



# Recall, Robustness, and Lexicographic Evaluation

FERNANDO DIAZ, Carnegie Mellon University, Pittsburgh, United States

MICHAEL D. EKSTRAND, Information Science, Drexel University, Philadelphia, United States

BHASKAR MITRA, Research, Microsoft Corp, Montréal, Canada

Although originally developed to evaluate *sets* of items, recall is often used to evaluate *rankings* of items, including those produced by recommender, retrieval, and other machine learning systems. The application of recall without a formal evaluative motivation has led to criticism of recall as a vague or inappropriate measure. In light of this debate, we reflect on the measurement of recall in rankings from a formal perspective. Our analysis is composed of three tenets: recall, robustness, and lexicographic evaluation. First, we formally define “recall orientation” as the sensitivity of a metric to a user interested in finding every relevant item. Second, we analyze recall orientation from the perspective of robustness with respect to possible content consumers and providers, connecting recall to recent conversations about fair ranking. Finally, we extend this conceptual and theoretical treatment of recall by developing a practical preference-based evaluation method based on lexicographic comparison. Through extensive empirical analysis across multiple recommendation and retrieval tasks, we establish that our new evaluation method, *lexirecall*, has convergent validity (i.e., it is correlated with existing recall metrics) and exhibits substantially higher sensitivity in terms of discriminative power and stability in the presence of missing labels. Our conceptual, theoretical, and empirical analysis substantially deepens our understanding of recall and motivates its adoption through connections to robustness and fairness.

CCS Concepts: • **Hardware** → **Robustness**; • **General and reference** → **Evaluation**;

Additional Key Words and Phrases: Evaluation, recall, robustness

## ACM Reference Format:

Fernando Diaz, Michael D. Ekstrand, and Bhaskar Mitra. 2026. Recall, Robustness, and Lexicographic Evaluation. *ACM Trans. Recomm. Syst.* 4, 1, Article 13 (July 2026), 50 pages. <https://doi.org/10.1145/3728373>

## 1 Introduction

Researchers use “recall” to evaluate rankings across a variety of retrieval, recommendation [126], and machine learning tasks [40, 86, 88]. “Recall at  $k$  items” ( $R_k$ ) and **R-precision (RP)** are popular metrics used for measuring recall in rankings. Since the beginning of the ACM Conference on Recommender Systems, on average one third of full papers measure recall in experiments (Figure 1). While there is a colloquial interpretation of recall as measuring coverage (as it might be rightfully interpreted in set retrieval), the research community is far from a principled understanding of recall

Fernando Diaz work done at Google.

Authors’ Contact Information: Fernando Diaz, Carnegie Mellon University, Pittsburgh, Pennsylvania, United States; e-mail: [diazf@acm.org](mailto:diazf@acm.org); Michael D. Ekstrand, Information Science, Drexel University, Philadelphia, Pennsylvania, United States; e-mail: [mdekstrand@drexel.edu](mailto:mdekstrand@drexel.edu); Bhaskar Mitra, Research, Microsoft Corp, Montréal, Quebec, Canada; e-mail: [bmitra@microsoft.com](mailto:bmitra@microsoft.com).



This work is licensed under a Creative Commons Attribution 4.0 International License.

© 2026 Copyright held by the owner/author(s).

