methods under different conditions, e.g., varying amounts of data collected under different ranker pairs, without the risk of hurting the user experience in a production system.[5]

The simulation framework makes the following assumptions about user interactions. A user interaction consists of submitting a query to the system, examining up to 10 top-ranked documents of the returned result list, and clicking links to promising documents. Since we do not model query sessions, queries are independent of previous queries and previously shown result lists. Users inspect and click documents following the Dependent Click Model, which has been shown to accurately model user behavior in a web search setting [Guo et al. 2009]. They start with the top-ranked document and proceed down the list, clicking on promising documents (with probability $P(C|R)$, the probability of a click given the document's relevance level R) and, after viewing a document, deciding whether to stop (with stopping probability $P(S|R)$) or examine more documents. Click and stop probabilities are instantiated using the graded relevance assessments provided with the learning to rank data set. It is assumed that users are more likely to click on more relevant documents, based on the attractiveness of e.g., the document title and snippet. As argued in [Hofmann et al. 2011], the assumptions of the model are appropriate for comparing the performance of interleaved comparison methods, as they satisfy the assumptions of these methods.

Table II. Overview of the click models used in our experiments.

	click probabilities					*stop probabilities*				
relevance grade R	0	1	2	3	4	0	1	2	3	4
perfect	0.0	0.2	0.4	0.8	1.0	0.0	0.0	0.0	0.0	0.0
navigational	0.05	0.1	0.2	0.4	0.8	0.0	0.2	0.4	0.6	0.8
informational	0.4	0.6	0.7	0.8	0.9	0.1	0.2	0.3	0.4	0.5
almost random	0.4	0.45	0.5	0.55	0.6	0.5	0.5	0.5	0.5	0.5
random	0.5	0.5	0.5	0.5	0.5	0.0	0.0	0.0	0.0	0.0

We instantiate the click model in five different ways, to assess interleaved comparison methods under various levels of noise. The click models (for a data set annotated with 5 relevance levels) are shown in Table II. The *perfect click model* simulates a user who clicks on all highly relevant document ($R = 4$), and never clicks on non-relevant documents ($R = 0$). Click probabilities for intermediate relevance levels have a linear decay, except for a higher increase in click probability between relevance levels 2 and 3 (based on previous work that showed that grouping "good" documents with

[5]We do not consider the effects of limitations common to all interleaved comparison methods (e.g., bias in click behavior; see §2.1) as this has been addressed elsewhere [Hofmann et al. 2012a; Radlinski and Craswell 2010].

non-relevant documents is more effective than grouping them with relevant documents [Chapelle et al. 2009]). The stop probability for this click model is zero, meaning that there is no position bias (simulated users examine all top-10 results). The *navigational* click model simulates the focus on top-ranked and highly relevant results that are characteristic of navigational searches [Liu et al. 2006; Rose and Levinson 2004]. In comparison with the perfect click model, the navigational model results in fewer clicks on result documents, with a stronger focus on highly relevant and top-ranked results. Correspondingly, the *informational* click model captures the broader interests characteristic for informational searches [Liu et al. 2006; Rose and Levinson 2004]. In this model, the click and stop probabilities for lower relevance grades are more similar to those for highly relevant documents, resulting in more clicks, and more noisy click behavior than the previous models. As a lower bound on click reliability, we also include an *almost random* click model, with only a small linear decay in the click probabilities for different relevance grades. Finally, the *random* click model is one instantiation of a click model that satisfies the $random\_clicks$ predicate used in our formulation of fidelity condition (1). We use this last click model in our experiment to verify the fidelity of interleaved comparison methods.

In our first set of experiments, we generate synthetic result lists to ensure Pareto dominance between rankers. Details of the resulting synthetic rankers are described in §6.1. For the experiments described in §6.2–6.3, we use the MSLR-WEB30k Microsoft learning to rank data set.[6] As in [Hofmann et al. 2011], these experiments are run on the $18,919$ queries of the training set of fold 1 of this data set. The data set encodes relations between queries and candidate documents in 136 precomputed features, and provides (manual) relevance judgments on a 5-point scale (from 0 – "non-relevant" to 4 – "highly relevant"). We generate rankers from the individual features provided with the learning to rank data set. This means that our experiments simulate the task of comparing the effectiveness of individual features for retrieval using varying amounts of historical data, or a combination of historical and live data.

As specified in Definition 4.4, we compare the efficiency of rankers by comparing the accuracy they obtain after observing a sample of a given size. We measure accuracy after observing m queries as the portion of ranker pairs for which an interleaved comparison method correctly predicts the direction of the difference in Normalized Discounted Cumulative Gain (NDCG) [Järvelin and Kekäläinen 2002]. To compute NDCG difference, we use the manual relevance judgments provided with the learning to rank data set. NDCG is a standard IR evaluation measure used as ground truth in all previous work on interleaved comparison methods [Radlinski et al. 2008b]. The provided confidence intervals are 95% binomial confidence intervals. We determine whether differences are statistically significant based on the overlap between confidence intervals. Then, an interleaved comparison method is deemed more efficient than another if it Pareto dominates it (i.e., its accuracy is at least not significantly lower for all sample sizes, and significantly higher for at least one sample size).

In comparison to previous work, our setup allows evaluating interleaved comparison methods on a large set of ranker pairs in a controlled experiment. Previous work validated interleaved comparisons in real usage data [Chapelle et al. 2012; Radlinski and Craswell 2010; Radlinski et al. 2008b], which allowed assessment of these methods in a realistic setting but limited the number of possible ranker comparisons. On the other hand, [He et al. 2009] used a small number of hand-constructed test cases for their analysis. Our setup falls in between these as it is more controlled than the former, but has fewer assumptions than the latter.

The following subsections detail the experimental procedures used to investigate the fidelity and soundness of interleaved comparison methods (§6.1), and the efficiency of these methods under live (§6.2) and historical data (§6.3).

6.1. Fidelity and Soundness of Interleaved Comparison Methods

In §4.3–4.5, we analyzed the fidelity and soundness of existing interleaved comparison methods, and found that none exhibit both conditions of fidelity as defined in §4.2. In §5, we proposed a set of

[6] `http://research.microsoft.com/en-us/projects/mslr/default.aspx`

ALGORITHM 5: Experiment 1: Verifying fidelity and soundness of interleaved comparison methods.

```
1:  Input: interleave(·), compare(·), length, num_relevant, m, n
2:  correct[1..m] = zeros(m)
3:  for i = 1..n do
4:      docids, labels = generate_document_list(length, num_relevant)
5:      target_outcome = 0
6:      repeat
7:          l1 = random_permutation(docids)
8:          l2 = random_permutation(docids)
9:          if pareto_dominates(l1, l2, labels) then
10:             target_outcome = −1
11:         else if pareto_dominates(l2, l1, labels) then
12:             target_outcome = 1
13:     until target_outcome != 0
14:     for (j = 1..m) do
15:         (a, c, l) = interleave(l1, l2)
16:         append(O, compare(l1, l2, a, c, l))
17:         if sign(∑ O) = sign(target_outcome) then
18:             correct[j] + +
19: return correct[1..m]/n
```

probabilistic interleave methods and verified theoretically that they possess fidelity and soundness. In this section we detail experiments that are designed to empirically validate our theoretical results.

We conduct two experiments to address the two conditions of fidelity. Both use the experimental setup outlined in Algorithm 5. The experiment receives as input two functions $interleave$ and $compare$, which together specify an interleaving method, such as BI in Algorithm 1 ($interleave$ in lines 1–12, $compare$ in lines 13–17). It also receives arguments that specify the length and number of relevant documents of the (synthetic) documents lists to be compared, the number of impressions per run m, and the number of runs n (line 4). For each run, it initializes a list of document IDs and relevance labels as specified by $length$ and $num\_relevant$. It then randomly selects two permutations of the document IDs, and determines whether one permutation Pareto dominates the other. This test of Pareto dominance determines the target outcome to which interleaved comparison methods should converge (lines 9–12). We then interleave and compare the resulting lists as specified by the interleaved comparison method, and compare the resulting outcome to the known target outcome.

Given the experimental setup detailed above, condition (1) of fidelity (unbiasedness under random clicks) is tested by applying interleaved comparison methods under the *random* click model as specified in Table II. Under this click model, the click probability is independent of document relevance and therefore carries no information about the relative quality of rankers. Thus, interleaved comparison methods should have expected outcomes of 0 (i.e., the rankers tie in expectation). We verify this by counting the ranker pairs for which a significant difference between rankers is detected, and comparing this to the number of significant differences that would be expected solely due to random effects.

We test condition (2) of fidelity (detecting a preference when one ranker Pareto dominates the other) by running Algorithm 5 under perfect user feedback and verifying that the observed interleaved comparisons agree with the direction of the target outcome.

In both experiments, we examine the behavior of interleaved comparison methods over m queries, so both fidelity and soundness must be satisfied.

6.2. Interleaved Comparisons using Live Data

The main goal of our second set of experiments is to compare the efficiency of interleaved comparison methods in the live data setting. With "live data" we mean that click data can be collected for any

ALGORITHM 6: Experiment 2: Interleaved comparisons using *live* data.

1: **Input**: $interleave(\cdot)$, $compare(\cdot)$, Q, R, $\delta_{NDCG}(\cdot,\cdot)$, m, n
2: $correct[1..m] = zeros(m)$
3: **for** $i = 1..n$ **do**
4: $\quad \mathbf{O} = []$
5: $\quad q = random(Q)$
6: $\quad$ Sample target rankers (r_1, r_2) from R without replacement
7: $\quad$ **for** $(j = 1..m)$ **do**
8: $\quad\quad (\mathbf{a}, \mathbf{c}, \mathbf{l}) = interleave(q, r_1, r_2)$
9: $\quad\quad append(\mathbf{O}, compare(r_1, r_2, \mathbf{a}, \mathbf{c}, \mathbf{l}, q))$
10: $\quad\quad$ **if** $sign(\sum \mathbf{O}) = sign(\delta_{NDCG}(r_1, r_2))$ **then**
11: $\quad\quad\quad correct[j]++$
12: **return** $correct[1..m]/n$

interleaved lists generated by an interleaving algorithm. This means that data is collected directly for the target ranker pair being compared. Our experiments for the live data setting are detailed in Algorithm 6.

The experiment takes as input an interleaved comparison method as specified by the functions $interleave$ and $compare$, a set of queries Q, a set of rankers R, a method δ_{NDCG} which computes the true NDCG difference between two rankers, the maximum number of impressions per run m, and the number of runs n. The experiment starts by initializing a result vector $correct$ which keeps track of the interleaving method's accuracy after $1..m$ impressions (line 2). Then, for each run a query and target ranker pair are sampled from Q and R (lines 5 and 6). The target ranker pair is sampled without replacement, i.e., a ranker cannot be compared to itself. Also, we exclude cases for which the rankers have the same NDCG, so that there is a preference between rankers in all cases and we can formulate this experiment as a binary decision problem. Then, m impressions are collected by generating interleaved lists (line 8) and comparing the target rankers using the observed data (line 9). Comparison outcomes are aggregated over impressions to determine if a run would identify the preferred ranker correctly (line 10 and 11). Finally, the accuracy after $1..m$ impressions is obtained by dividing $correct$ by the number of runs n. An efficient ranker obtains a high accuracy after observing few impressions. The results of our experiments for the live data setting are reported in §7.2.

6.3. Interleaved Comparisons using Historical Data

The goal of our third set of experiments is to assess the efficiency of interleaved comparison method in a historical data setting. This setting assumes that interleaved lists cannot be directly observed for the target rankers being compared. Instead, interleaving data previously collected using a different but known original ranker pair is available. We simulate this setting by generating original ranker pairs, and collecting data for these original ranker pairs, which is then used to estimate comparison outcomes for the target pair. The detailed procedure is shown in Algorithm 7.

The arguments passed to Algorithm 7, as well as its initialization and overall structure, are identical to those for the live data experiments shown in Algorithm 6. The main differences are in lines 6 to 9. In addition to the target ranker pair, an original ranker pair is randomly sampled, again without replacement so that there is no overlap between the rankers used in a given run (line 6). Then, for each impression, the interleaving data is collected for the original ranker pair (line 8). The target rankers are compared using this data collected with the original rankers (line 9). Experiment outcomes are computed in terms of accuracy for the target rankers as before. Again, an efficient ranker obtains high accuracy after observing few historical samples. The results of our experiments for the historical data setting are reported in §7.3.

ALGORITHM 7: Experiment 3: Interleaved comparisons using *historical* data.

Input: $interleave(\cdot)$, $compare(\cdot)$, Q, R, $\delta_{NDCG}(\cdot,\cdot)$, m, n
$correct[1..m] = zeros(m)$
for $i = 1..n$ **do**
  $O = []$
  $q = random(Q)$
  Sample original pair (r_{o_1}, r_{o_2}) and target pair (r_{t_1}, r_{t_2}) from R without replacement
  for $j = 1..m$ **do**
    $(\mathbf{a}, \mathbf{c}, \mathbf{l}) = interleave(q, r_{o_1}, r_{o_2})$
    $O[i] = compare(r_{t_1}, r_{t_2}, r_{o_1}, r_{o_2}, \mathbf{a}, \mathbf{c}, \mathbf{l}, q)$
    if $sign(\sum O) = sign(\delta_{NDCG}(r_{t_1}, r_{t_2}))$ **then**
      $correct[j]++$
return $correct[1..m]/n$

7. RESULTS AND DISCUSSION

In this section we detail our three sets of experiments and present and analyze the obtained results. Our first set of experiments verifies the results of our theoretical analysis of the fidelity and soundness of interleaved comparison methods. Our second set of experiments examines the sample efficiency of interleaved comparison methods when comparing rankers using live data (§7.2). Our third set of experiments evaluates interleaved comparison methods using historical data (§7.3). In addition to presenting our main results, we analyze the interleaved comparison methods' robustness to noise in user feedback and to varying parameter settings.

7.1. Fidelity and Soundness of Interleaved Comparison methods

In this section, we present and discuss the results of our first set of experiments, designed to verify the results of our theoretical analysis of interleaved comparison methods. We compare the baseline methods BI, TD, and DC and our proposed method PI-MA, defined as follows:

— **BI**: the Balanced Interleave method following [Chapelle et al. 2012], as detailed in Algorithm 1 (§3).
— **TD**: the Team Draft method following [Chapelle et al. 2012], as detailed in Algorithm 2 (§3).
— **DC**: the Document Constraint method following [He et al. 2009], as detailed in Algorithm 3 (§3).
— **PI-MA**: probabilistic interleaving with marginalization over assignments as defined in Eq. 6–9 (cf. §5.2).

The experiments use the experimental setup described in §6.1. We run experiments for result lists of length 10 with the number of relevant documents $num\_relevant$ sampled uniformly at random from $[1, 3]$. We run the comparison of each ranker pair for $m = 500$ impressions, and repeat this for $n = 500$ ranker pairs.

Figure 7 shows the results of our first experiment, in which we verify condition (1) of fidelity and soundness. An interleaved comparison method that exhibits fidelity and soundness should, in expectation, not detect any differences between rankers when user clicks are random. We verify this as follows. For 500 ranker pairs, we perform interleaved comparisons on up to 500 impressions each. We then conduct a t-test on the observed mean outcome for each of these 500 ranker pairs and compare the number of detected significant differences to those expected under random comparison outcomes.

For the interleaved comparison outcomes TD and PI, we see the behavior expected under random clicks. For example, for a t-test with $p = 0.05$, these methods detect between 20 and 29 significant differences between rankers, where 25 (5%) differences are expected due to random variations. Similarly, for $p = 0.01$, we observe up to 9 significant differences (for TD, after 200 impressions). For $p = 0.001$, we observe up to 2 significant differences (TD after 400 and 500 impressions, and PI after 300 impressions). We compare the observed number of differences to the expected number

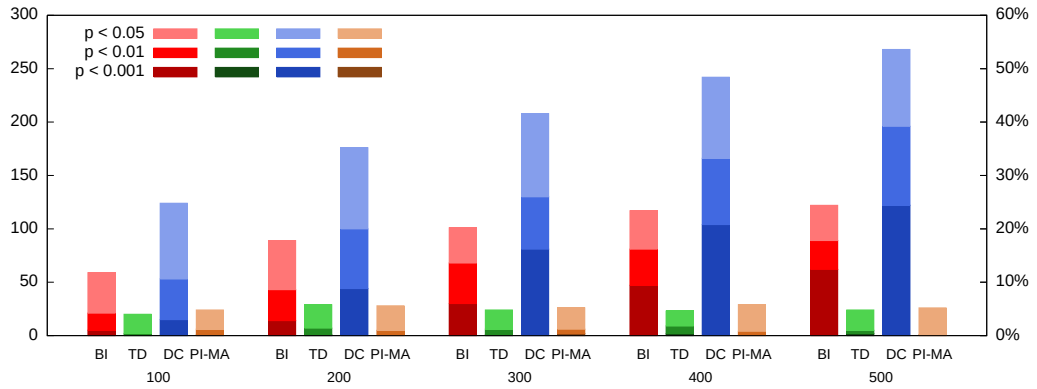


Fig. 7. Results, condition (1) of fidelity. Number of significant differences between rankers detected under random user clicks for different interleaved comparison methods and p-values on 500 ranker pairs after 100–500 user impressions. Colors represent interleaved comparison methods and shades represent levels of p. For a p-value of 0.05, an interleaved comparison method that exhibits fidelity condition (1) should detect significant differences in about 25 (5%) of the performed comparisons, independently of the number of impressions.

using a binomial significance test with $p = 0.001$. For TD and PI, these are not significantly different from the expected number of differences. This result supports the conclusion that, in expectation, TD and PI do not detect differences between rankers under random user clicks.