ACM 2770-6699/2026/07-ART13

<https://doi.org/10.1145/3728373>

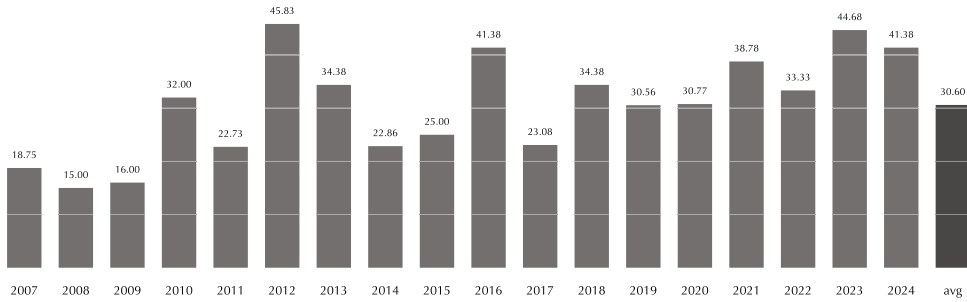


Fig. 1. Percentage of full papers published at the ACM Conference on Recommender Systems that measure recall in experiments. Percentages are macro-averaged across years to control for growth in the number of full papers in the conference proceedings. Details can be found in Appendix A. Please review the alt-text you provided for figures in your article and confirm that it is correct. If you have not provided alt-text for the figures in your article, please note that ACM strongly recommends that you add alt-text to the proof. Alt-text is a brief description of the image that is read by screen readers or other assistive technology. It provides information about an image’s purpose for the reader and provides context on how the image relates to the page content. If you are adding alt-text for any figure, please add the alt-text [via vendor-defined mechanism] in the proof.

metrics for rankings. Nevertheless, authors continue to informally refer to evaluation metrics as more or less “recall oriented” or “precision oriented” without a formal definition of what this means or quantifying how existing metrics relate to these constructs [25, 27, 55, 60, 73, 74].

Given the prevalence of recall as a metric in recommender system research, understanding recall conceptually, theoretically, and empirically is fundamental to sound evaluation. Indeed, the lack of a principled understanding of or motivation for recall has caused some to question whether recall is useful as a construct at all. Herlocker et al. [50] suggests that the sparse nature of recommendation data may make recall inappropriate. Cooper [22] argues that recall orientation is inappropriate because user search satisfaction depends on the number of items the user is looking for, which may be fewer than *all* of the relevant items. Zobel et al. [130] refute several informal justifications for recall: persistence (the depth a user is willing to browse), cardinality (the number of relevant items found), coverage (the number of user intents covered), density (the rank-locality of relevant items), and totality (the retrieval of all relevant items). So, while many ranking experiments compute recall metrics, precisely what and why we are measuring remains unclear.

In this light, we approach the measurement of recall in rankings from a formal perspective, with an objective of proposing a new interpretation of recall with precise conceptual and theoretical grounding. Our analysis is composed of three interrelated tenets: recall, robustness, and lexicographic evaluation. First, we consider recall an essentially contested construct [41, 53]: It is a high-level construct believed to be important but with different, conflicting interpretations, as suggested by Zobel et al. [130]. By adopting the interpretation of recall as finding the totality of relevant items, we formally define “recall orientation” as sensitivity to movement of the bottom-ranked relevant item. Although simple, this definition of recall connects to both early work in position-based evaluation as well as recent work in technology-assisted review. Moreover, by formally defining recall orientation, we can design a new metric, total search efficiency, that precisely measures recall. Second, we consider robustness another essentially contested construct, again with different, conflicting interpretations [32]. By adopting an interpretation of robustness as the effectiveness for the worst-off user, we can connect it to our notion of recall orientation. We demonstrate that recall—and totality specifically—is aligned with worst-case robustness. Finally, we extend this conceptual and theoretical treatment of recall by developing a practical

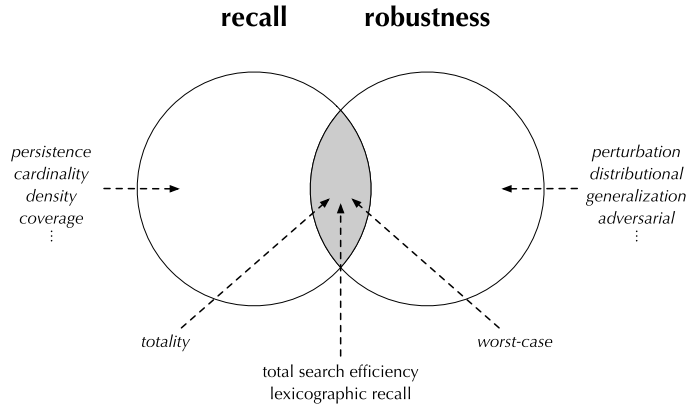


Fig. 2. Both recall and robustness as a theoretical construct can be conceptualized in multiple ways [32, 130]. These two constructs are equivalent when conceptualized as totality (recall) and worst-case analysis (robustness). Lexicographic recall is located in this intersection.