For BI and DC, our earlier theoretical analysis showed that condition (1) of fidelity is violated. This is confirmed by our experimental results in Figure 7. After 100 impressions, BI detects significant differences between rankers in 59 cases for $p = 0.05$ and in 5 cases for $p = 0.001$. The number of detected significant differences increases to 122 (for $p = 0.05$) and 62 (for $p = 0.001$) after 500 impressions. For DC, the number of falsely identified differences between rankers is even higher, ranging from 124 ($p = 0.05$) and 15 ($p = 0.001$) after 100 impressions up to 268 ($p = 0.05$) and 122 ($p = 0.001$) after 500 impressions. This means that after 500 impressions DC detects a significant difference for more than half the compared ranker pairs when we assume a p-value of $p = 0.05$. For all numbers of impressions and all levels of p, the number of differences detected by BI and DC is significantly higher than expected due to random variations (again using a binomial significance test and $p = 0.001$). This shows that BI and DC violate condition (1) of fidelity as predicted by our analysis.

Figure 8 shows our results for condition (2) of fidelity. As detailed in §6.1, we generate ranker pairs where one ranker Pareto dominates the other, so that we know which ranker should be preferred in expectation by an interleaved comparison method that exhibits fidelity condition (2) and soundness. We then compare the known target outcome to the comparison outcomes detected by each interleaved comparison method after m impressions. An interleaved comparison method that exhibits condition (2) of fidelity and soundness should always converge to agreement with the target outcome as $n \to \infty$. For BI, we see that the detected outcome agrees with the target ranker in 88% of the tested ranker pairs. Manual inspection of the remaining cases confirmed that they were instances with several relevant documents (i.e., several clicks), where the rankers were tied on the lowest-ranked clicked document but not on higher-ranked clicked documents (as in the example discussed in §4.3, Figure 2).

The agreement of TD with the target outcome is initially lower than that of BI, but converges to a similar level (87%).[7] The convergence over several impressions is due to the higher level of randomization under TD as compared to BI. The cases where TD violates fidelity condition (2) are

[7]Note that the observed agreement levels are not predictive for real-live settings, as the cases where BI or TD violate fidelity may be more or less frequent than in our simulation, depending on the rankers being compared.

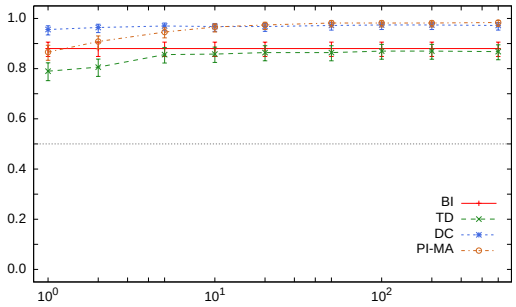


Fig. 8. Results, condition (2) of fidelity. Portion of correctly identified preferences when the known better ranker Pareto dominates the other ranker after 1–500 user impressions under *perfect* click feedback.

found to be cases where a relevant document is moved up by one rank by the dominating ranker (cf., §4.4, Figure 3).

DC shows high agreement with the target outcome, with fast convergence due to a low level of randomization. The method converges to an agreement with the target outcome of 97.2%. The observed cases where the method disagrees with the target outcome are ones where the relative order of two relevant documents is switched (i.e., d_1 and d_3 are both relevant and clicked, and one ranker places d_1 before d_3 and the other places d_3 above d_1, also see §4.5).

Finally, PI-MA starts at an agreement level slightly below that of BI. It takes longer to converge than the other interleaved comparison methods, due to the higher level of randomization during interleaving. The method passes DC after 20 impressions have been observed. After 500 impressions, it agrees with the target outcome in 98.4% of the cases. In none of the remaining 8 cases could we find any systematic disagreement with the target outcomes, and they converged to the target outcome when run for larger n. These results support our analysis that PI-MA detects a preference for a ranker that Pareto dominates another ranker as $n \to \infty$.

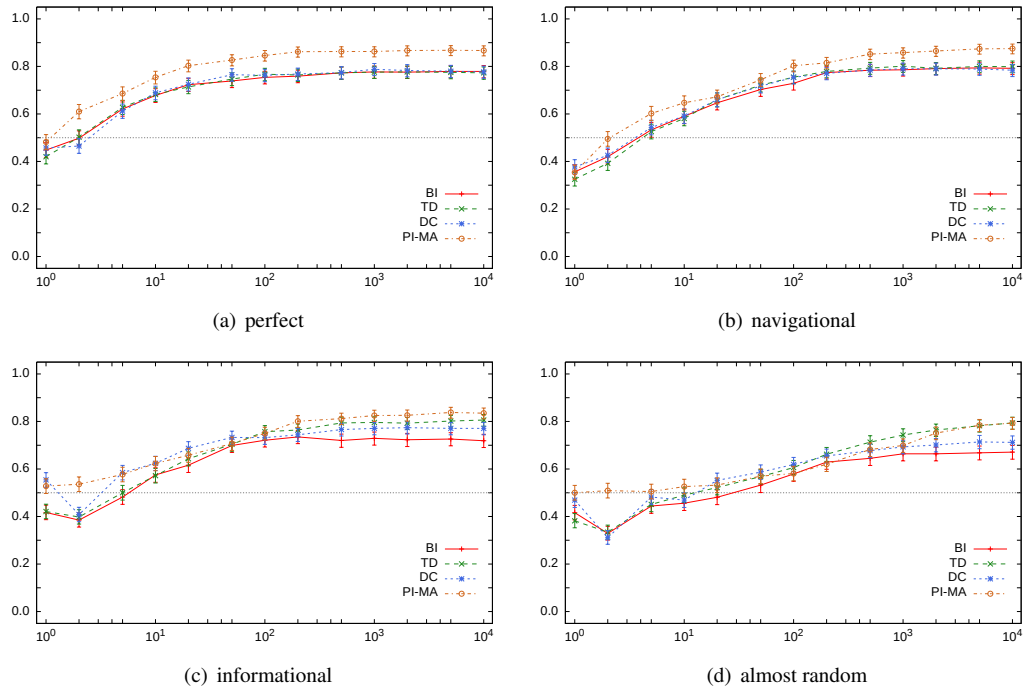
Our first two experiments confirm the results of our theoretical analysis regarding fidelity and soundness. As predicted by our theoretical results, BI and DC violate both conditions of fidelity, while TD violates condition (2) of fidelity. Our results support the conclusion that PI-MA exhibits both soundness and fidelity.

7.2. Interleaved Comparisons using Live Data

In this section, we present the results of our evaluation of interleaved comparison methods in a live data setting, where interleaving methods interact directly with users. We compare the same three baseline methods BI, TD, and DC and our proposed method PI-MA, as defined in the previous section (§7.1). We run experiments for $m = 10{,}000$ impressions, $n = 1{,}000$ times. The experiments use the experimental setup described in §6.2.

The results obtained for our four user models are shown in Figure 9. Each plot shows the accuracy achieved by each interleaved comparison method over the number of impressions seen for a given user model. The performance of a random baseline would be 0.5, and is marked in grey. Note that the performance of an interleaving method can be below the random baseline in cases where no decision is possible (e.g., the method infers a tie when not enough data has been observed to infer a preference for one of the rankers; the rankers are sampled in such a way that there always is a difference according to the NDCG ground truth). When comparing the efficiency of interleaved comparison methods, we consider both how many impressions are needed before a specific accuracy level is achieved, and what final accuracy is achieved after e.g., $10{,}000$ impressions.

For the *perfect* click model (cf., Figure 9(a)) we find that the baseline methods BI, TD and DC achieve close to identical performance throughout the experiment. The final accuracies of these methods after observing $10{,}000$ impressions are 0.78, 0.77, and 0.78 respectively, and there is no



(a) perfect (b) navigational

(c) informational (d) almost random

Fig. 9. Results, live setting. Portion of correctly identified preferences (accuracy) on 1,000 randomly selected ranker pairs and queries, after 1-10,000 user impressions with varying click models.

significant difference between the methods. We conclude that these methods are similarly efficient when comparing rankers on highly reliable live data. Our proposed method PI-MA is found to be more efficient than all baseline methods on live data under the perfect click model as it outperforms them by a large and statistically significant margin for all sample sizes. After observing only 50 impressions, PI-MA can more accurately distinguish between rankers than either of the other methods after observing $10{,}000$ impressions. Its final accuracy of 0.87 is significantly higher than that of all baselines. Compared to the best-performing baseline (here, BI), PI-MA can correctly detect a preference on 11.5% more ranker pairs after observing $10{,}000$ impressions.

Results for the *navigational* click model are shown in Figure 9(b). In comparison to the perfect click model, this model has a higher position bias (higher stop probabilities), and a steeper decay of click probabilities (quadratic, so that the difference between the highest relevance grades is relatively bigger than under the perfect click model). The increase in position bias is expected to lead to a decrease in efficiency (this effect was identified for BI, TD, and DC in [He et al. 2009]). This effect is confirmed by our results, which can be seen in the slower increase in accuracy as compared to the perfect click model. For example, under the navigational model, approximately 50 impressions are needed before all interleaved comparison methods achieve an accuracy of at least 0.7, while for the perfect model, only about 20 impressions need to be observed for the same level of accuracy. The steeper decay in click probabilities is expected to lead to click data that better corresponds to the implementation of gain values in NDCG than the linear decay implemented in the perfect click model. We find that the accuracy of all methods after $10{,}000$ iterations is slightly higher under the navigational model (the accuracy for BI is 0.79, for TD 0.80, for DC 0.78, and for PI-MA 0.88), but none of the differences is statistically significant. We can conclude that under the navigational model, interleaving methods have lower efficiency (due to increased position bias), but they converge to at least the same level of accuracy (possibly slightly higher, due to the better match with NDCG

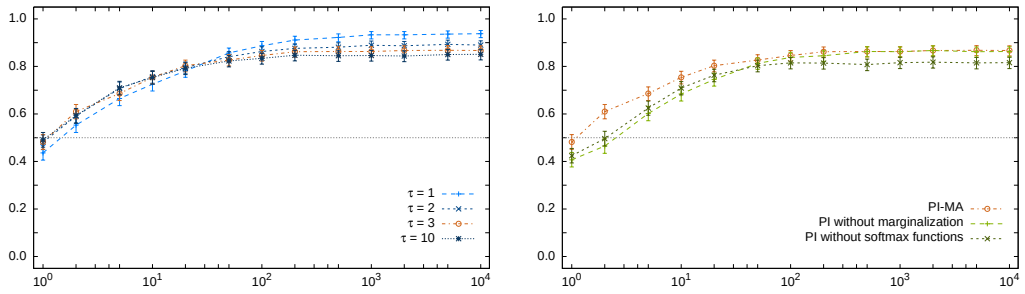
gain values) as under the perfect click model. Comparing the individual methods, we again find that PI-MA is more efficient than any of the baseline methods. After $10{,}000$ impressions, it achieves a significant increase in accuracy of 10% over the baseline methods.

The *informational* click model has a level of position bias that is similar to that of the navigational click model, but a higher level of noise. Thus, users consider more documents per query, but their click behavior makes documents more difficult to distinguish. Figure 9(c) shows the results for this click model. The increase in noise affects the efficiency of the different interleaved comparison methods in different ways. For small sample sizes, the efficiency of all interleaved comparison methods is similar to that of the navigational model, with all methods achieving an accuracy of 0.7 within 50 samples. However, after $10{,}000$ impressions, BI achieves a final accuracy of 0.72 (TD – 0.81, DC – 0.77, and PI-MA – 0.84). The final accuracy of BI and of PI-MA is substantially lower than under the navigational model. The final accuracy of PI-MA is significantly higher than that of BI and DC under the informational model, and higher (but not significantly so) than that of TD. BI appears to be particularly strongly affected by noise. This method performs significantly worse than all other interleaved comparison methods in this setting. Outcomes computed under this method rely on rank-differences at the lowest-clicked document. As individual clicks become less reliable, so do the comparison outcomes.

Results for the *almost random* click model reflect the performance of interleaved comparison methods under high noise and high position bias (Figure 9(d)). We find that efficiency decreases substantially for all methods. For example, TD is the first method to achieve an accuracy of 0.7 after 500 impressions. After $10{,}000$ impressions, BI achieves an accuracy of only 0.67 and the accuracy of DC is 0.71. TD appears to be the most robust against this form of noise, maintaining an accuracy of 0.79. PI-MA performs better than the baseline methods on small sample sizes, because marginalization helps avoid noisy inferences. Its performance after $10{,}000$ impressions is the same as for TD, thus its overall efficiency is still higher than that of TD. In general, PI-MA is expected to converge to the same results as TD in settings with high noise and high position bias, such as the one simulated here. In these settings, the method cannot accurately trade-off between clicks at different positions.

Our results for the different user models indicate that PI-MA Pareto dominates the baseline methods in terms of performance. Under highly reliable click feedback, the baseline methods perform similarly well, while PI-MA is significantly more accurate at all sample sizes. The reason is that PI-MA can trade off differences between ranks more accurately. For all methods, efficiency decreases as position bias increases, which is in line with earlier work. Increasing noise affects the interleaving methods differently. BI appears to be affected the most strongly, followed by DC. TD is relatively robust to noise. PI-MA reduces to TD when the level of noise becomes extreme. None of the baseline methods was found to be significantly more accurate than PI-MA at any sample size or level of click noise. Therefore, we conclude that PI-MA is more efficient than the other methods.

After comparing PI-MA to the baseline methods, we now turn to analyzing PI-MA in more detail. PI-MA has one parameter τ. This parameter can be set to change the trade-off between clicked documents at different ranks, similar to the position discount in NDCG. Low values of τ result in slightly more randomization in the constructed interleaved result lists, which means that documents at lower ranks have a higher chance of being placed in the top of the result list and are more likely to be clicked. When comparing interleaving outcomes to NDCG difference, we expect more accurate results for smaller values of τ, as NDCG uses a relatively weak position discount (namely $\log(r)$). This is confirmed by our results in Figure 10(a) (here: perfect click model). For settings of τ that are smaller than the default value $\tau = 3$ (i.e., $\tau \in (1, 2)$, accuracy is higher than for the default settings. Increasing the parameter value to $\tau = 10$ decreases the accuracy. While all parameter settings $\tau > 0$ result in an interleaved comparison method that exhibits fidelity as defined in Definition 4.2, an appropriate value needs to be chosen for given applications of this method. Higher values place more emphasis on even small differences between rankings, which may be important in settings where users are typically impatient (e.g., for navigational queries). In settings where users are expected to be more patient, or tend to explore results more broadly, a lower value should be chosen. In comparison,



(a) PI-MA with varying settings of τ

(b) PI without marginalization / without softmax functions

Fig. 10. Analysis, live setting. Accuracy on 1,000 randomly selected ranker pairs and queries, after 1–10,000 user impressions using PI-MA with varying τ, and without softmax functions / marginalization under the perfect click model.

the baseline methods BI, TD, and DC make implicit assumptions about how clicked documents at lower ranks should be weighted, but do not allow the designer of the retrieval system to make this decision explicit.

Finally, we analyze PI-MA in more detail by evaluating its performance after removing individual components of the method (Figure 10(b)). The figure shows PI-MA ($\tau = 3$), compared to PI-MA without marginalization, and without softmax functions. We find that the complete method has the highest efficiency, as expected. Without marginalization, comparisons are less reliable, leading to lower efficiency, especially on small sample sizes. The performance difference is compensated for with additional data, confirming that PI and PI-MA converge to the same comparison outcomes. When deterministic ranking functions are used instead of softmax functions, we observe lower accuracy throughout the experiment. Without softmax functions, PI-MA does not trade off between differences at different ranks, leading to lower agreement with NDCG. We conclude that PI-MA is more efficient than variants of the method without marginalization, and without softmax functions. This result confirms the results of our analysis.

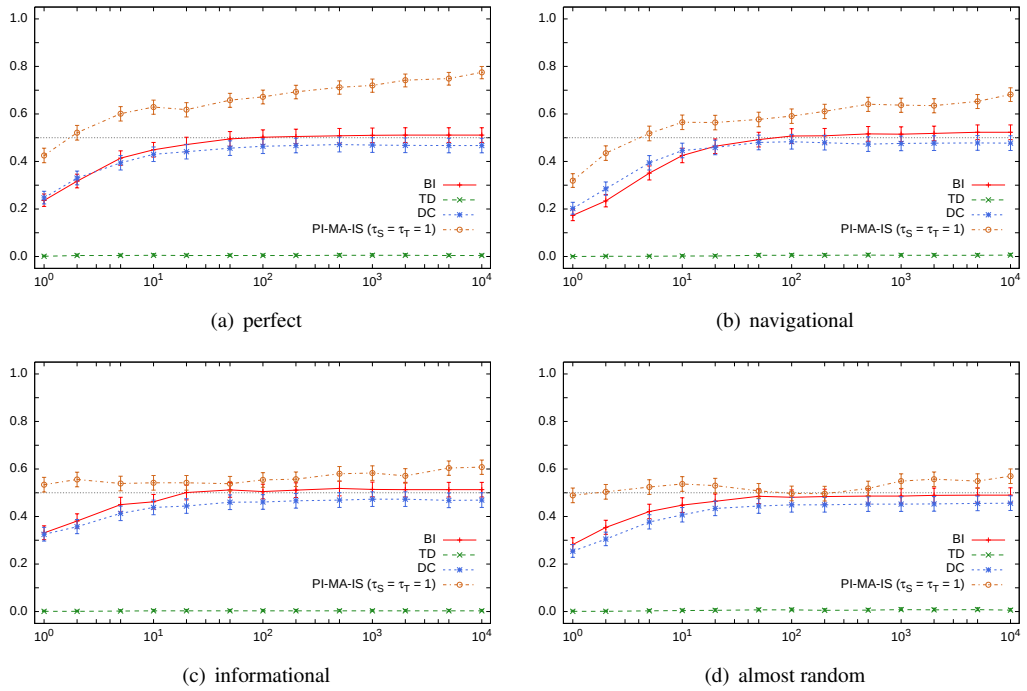
In this section, we studied the efficiency of interleaved comparison methods in the live data setting, where click feedback for all interleaved lists can be observed directly. We found that our proposed method PI-MA is more efficient than all baseline methods. Performance gains were found to be particularly high under perfect click feedback, and we identified the effects of increased noise and position bias on all methods. Finally, we analyzed our method PI-MA in more detail, which confirmed that our marginalization step increases the efficiency of PI as expected.