preference-based evaluation method based on lexicographic comparison. We present a conceptual relationship between recall, robustness, and lexicographic evaluation in Figure 2. Through extensive empirical analysis across various retrieval and recommendation tasks, we establish that our new evaluation method, lexicographic recall or *lexirecall*, is correlated with existing recall metrics but exhibits substantially higher discriminative power and stability in the presence of missing labels. While conceptually and theoretically grounded in notions of robustness and fairness, *lexirecall* pragmatically is appropriate when we are concerned with understanding system behavior for users who are focused on finding all relevant items in the same way that experiments adopt reciprocal rank as a high-precision metric. This article, by focusing on worst-case performance, also complements existing work that has focused on average case [29] and best-case [28] preference-based evaluation. Our conceptual, theoretical, and empirical analysis substantially deepens our understanding of recall as a construct and motivates its adoption through connections to robustness and fairness constructs.

## 2 Preliminaries

We begin by defining our core concepts and notation to provide a clear foundation for our analysis. While many of these concepts will be familiar to those with a background in ranking evaluation, we adopt a specific mathematical framework that will be important when proving properties of recall, robustness, and lexicographic evaluation.

We consider ranking systems that are designed to satisfy users<sup>1</sup> with with some information need, broadly construed. This information need may be specific or vague and may or may not be explicit even in the user’s own mind (e.g., “entertain me for the evening” can be considered an information need for home-page recommendations on a video streaming service; “what happened yesterday that I should know about?” for a daily e-mail of recommended new articles from a news publisher). Although users in many recommendation scenarios may not seem to have an information need, they do, at any point in time, have a latent order over items in the catalog. This is true in relatively passive scenarios like radiolike streaming audio where a user, although not

<sup>1</sup>We adopt *user* as a general term for searcher in the information retrieval context and content consumer in the multi-stakeholder recommendation context.

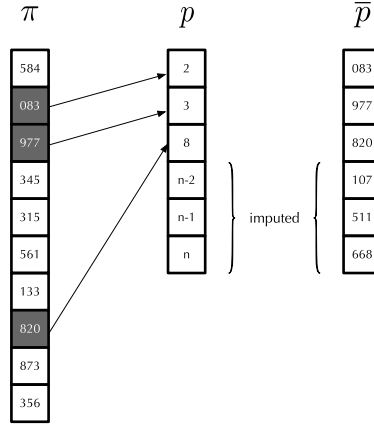


Fig. 3. Relevance projection with imputation. Given a set of relevant item ids  $\mathcal{R} = \{083, 107, 511, 668, 820, 977\}$ , relevance projection of an incomplete top-10 ranking  $\pi$  and relevant set  $\mathcal{R}$  to a  $m \times 1$  vector of positions  $p$ . We also show the inverse projection vector  $\bar{p}$  of items at specific recall levels.

explicitly expressing or thinking about their preferences, nevertheless reveals them through their consumption behavior.

While a user's information need is never directly revealed to the ranking system, the user expresses an observable *request* to the ranking system. A request can include information provided by the user either explicitly (e.g., a query or question for text-based search; an application or site context for recommendation) or implicitly (e.g., geo-location for text-based search; engagement or session history for recommendation [61]) or both [102, 124].

A system attempts to satisfy an information need by ranking all of the items in a corpus  $\mathcal{D}$ . A corpus might consist of text documents (e.g., a web crawl) or cultural media (e.g., a music catalog). And so, if  $n = |\mathcal{D}|$ , then a ranking system is a function that, given a request, produces a permutation of the  $n$  items in the collection. As such, the space of possible system outputs is the set of all permutations of  $n$  items, also known as the symmetric group of degree  $n$  or  $S_n$ .

The objective of ranking evaluation is to determine the quality of a permutation  $\pi \in S_n$  for the user. In the remainder of this section, we will detail precisely how we do this.<sup>2</sup>

## 2.1 Relevance

The relevance of an item refers to its value with respect to a user's information need. In this work, we focus on binary relevance, an approach regularly used in information retrieval and recommender system literature, especially when measuring recall.<sup>3</sup> Let  $\mathcal{R} \subset \mathcal{D}$  be the set of items labeled relevant to the request, where  $m = |\mathcal{R}|$ .

Given a ranking  $\pi$ , then we define  $p$  as the  $m \times 1$  vector where  $p_i$  is the position of the  $i$ th ranked relevant item in  $\pi$ . Each index  $i \in [1, m]$ , referred to as the *recall level*, represents how many relevant items have been found at position  $p_i$ . We present an example of how to construct  $p$  in Figure 3. We will use  $\bar{p}$  to represent the  $m \times 1$  vector where  $\bar{p}_i \in \mathcal{D}$  is the identity of the  $i$ th

<sup>2</sup>While ranking does not cover all recommendation scenarios, it covers many, including personalized search, home-page recommendations, related product recommendations, and social media timeline ranking, and is frequently considered in research; further, nonranking interfaces such as music or video streams may well be implemented as a final selection or transformation of an underlying (partial) ranking, and therefore better understanding of ranking evaluation may be applicable even to recommendation surfaces that do not directly expose the ranking to the user.

<sup>3</sup>We discuss how our analysis extends in ordinal grades and preferences in Section 7.2.3.

ranked relevant item. There are a total of  $\binom{n}{m}$  unique  $p$  and each unique  $p$  corresponds to a subset of  $C = m!(n - m)!$  unique permutations in  $S_n$ .

## 2.2 Permutation Imputation

Many ranking systems only provide a ranking on the top  $\tilde{n} \ll n$  items, which may not include all relevant items, leaving elements of  $p$  undefined. To use many metrics, especially recall-oriented metrics, we need to impute the positions of the unranked relevant items. An *optimistic imputation* would place the unranked relevant items immediately after the last retrieved items (i.e.,  $\tilde{n} + 1, \tilde{n} + 2, \dots$ ). Such a protocol would be susceptible to manipulation by returning few or no items. Alternatively, we consider *pessimistic imputation*, placing the unretrieved relevant items at the bottom of the total order over the corpus. For example, if a system returns only three of six relevant items in the top  $\tilde{n}$  at positions 2, 3, and 8, then we would define  $p$  as

$$p = \underbrace{2, 3, 8}_{\text{top } \tilde{n}}, \underbrace{n - 2, n - 1, n}_{\text{bottom } n - \tilde{n}}.$$

Pessimistic imputation is a conservative placement of the unretrieved relevant items and is well aligned with our interest in robust performance. Moreover, it is consistent with behavior of metrics like rank-biased precision [72], which implicitly applies an exposure of 0 for unretrieved relevant items (i.e., for large  $n$ ,  $\lim_{i \rightarrow n} \gamma^i \approx 0$ ); and average precision,<sup>4</sup> which implicitly applies an exposure of 0 for unretrieved relevant items (i.e., for large  $n$ ,  $\lim_{i \rightarrow n} \frac{1}{i} \approx 0$ ).<sup>5</sup> We show an example of projection with imputation in Figure 3.

## 2.3 Measuring Effectiveness

When evaluating a ranking system, we consider two sets of important stakeholders: users and providers. Users approach the system with information needs and requests and ultimately define what is relevant. Providers contribute items to the search system that serve to satisfy users. Each item in the corpus is attributable, explicitly or not, to a content provider. We include providers in our treatment to draw connections in existing work in the fair ranking literature, which often focuses on providers as stakeholders [35]. In this section, we will characterize a broad family of evaluation metrics for these two sets of users that will allow us, in subsequent sections, to define formal notions of recall and robustness.

**2.3.1 Measuring Effectiveness for Users.** For a fixed information need and ranking  $\pi$ , an evaluation metric is a function that scores rankings,  $\mu : S_n \times \mathcal{D}^+ \rightarrow \mathbb{R}^*$ , where  $\mathcal{D}^+$  is set of all subsets of  $\mathcal{D}$  excluding the empty set. An evaluation metric, then, is a function whose domain is the joint space of all corpus permutations and possible relevance judgments and whose range is a non-negative scalar value. We are specifically interested in a class of metrics that can be expressed in terms of a summation over recall levels.