7.3. Interleaved Comparisons using Historical Data

In this section, we evaluate interleaved comparison methods in a historical data setting, where only previously observed interaction data is available. Our experiments do not focus on how to collect such data, but rather assume that data is available from previous experiments and the task is to use this data effectively. We compare the following methods for interleaved comparisons using historical data:

— **BI**: directly applies BI to historical data, as discussed in §4.3.
— **TD**: applies TD to all assignments that match historical data, as discussed in §4.4.
— **DC**: directly applies DC to historical data, as discussed in §4.5.
— **PI-MA-IS**: our full importance sampling estimator with marginalization over assignments, as defined in Eq. 11 (cf., §5.3). Unless specified otherwise, we use a setting of $\tau = 1$ for both the source and the target distribution.

We use the experimental setup described in §6, and the procedure detailed in §6.3. Each run is repeated $n = 1{,}000$ times and has a length of $m = 10{,}000$ impressions. For each run, we collect



(a) perfect (b) navigational

(c) informational (d) almost random

Fig. 11. Results, portion of correctly identified preferences (accuracy) on 1,000 randomly selected ranker pairs and queries, after 1–10,000 user impressions with varying click models.

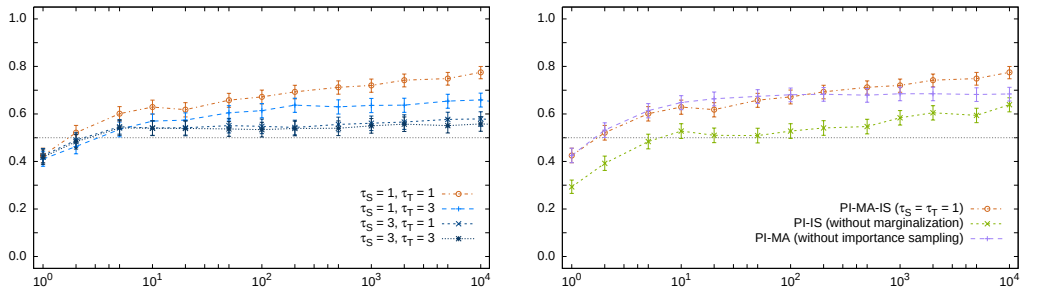
historical data using a randomly selected source ranker pair, and use the collected data to infer information about relative performance of a randomly selected target ranker pair.

In comparison to the live data setting, we expect interleaved comparison methods to have lower efficiency. This is particularly the case for this setting where source and target distributions can be very different from each other. In settings where source and target distributions are more similar to each other (such as learning to rank settings), efficiency under historical data is expected to be much higher, so the results presented here constitute a lower bound on performance.

Figure 11 shows the results obtained in the historical data setting. For the perfect click model (Figure 11(a)), we see the following performance. BI shows close to random performance, and its performance after 10,000 impressions is not statistically different from the random baseline. DC stays significantly below random performance. These results suggest that the two methods cannot use historical data effectively, even under very reliable feedback. The reason is that differences between the observed interleaved lists and the lists that would be generated by the target rankers are not compensated for. TD shows very low accuracy, close to zero. This result confirms our analysis that indicated that this method can reuse only a small portion of the historical data. Since few lists are useable by this method, most comparisons result in a tie between the compared target rankers.

The results in Figure 11(a) confirm that PI-MA makes it possible to effectively reuse previously collected data, and that it is much more efficient than the baseline methods. After 10,000 impressions, this method achieves an accuracy of 0.78. Following the trend of this experiment, accuracy is expected to continue to increase as more impressions are added.

The relative performance of the interleaved comparison methods is the same for all investigated click models. In comparison to the perfect click model, efficiency of PI-MA-IS decreases with increases in click noise as expected. However, the method performs significantly better than all baseline methods under any noise level. For the navigational model, performance after 10,000



(a) PI-MA-IS with varying settings of τ

(b) PI-MA-IS vs. PI without marginalization / without importance sampling

Fig. 12. Results, accuracy on 1,000 randomly selected ranker pairs and queries, after 1-10,000 user impressions using PI-MA with varying τ under the perfect click model.

impressions is 0.68 (Figure 11(b)), for the informational model it is 0.61 (Figure 11(c)), and for the almost random model 0.57 (Figure 11(d)). Thus, it can be seen that efficiency degrades gracefully with increases in noise. For high levels of noise (such as under the almost random click model) the required amount of data can be several orders of magnitude higher than under the perfect click model to obtain the same level of accuracy. Performance of the baseline methods does not appear to be substantially impacted by the level of noise in user feedback in the historical data setting.

After comparing interleaved comparison methods in the historical feedback setting, we turn to analyzing the characteristics of PI-IS-MA in more detail. So far, we assumed that historical data was available from an earlier interleaved comparison experiment, and that we had no influence on how that data was sampled. Now we relax this assumption and investigate the effect of choosing different values of τ during both data collection and inference (Figure 12(a)). Under historical data, τ has several effects. For the source rankers (τ_S), it determines the level of exploration during data collection. As $\tau_S \rightarrow \infty$, the level of exploration approaches uniformly random selection of document permutations. A high level of exploration can ensure that result lists that are likely under the target rankers are sufficiently well covered during data collection, which reduces variance in the later comparisons. This is confirmed by comparing our results for PI-MA-IS with the parameter setting $\tau_S = 1, \tau_T = 3$ to those for the setting $\tau_S = 3, \tau_T = 3$. In both runs, the comparison function is identical, however in the first setting, data collection was more exploratory. This leads to a significant increase in efficiency.

Changing τ only for the target distribution (τ_T) also has an effect on variance, although it is weaker than that observed for the source distribution. Two factors play a role here, 1) smaller values of τ_T lead to comparisons that more accurately correspond to NDCG position discounts (cf., §7.2, Figure 10(a)), and 2) smaller values of τ_T make the target distribution slightly broader (the differences between most and least likely interleaved lists becomes smaller), resulting in smaller differences between the source and target distributions and therefore smaller importance weights. The relative importance of these two effects can be estimated with the help of our results obtained in the live setting. There, the accuracy for $\tau = 1$ after 10,000 impressions is significantly (by 7.5%) higher than for $\tau = 3$. Under historical data, performance for the setting $\tau_S = 1, \tau_T = 1$ is also significantly higher than for the setting $\tau_S = 1, \tau_T = 3$. Here, the increase is 17.6% – more than twice as high as in the live setting. We conclude that a large portion of this increase is due to the reduced distance between source and target distribution and the resulting reduction in variance. Finally, when comparing settings with low exploration under the source distribution ($\tau_S = 3$) we see only marginal performance differences. This result suggests that a high amount of exploration during data collection is crucial for achieving high efficiency under PI-IS-MA.

Finally, we examine how different components of PI-IS-MA contribute to the performance of this method under historical data. Figure 12(b) shows our previous results for PI-IS-MA and for the following additional runs:

— **PI-IS**: PI that uses the naive importance sampling estimator in Eq. 10 to compensate for differences between source and target distribution (cf., §5.3).
— **PI-MA**: directly applies PI-MA as defined in Eq. 6-9 (cf. §5.2), without compensating for differences between source and target distributions.