**Definition 2.1.** Given a ranking  $\pi \in S_n$  and relevant items  $\mathcal{R} \in \mathcal{D}^+$ , a *recall-level metric* is an evaluation metric defined as a summation over  $m$  recall levels,

$$\mu(\pi, \mathcal{R}) = \sum_{i=1}^m e(p_i)z(i, m), \quad (1)$$

<sup>4</sup>As defined in trec\_eval.

<sup>5</sup>Although recall is often normalized for top  $\tilde{n}$  rankings by dividing by  $\min(\tilde{n}, m)$ , this practice results in measuring precision instead of recall for information needs where  $\tilde{n} < m$ . To compare rankings of different lengths and to emphasize recall measurement, we focus on imputed full rankings.

where  $e : \mathbb{Z}^+ \rightarrow \mathbb{R}^*$  is a strictly monotonically decreasing *exposure function* proportional to the probability that the user reaches rank position  $i$  in their scan of the list; and  $z : \mathbb{Z}^+ \times \mathbb{Z}^+ \rightarrow \mathbb{R}^*$  is a metric-specific *normalization function* of recall level and size of  $\mathcal{R}$ .

The product  $e(p_i)z(i, m)$  is a decomposition of what [Carterette](#) refers to as a “discount function” into an explicit function that models exposure and another that addresses any recall normalization [12]. Within the set of recall-level metrics, we are further interested in the sub-class of metrics that satisfy the following criteria for “top-heaviness.”

*Definition 2.2.* We refer to a recall-level metric as *top-heavy* if, for  $j \in [0 \dots m)$ ,

$$\sum_{i=1}^m e(p_i)z(i, m) \geq \sum_{i=j+1}^m e(p_i)z(i-j, m-j).$$

Top-heaviness indicates that, in the event that there are  $j$  unjudged, relevant items in positions above the remaining  $m-j$  relevant items, the metric computed over all  $m$  items must be greater than or equal to the metric value computed over only the  $m-j$  items. Because we deal with metrics that include functions of  $m$ , this is not an obvious property, but one that will be important as we consider incomplete judgments and relationships between possible users in Section 4.<sup>6</sup>

Top-heavy recall-level metrics are a precise subclass of discounted metrics, covering a broad class of existing metrics such as **average precision (AP)**, **reciprocal rank (RR)**, **normalized discounted cumulative gain (NDCG)**, and **rank-biased precision (RBP)**, where we define the exposure and normalization as

$$\begin{aligned} e_{\text{AP}}(i) &= \frac{1}{i} & z_{\text{AP}}(i, m) &= \frac{i}{m} \\ e_{\text{RR}}(i) &= \frac{1}{i} & z_{\text{RR}}(i, m) &= \begin{cases} 1 & \text{if } i = 1 \\ 0 & \text{otherwise} \end{cases} \\ e_{\text{NDCG}}(i) &= \frac{1}{\log_2(i+1)} & z_{\text{NDCG}}(i, m) &= \left( \sum_{k=1}^m \frac{1}{\log_2(k+1)} \right)^{-1} \\ e_{\text{RBP}}(i) &= (1-\gamma)\gamma^{i-1} & z_{\text{RBP}}(i, m) &= 1. \end{aligned}$$

Beyond classic ranking metrics, top-heavy recall-level metrics include nontraditional metrics such as those based on linear discounting (e.g.,  $e_{\text{lin}}(i) = 1 - \frac{i}{n}$ ). This results in a much broader class of metrics than those normally considered, for example, in the formal analysis of ranking metrics [1, 38, 71]. As a result, while all top-heavy recall-level metrics satisfy some formal properties of ranking evaluation metrics, large subsets of top-heavy recall-level metrics may satisfy more. A detailed analysis of formal properties of top-heavy recall-level metrics can be found in Appendix B. As mentioned, we can contrast this with [Carterette](#)’s decomposition that focuses on the decomposition of metrics into gain and discount components [12]. In our case, we do not model gain, since we deal with binary relevance. Our exposure and normalization functions, then, precisely define a subset of [Carterette](#)’s discount functions that do not fit into his metric taxonomy since they do not consider recall normalization [12].

We focus on this class of metrics to prove properties of robustness in Section 4.

**2.3.2 Measuring Effectiveness for Providers.** For content providers, we define the utility they receive from a ranking  $\pi$  as a function of their items’ *cumulative positive exposure*, defined as

<sup>6</sup>For more information on the relationship between incomplete judgments and metric stability, see References [10, 91, 111].

exposure of a provider's relevant content.<sup>7</sup> Let  $\mathcal{R}' \subseteq \mathcal{R}$  be the subset of relevant items belonging to a specific provider. Since  $e$  captures the likelihood that a user inspects a specific rank position, we can compute the cumulative positive exposure as

$$\eta_e(\pi, \mathcal{R}, \mathcal{R}') = \sum_{i=1}^m e(p_i) \mathbb{I}(\bar{p}_i \in \mathcal{R}'), \quad (2)$$

where  $m$  and  $p$  are based on  $\mathcal{R}$ . Unless necessary, we will drop the subscript  $e$  from  $\eta$  for clarity.

## 2.4 Evaluation Method Desiderata

Because there is no consensus on a single approach to validate a new evaluation method, we assemble desired theoretical and empirical properties of a method drawn from work in information retrieval and recommender system evaluation [90, 111] and measurement theory [53, 121].

- **Validity**
  - *Content validity*.
    - \* Is the evaluation theoretically related to the higher-level concept? (Sections 4.1, 4.2)
    - \* Is the evaluation better correlated with the higher-level concept than existing metrics? (Section 4.3)
  - *Convergent validity*. Is the evaluation method empirically *correlated* with existing methods for measuring the *same* higher-level concept? (Section 6.2.1)
  - *Discriminant validity*. Is the evaluation method empirically *uncorrelated* with existing methods for measuring *different* higher-level concepts? (Section 6.2.1)
- **Sensitivity**
  - *Decision sensitivity*. Is the evaluation method better able to distinguish between *rankings* compared to existing methods? (Section 6.2.2)
  - *System sensitivity*. Is the evaluation method better able to distinguish between *rankers* compared to existing methods? (Section 6.2.3)
- **Reliability**
  - *Stable validity*. Is the evaluation method stable when labels are missing compared to existing methods? (Section 6.2.4)
  - *Stable sensitivity*. Does the evaluation method maintain sensitivity when labels are missing compared to existing methods? (Section 6.2.4)

Throughout this article, when we assess or compare evaluation methods, we will focus on these properties.

We note that the evaluation methods we develop in Sections 3 and 5 measure population-based properties using worst-case analysis. This means that, for each request, these methods consider a *population of users* that tend to emphasize *under-represented* behaviors and intents. We contrast this with traditional metrics that model *individual users* and emphasize *well-represented* behaviors and intents. This means that validation with, for example, behavioral feedback [13] or user studies [93] is not possible. In lieu of empirical validation, we emphasize both conceptual and theoretical properties of our evaluation methods, grounding them in the relevant work in philosophy and

<sup>7</sup>We do not consider provider utility when none of their associated items are relevant to the user's information need. Although not covering situations where providers benefit from *any* exposure (including of nonrelevant content), it is consistent with similar definitions used in the fair ranking literature [30, 100].

While we adopt a cumulative exposure model in this work, alternative notions of provider effectiveness are possible. For example, normalizing by the number of relevant items contributed  $|\mathcal{R}'|$  would emphasize providers who contribute more content.



economics. This normative design of an evaluation method is consistent with recent work in the recommender system community [39, 115–117].