Our results confirm the outcomes of our analysis and derivation of PI-MA-IS (cf., 5.3). The variant PI-IS (i.e., without marginalization) is significantly less efficient than the full method PI-IS-MA. This confirms that marginalization is an effective way to compensate for noise. The effect is much stronger than in the live data setting, because under historical data the level of noise is much higher (due to the variance introduced by importance sampling). In the limit, we expect that performance of PI-IS converges to the same value as PI-IS-MA, but after $10{,}000$ impressions its accuracy is 0.639, 17.5% lower than after observing the same number of samples in the live setting. If PI-MA is applied without importance sampling, we see that efficiency is as high as for PI-IS-MA for small sample sizes. However, we also observe the bias introduced under this method, as it converges to a lower accuracy after processing approximately 200 impressions. The performance of PI-MA when applied to historical data is found to be 0.68 after $10{,}000$ impressions, 12% lower than that of PI-MA-IS. These results demonstrate that PI-MA-IS successfully compensates for bias while maintaining high efficiency.

To summarize, our experiments in the historical data setting confirm that PI-MA-IS can effectively reuse historical data for inferring interleaved comparison outcomes. Alternatives based on existing interleaved comparison methods were not able to do this effectively, due to data sparsity and bias. The efficiency of PI-MA-IS under historical data is found to decrease with increases in click noise, as expected. More detailed analysis shows that choosing a sufficiently exploratory source distribution is crucial for obtaining good performance. Finally, our analysis showed that marginalization and importance sampling contribute to the effectiveness of PI-MA-IS as suggested by our analysis.

8. CONCLUSION

In this article, we introduced a new framework for analyzing interleaved comparisons methods, analyzed existing methods, and proposed a novel, probabilistic interleaved comparison method that addresses some of the challenges raised in our analysis. The proposed analysis framework characterizes interleaved comparison methods in terms of fidelity, soundness, and efficiency. Fidelity reflects whether the method measures what it is intended to measure, soundness refers to its statistical properties, and efficiency reflects how much sample data a method requires to make comparisons.

We analyzed existing interleaved comparison methods using the proposed framework, and found that none exhibit a minimal requirement of fidelity, namely that the method prefers rankers that rank clicked documents higher. We then proposed a new interleaved comparison method, probabilistic interleave, and showed that it does exhibit fidelity. Next, we devised several estimators for our probabilistic interleave method, and proved their statistical soundness. These estimators included a naive estimator, a marginalized estimator designed to improve effectiveness by reducing variance (PI-MA), and an estimator based on marginalization and importance sampling (PI-MA-IS), that increases efficiency by allowing the reuse of previously collected (historical) data.

We empirically confirmed the results of our analysis through a series of experiments that simulate user interactions with a retrieval system using a fully annotated learning to rank data set and click models. First, we verified our theoretical results regarding soundness and fidelity of interleaved comparison methods. Second, our experiments in the live data setting showed that PI-MA is more efficient than all existing interleaved comparison methods. Our detailed analysis of different variants of PI-MA confirms that PI-MA with marginalization and softmax functions is more efficient than variants without either component. Third, in our experiments with simulated historical click feedback, we found that PI-MA-IS can effectively reuse historical data. Due to the increase in noise due to

importance sampling, sample efficiency is lower than under live data, as expected. Our last set of experiments also confirmed that the difference between the source and target distributions has a strong effect on the sample efficiency of PI-MA-IS.

Our work is relevant to research and application of IR evaluation methods. First, our analysis framework is a step towards formalizing the requirements for interleaved comparison methods. Using this framework, we can make more concrete statements about how interleaved comparison methods should behave. Our analysis of existing methods shows how the use of this framework can shed more light on their characteristics. In addition, our proposed probabilistic interleaved comparison method is the first to exhibit fidelity, and we showed how different components of the method relate to frequently-made assumptions about user behavior and expectations (e.g., relating to the position discount in NDCG). Regarding the application of interleaved comparison methods, our method PI-MA can be used to more explicitly define and better understand what an experimental outcome captures. Finally, the method was shown to improve upon the sample efficiency of previous methods.

Our extension of probabilistic interleaving to the historical data setting resulted in the first method that can effectively estimate interleaved comparison outcomes from data that was not collected using the target ranker pair. This extension can lead to substantial improvements in sample efficiency, especially in settings where many comparisons of similar rankers need to be made, such as large-scale evaluation of (Web) search engines, or in learning to rank. In such settings, where the compared rankers are relatively similar to each other, the differences between source and target rankers are expected to be particularly small, which results in low variance and therefore high efficiency of our importance-sampling-based method. A first approach that uses probabilistic interleaving for data reuse in online learning to rank was shown to substantially and significantly speed up learning, especially under noisy click data [Hofmann et al. 2013].

Interleaved comparison methods are still relatively new, and an important direction for future research is to better understand and formalize what differences between rankers these methods can measure. Such work could focus, for example, on more detailed analysis and experimental characterization of the relationships between interleaved comparison methods and traditional IR evaluation metrics.

Our analysis and experiments explicitly made a number of assumptions about the relationship between relevance and user click behavior. These assumptions were based on earlier work on click models, but there is still a large gap between the current models and the very noisy observations of user behavior in real (Web) search environments. As more and more accurate click models are being developed, we expect the resulting understanding of click behavior to influence and complement work on interleaved comparison methods. Open questions include whether and how click models can be used to evaluate rankers, and how this type of evaluation relates to interleaved comparison methods; Chuklin et al. [2013] provide a recent exploration of these questions, showing that offline metrics that are based on click models are more strongly correlated with online experimental outcomes (obtained using A/B-testing or interleaving) than traditional offline metrics. Also, while current interleaved comparison methods focused on aggregating clicks, a broader range of user behavior could be taken into account, and may help to e.g., decrease noise.

In our evaluation of the historical data setting, we assumed that historical data was obtained from earlier comparisons, and we focused on identifying methods that can effectively use the given data. In learning to rank settings, it may be possible to influence data collection, possibly using original distributions that reduce variance for the target ranker comparisons. Such sampling methods could make the reuse of historical data for interleaved comparisons even more effective. Finally, in settings where both historical and live data are available, combining these estimators using statistical tools for combining estimators in an unbiased way that minimizes variance [Graybill and Deal 1959] is expected to result in further performance gains. This is another promising direction for future research.

ACKNOWLEDGMENTS

We would like to thank our anonymous reviewers for their valuable feedback.

This research was partially supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements nr 258191 (PROMISE Network of Excellence) and 288024 (LiMoSINe project), the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.061.814, 612.061.815, 640.004.802, 727.011.005, 612.001.116, HOR-11-10, the Center for Creation, Content and Technology (CCCT), the Hyperlocal Service Platform project funded by the Service Innovation & ICT program, the WAHSP and BILAND projects funded by the CLARIN-nl program, the Dutch national program COMMIT, by the ESF Research Network Program ELIAS, and the Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW).

REFERENCES

AGICHTEIN, E., BRILL, E., AND DUMAIS, S. 2006. Improving web search ranking by incorporating user behavior information. In *SIGIR '06*. ACM, New York, NY, USA, 19–26.

CARTERETTE, B. AND JONES, R. 2008. Evaluating search engines by modeling the relationship between relevance and clicks. In *Advances in Neural Information Processing Systems 20*, J. Platt, D. Koller, Y. Singer, and S. Roweis, Eds. NIPS '07. MIT Press, Cambridge, MA, USA, 217–224.

CHAPELLE, O., JOACHIMS, T., RADLINSKI, F., AND YUE, Y. 2012. Large-scale validation and analysis of interleaved search evaluation. *ACM Trans. Inf. Syst. 30,* 1, 6:1–6:41.

CHAPELLE, O., METLZER, D., ZHANG, Y., AND GRINSPAN, P. 2009. Expected reciprocal rank for graded relevance. In *CIKM '09*. ACM, New York, NY, USA, 621–630.

CHAPELLE, O. AND ZHANG, Y. 2009. A dynamic bayesian network click model for web search ranking. In *SIGIR '09*. ACM, New York, NY, USA, 1–10.

CHEN, Y. 2005. Another look at rejection sampling through importance sampling. *Statistics & probability letters 72,* 4, 277–283.

CHUKLIN, A., SERYUKOV, P., AND DE RIJKE, M. 2013. Click model-based information retrieval metrics. In *SIGIR '13*. ACM.

DUDÍK, M., LANGFORD, J., AND LI, L. 2011. Doubly robust policy evaluation and learning. In *ICML'11*. ACM, New York, NY, USA, 1097–1104.

DUPRET, G. AND LIAO, C. 2010. A model to estimate intrinsic document relevance from the clickthrough logs of a web search engine. In *WSDM '10*. ACM, New York, NY, USA, 181–190.

DUPRET, G., MURDOCH, V., AND PIWOWARSKI, B. 2007. Web search engine evaluation using click-through data and a user model. In *In Proceedings of the Workshop on Query Log Analysis*.

FOX, S., KARNAWAT, K., MYDLAND, M., DUMAIS, S., AND WHITE, T. 2005. Evaluating implicit measures to improve web search. *ACM TOIS 23,* 2, 147–168.

GRAYBILL, F. AND DEAL, R. 1959. Combining unbiased estimators. *Biometrics 15,* 4, 543–550.

GUO, F., LIU, C., AND WANG, Y. M. 2009. Efficient multiple-click models in web search. In *WSDM '09*. ACM, New York, NY, USA, 124–131.

HALMOS, P. R. 1946. The theory of unbiased estimation. *Ann. Math. Statist. 17,* 1, 34–43.

HE, J., ZHAI, C., AND LI, X. 2009. Evaluation of methods for relative comparison of retrieval systems based on clickthroughs. In *CIKM '09*. ACM, New York, NY, USA, 2029–2032.

HOFMANN, K., BEHR, F., AND RADLINSKI, F. 2012a. On caption bias in interleaving experiments. In *CIKM '12*. ACM, New York, NY, USA.

HOFMANN, K., HUURNINK, B., BRON, M., AND DE RIJKE, M. 2010. Comparing click-through data to purchase decisions for retrieval evaluation. In *SIGIR '10*. ACM, New York, NY, USA, 761–762.

HOFMANN, K., SCHUTH, A., WHITESON, S., AND DE RIJKE, M. 2013. Reusing historical interaction data for faster online learning to rank for IR. In *WSDM '13*.

HOFMANN, K., WHITESON, S., AND DE RIJKE, M. 2011. A probabilistic method for inferring preferences from clicks. In *CIKM '11*. ACM, USA, 249–258.

HOFMANN, K., WHITESON, S., AND DE RIJKE, M. 2012b. Estimating interleaved comparison

outcomes from historical click data. In *CIKM '12*. ACM, USA.
JÄRVELIN, K. AND KEKÄLÄINEN, J. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst. 20,* 4, 422–446.
JI, S., ZHOU, K., LIAO, C., ZHENG, Z., XUE, G.-R., CHAPELLE, O., SUN, G., AND ZHA, H. 2009. Global ranking by exploiting user clicks. In *SIGIR '09*. ACM, New York, NY, USA, 35–42.
JOACHIMS, T. 2002. Optimizing search engines using clickthrough data. In *KDD '02*. ACM, New York, NY, USA, 133–142.
JOACHIMS, T. 2003. Evaluating retrieval performance using clickthrough data. In *Text Mining*, J. Franke, G. Nakhaeizadeh, and I. Renz, Eds. Springer, Berlin, Germany, 79–96.
JUNG, S., HERLOCKER, J. L., AND WEBSTER, J. 2007. Click data as implicit relevance feedback in web search. *Information Processing & Management 43,* 3, 791 – 807.
KAMPS, J., KOOLEN, M., AND TROTMAN, A. 2009. Comparative analysis of clicks and judgments for IR evaluation. In *WSCD'09*. 80–87.
LANGFORD, J., STREHL, A., AND WORTMAN, J. 2008. Exploration scavenging. In *ICML '08*. ACM, New York, NY, USA, 528–535.
LEHMANN, E. L. 1999. *Elements of Large-Sample Theory*. Springer, Berlin, Germany.
LI, L., CHU, W., LANGFORD, J., AND WANG, X. 2011. Unbiased offline evaluation of contextual-bandit-based news article recommendation algorithms. In *WSDM '11*. ACM, New York, NY, USA, 297–306.
LIPPMANN, R. 2002. Pattern classification using neural networks. *Communications Magazine, IEEE 27,* 11, 47–50.
LIU, Y., ZHANG, M., RU, L., AND MA, S. 2006. Automatic Query Type Identification Based on Click Through Information Information Retrieval Technology. In *AIRS'06*. Springer, Berlin, Germany, 593–600.
MACKAY, D. J. C. 1998. Introduction to Monte Carlo methods. In *Learning in Graphical Models*, M. I. Jordan, Ed. NATO Science Series. Kluwer Academic Press, Boston, MA, USA, 175–204.
MOFFAT, A. AND ZOBEL, J. 2008. Rank-biased precision for measurement of retrieval effectiveness. *ACM TOIS 27,* 1, 2:1–2:27.
OZERTEM, U., JONES, R., AND DUMOULIN, B. 2011. Evaluating new search engine configurations with pre-existing judgments and clicks. In *WWW'11*. ACM, New York, NY, USA, 397–406.
PRECUP, D., SUTTON, R., AND SINGH, S. 2000. Eligibility traces for off-policy policy evaluation. In *ICML'00*. ACM, New York, NY, USA, 759–766.
RADLINSKI, F., BENNETT, P. N., AND YILMAZ, E. 2011. Detecting duplicate web documents using clickthrough data. In *WSDM '11*. ACM, New York, NY, USA, 147–156.
RADLINSKI, F. AND CRASWELL, N. 2010. Comparing the sensitivity of information retrieval metrics. In *SIGIR '10*. ACM, New York, NY, USA, 667–674.
RADLINSKI, F. AND CRASWELL, N. 2013. Optimized interleaving for online retrieval evaluation. In *WSDM '13*.
RADLINSKI, F., KLEINBERG, R., AND JOACHIMS, T. 2008a. Learning diverse rankings with multi-armed bandits. In *ICML '08*. ACM, New York, NY, USA, 784–791.
RADLINSKI, F., KURUP, M., AND JOACHIMS, T. 2008b. How does clickthrough data reflect retrieval quality? In *CIKM '08*. ACM, New York, NY, USA, 43–52.
ROSE, D. E. AND LEVINSON, D. 2004. Understanding user goals in web search. In *WWW'04*. ACM, New York, NY, USA, 13–19.
SCHOLER, F., SHOKOUHI, M., BILLERBECK, B., AND TURPIN, A. 2008. Using clicks as implicit judgments: expectations versus observations. In *ECIR'08*. Springer, Berlin, Germany, 28–39.
SHEN, X., TAN, B., AND ZHAI, C. 2005. Context-sensitive information retrieval using implicit feedback. In *SIGIR '05*. ACM, New York, NY, USA, 43–50.
STREHL, A. M., LANGFORD, J., LI, L., AND KAKADE, S. M. 2010. Learning from logged implicit exploration data. In *Advances in Neural Information Processing Systems 23*, J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, Eds. NIPS '10. 2217–2225.
SUTTON, R. S. AND BARTO, A. G. 1998. *Introduction to Reinforcement Learning*. MIT Press,

Cambridge, MA, USA.
VOORHEES, E. M. AND HARMAN, D. K. 2005. *TREC: Experiment and Evaluation in Information Retrieval*. Digital Libraries and Electronic Publishing. MIT Press, Cambridge, MA, USA.
XUE, G., ZENG, H., CHEN, Z., YU, Y., MA, W., XI, W., AND FAN, W. 2004. Optimizing web search using web click-through data. In *CIKM '04*. Vol. 8. ACM, New York, NY, USA, 118–126.
YUE, Y., GAO, Y., CHAPELLE, O., ZHANG, Y., AND JOACHIMS, T. 2010. Learning more powerful test statistics for click-based retrieval evaluation. In *SIGIR '10*. ACM, New York, NY, USA, 507–514.
ZHANG, J. AND KAMPS, J. 2010. A search log-based approach to evaluation. In *ECDL'10*. Springer, Berlin, Germany, 248–260.

APPENDIX

A. PROOF OF THEOREM 4.1

THEOREM 4.1 *The estimator in Equation 2 is equal to two times the sample mean.*

PROOF. Below, we use the fact that $\frac{1}{n}\sum_{i=0}^{n} o_i = \frac{1}{n} wins(\mathbf{l}_2) - wins(\mathbf{l}_1)$ (following from the definition of $wins(\mathbf{l}_i)$ ($o = -1$ and $o = +1$ for $\mathbf{l}_1$ and $\mathbf{l}_2$ respectively) and $ties(\mathbf{l}_{1,2})$ ($o = 0$) (cf., §3), and that the number of samples is $n = wins(\mathbf{l}_1) + wins(\mathbf{l}_2) + ties(\mathbf{l}_{1,2})$.

$$\begin{aligned}
2\hat{E}_{wins} &= 2\left(\frac{wins(\mathbf{l}_2) + \frac{1}{2}ties(\mathbf{l}_{1,2})}{wins(\mathbf{l}_2) + wins(\mathbf{l}_1) + ties(\mathbf{l}_{1,2})} - 0.5\right) \\
&= 2\left(\frac{wins(\mathbf{l}_2) + \frac{1}{2}ties(\mathbf{l}_{1,2})}{n} - \frac{\frac{1}{2}n}{n}\right) \\
&= \frac{1}{n}\left(2\ wins(\mathbf{l}_2) + ties(\mathbf{l}_{1,2}) - (wins(\mathbf{l}_2) + wins(\mathbf{l}_1) + ties(\mathbf{l}_{1,2}))\right) \\
&= \frac{1}{n}\left(wins(\mathbf{l}_2) - wins(\mathbf{l}_1)\right) = \frac{1}{n}\sum_{i=0}^{n} o_i. \quad \square
\end{aligned}$$

B. PROOF OF THEOREM 5.1

THEOREM 5.1 *The following estimator is unbiased and consistent given samples from an interleaving experiment conducted according to the graphical model in Figure 1(b) (Eq. 6):*

$$\hat{E}[O] = \frac{1}{n}\sum_{i=1}^{n}\sum_{\mathbf{a}\in\mathbf{A}}\sum_{o\in O} oP(o|\mathbf{a}, \mathbf{c}_i)P(\mathbf{a}|\mathbf{l}_i, q_i).$$

PROOF. We start by defining a new function f:

$$f(\mathbf{C}, \mathbf{L}, Q) = \sum_{\mathbf{a}\in\mathbf{A}}\sum_{o\in O} oP(o|\mathbf{C}, \mathbf{a})P(\mathbf{a}|\mathbf{L}, Q).$$

Note that Eq. 6 is just the sample mean of $f(\mathbf{C}, \mathbf{L}, Q)$ and is thus an unbiased and consistent estimator of $E[f(\mathbf{C}, \mathbf{L}, Q)]$. Therefore, if we can show that $E[O] = E[f(\mathbf{C}, \mathbf{L}, Q)]$, that will imply that Eq. 6 is also an unbiased and consistent estimator of $E[O]$.

We start with the definition of $E[O]$:

$$E[O] = \sum_{o\in O} oP(o).$$

$P(O)$ can be obtained by marginalizing out the other variables:

$$P(O) = \sum_{\mathbf{a}\in\mathbf{A}}\sum_{\mathbf{c}\in\mathbf{C}}\sum_{\mathbf{l}\in\mathbf{L}}\sum_{q\in Q} P(\mathbf{a}, \mathbf{c}, \mathbf{l}, q, O),$$

where, according to the graphical model in Figure 1(b), $P(\mathbf{A}, \mathbf{C}, \mathbf{L}, Q, O) = P(O|\mathbf{C}, \mathbf{A})\ P(\mathbf{C}|\mathbf{L}, Q)\ P(\mathbf{L}|\mathbf{A}, Q)\ P(\mathbf{A})P(Q)$. Thus, we can rewrite $E[O]$ as

$$E[O] = \sum_{\mathbf{a}\in\mathbf{A}} \sum_{\mathbf{c}\in\mathbf{C}} \sum_{\mathbf{l}\in\mathbf{L}} \sum_{q\in Q} \sum_{o\in O} oP(o|\mathbf{a}, \mathbf{c})P(\mathbf{c}|\mathbf{l}, q)P(\mathbf{l}|\mathbf{a}, q)P(\mathbf{a})P(q).$$

Observing that $P(\mathbf{L}|\mathbf{A}, Q) = \frac{P(\mathbf{A}|\mathbf{L},Q)P(\mathbf{L}|Q)}{P(\mathbf{A}|Q)}$ (Bayes rule) and $P(\mathbf{A}|Q) = P(\mathbf{A})$ ($\mathbf{A}$ and Q are independent), gives us

$$E[O] = \sum_{\mathbf{a}\in\mathbf{A}} \sum_{\mathbf{c}\in\mathbf{C}} \sum_{\mathbf{l}\in\mathbf{L}} \sum_{q\in Q} \sum_{o\in O} oP(o|\mathbf{a}, \mathbf{c})P(\mathbf{a}|\mathbf{l}, q)P(\mathbf{c}|\mathbf{l}, q)p(\mathbf{l}|q)P(q).$$

Figure 1(b) implies $P(\mathbf{C}, \mathbf{L}, Q) = P(\mathbf{C}|\mathbf{L}, Q)P(\mathbf{L}|Q)P(Q)$, yielding:

$$E[O] = \sum_{\mathbf{a}\in\mathbf{A}} \sum_{\mathbf{c}\in\mathbf{C}} \sum_{\mathbf{l}\in\mathbf{L}} \sum_{q\in Q} \sum_{o\in O} oP(o|\mathbf{a}, \mathbf{c})P(\mathbf{a}|\mathbf{l}, q)P(\mathbf{c}, \mathbf{l}, q).$$

From the definition of $f(\mathbf{C}, \mathbf{L}, Q)$ this gives us:

$$E[O] = \sum_{\mathbf{c}\in\mathbf{C}} \sum_{\mathbf{l}\in\mathbf{L}} \sum_{q\in Q} f(\mathbf{c}, \mathbf{l}, q)P(\mathbf{c}, \mathbf{l}, q),$$

which is the definition of $E[f(\mathbf{C}, \mathbf{L}, Q)]$, so that:

$$E[O] = E[f(\mathbf{C}, \mathbf{L}, Q)]. \quad \square$$

C. PROOF OF THEOREM 5.2

THEOREM 5.2 *The following estimator is unbiased given samples from an interleaving experiment conducted according to the graphical model in Figure 1 under P_S:*

$$\hat{E}_T[O] = \frac{1}{n} \sum_{i=1}^{n} \sum_{a\in A} \sum_{o\in O} oP(o|c_i, a)P(a|l_i, q_i) \frac{P_T(l_i|q_i)}{P_S(l_i|q_i)}.$$

PROOF. As in Theorem 5.1, we start by defining f:

$$f(C, L, Q) = \sum_{\mathbf{a}\in\mathbf{A}} \sum_{o\in O} oP(o|\mathbf{a}, \mathbf{C})P(\mathbf{a}|\mathbf{L}, Q).$$

Plugging this into the importance sampling estimator in Eq. 1 gives:

$$\hat{E}_T[O] = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{c}_i, \mathbf{l}_i, q_i) \frac{P_T(\mathbf{c}_i, \mathbf{l}_i, q_i)}{P_S(\mathbf{c}_i, \mathbf{l}_i, q_i)},$$

which is unbiased and consistent if $P_S(\mathbf{C}, \mathbf{L}, Q)$ is non-zero at all points at which $P_T(\mathbf{C}, \mathbf{L}, Q)$ is non-zero. Figure 1(b) implies that $P(\mathbf{C}, \mathbf{L}, Q) = P(\mathbf{C}|\mathbf{L}, Q)P(\mathbf{L}|Q)P(Q)$, yielding:

$$\hat{E}_T[O] = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{c}_i, \mathbf{l}_i, q_i) \frac{P_T(\mathbf{c}_i|\mathbf{l}_i, q_i)P_T(\mathbf{l}_i|q_i)P_T(q_i)}{P_S(\mathbf{c}_i|\mathbf{l}_i, q_i)P_S(\mathbf{l}_i|q_i)P_S(q_i)}.$$

Because we assume that clicks and queries are drawn from the same static distribution, independent of the ranker pair used to generate the presented list, we know that $P_T(Q) = P_S(Q)$ and $P_T(\mathbf{C}|\mathbf{L}, Q) = P_S(\mathbf{C}|\mathbf{L}, Q)$, giving us:

$$\hat{E}_T[O] = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{c}_i, \mathbf{l}_i, q_i) \frac{P_T(\mathbf{l}_i|q_i)}{P_S(\mathbf{l}_i|q_i)}.$$

From the definition of $f(\mathbf{C}, \mathbf{L}, Q)$ we obtain:

$$\hat{E}_T[O] = \frac{1}{n} \sum_{i=1}^{n} \sum_{\mathbf{a} \in \mathbf{A}} \sum_{o \in O} oP(o|\mathbf{a}, \mathbf{c}_i) P(\mathbf{a}|\mathbf{l}_i, q_i) \frac{P_T(\mathbf{l}_i|q_i)}{P_S(\mathbf{l}_i|q_i)}.$$

To show that $P_S(\mathbf{C}, \mathbf{L}, Q)$ is non-zero whenever $P_T(\mathbf{C}, \mathbf{L}, Q)$ is non-zero, we need only show that $P_S(\mathbf{L}|Q)$ is non-zero at all points at which $P_T(\mathbf{L}|Q)$ is non-zero. This follows from three facts already mentioned above: 1) $P(\mathbf{C}, \mathbf{L}, Q) = P(\mathbf{C}|\mathbf{L}, Q)P(\mathbf{L}|Q)P(Q)$, 2) $P_T(Q) = P_S(Q)$, and 3) $P_T(\mathbf{C}|\mathbf{L}, Q) = P_S(\mathbf{C}|\mathbf{L}, Q)$. Figure 1(b) implies that $P(\mathbf{L}|Q) = \sum_{\mathbf{a} \in \mathbf{A}} P(\mathbf{L}|\mathbf{a}, Q)$ (Eq. 9), which is non-zero if $P(\mathbf{L}|\mathbf{A}, Q)$ is non-zero for at least one assignment. From the definition of the interleaving process (Eq. 8) we have that $P_S(\mathbf{L}|\mathbf{A}, Q)$ is non-zero for all assignments. □