### 3 Recall

As mentioned in Section 1, the description of ranked ranking metrics as “recall oriented” remains poorly defined, leaving the formal analysis of metrics for recall orientation difficult. From a technical point of view, some work considers recall orientation to be a binary criterion, dependent on whether a metric includes the recall base (i.e.,  $\mathcal{R}$ ) to be computed [55, 89]. This would include metrics that compute set-based recall at some rank cutoff [24, 109] as well as metrics like AP and NDCG. Using a set-based recall metric is particularly well suited for recall orientation in early stages of multi-stage ranking [63, 73]. A binary notion of recall orientation does not capture that some metrics may be more recall oriented than others. This is captured, in part, by references to recall orientation as related to the depth in the ranking considered by the user [27]. More frequently, authors appeal to metrics like AP and  $R_{1000}$  as being recall oriented without clear discussion of what this means [25, 47, 60, 74]. However, both Mackie et al. [64] and [66] refer to  $R_{1000}$  as recall oriented but AP being precision oriented. In light of the lack of consensus on recall orientation, in Section 3.1, we propose a new quantitative view of recall orientation based on how sensitive a metric is for a user interested in finding every relevant item. This allows us to see recall orientation along a spectrum and compare the degrees of recall orientation of different metrics. In Section 3.2, based on this definition, we derive a new recall metric, total search efficiency.

#### 3.1 Metric Orientation

We are interested in more precisely defining precision and recall as constructs to be measured in ranking evaluation. Although most evaluation metrics colloquially capture some aspects of both precision and recall, understanding the sensitivity to each remains vague. We can address this vagueness by approaching precision and recall as two extremes of recall requirements. At one extreme, precision as a construct reflects the satisfaction of a user who only needs exactly one relevant item, the minimum amount of retrievable content. We might find this in domains like web search. At the other extreme, recall as a construct reflects the satisfaction of a user who needs *every* relevant item, the maximum amount of retrievable content. Herlocker et al. [50] call this the “find all good items” tasks; Zobel et al. [130] refers to this as the *totality* interpretation of recall, found in many technology-assisted review domains. Indeed, this perspective is supported by tasks like recommender systems for scholarly literature reviews, evaluation programs like the TREC Total Recall Track [85] and patent search [62], and by metrics like “position of the last relevant item” [131].

We begin by defining the *precision valence* of a ranking of  $n$  items as how efficiently a user can find the *first* relevant item. For a fixed request, assume that we have  $m$  relevant items. The ideal precision valence occurs when the first relevant item is at rank position 1. The worst precision valence occurs when the first relevant item is at rank position  $n - m + 1$ , just above the remaining  $m - 1$  relevant items. Similarly, we refer to the *recall valence* of a ranking as how efficiently a user can find *all* of the relevant items. The ideal recall valence occurs when the last relevant item is at position  $m$  (i.e., below the other  $m - 1$  relevant items) and the worst precision valence when it is at position  $n$ .

To define the *precision orientation* of a metric, we measure the difference in the best-case precision valence and worst-case precision valence for a given metric. Although there is only one arrangement of positions of relevant items where the top-ranked item is at position  $n - m + 1$ , there are  $\binom{n-1}{m-1}$  arrangements of positions of relevant items where the top-ranked item is at position 1. To control for the contribution of higher recall levels, we can consider, for the best-case



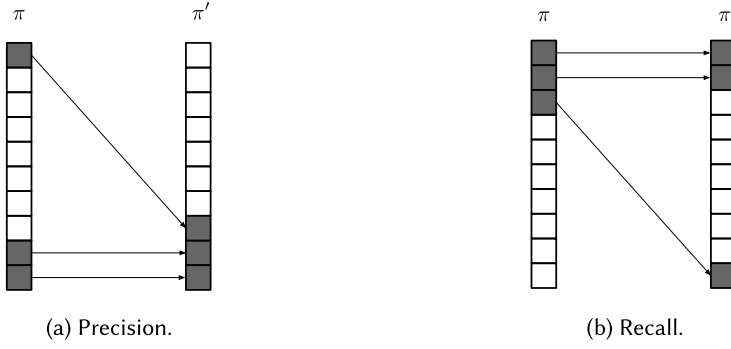


Fig. 4. Metric orientation. Each ranking  $\pi$  of 10 items represented with a vector of cells ordered from top to bottom with shaded cells representing relevant items  $\mathcal{R}$ . *Precision orientation* (left) measures the degradation in a metric when the highest ranked relevant item is moved to the bottom of the ranking while holding all other positions fixed. *Recall orientation* (right) measures the degradation in a metric when the lowest ranked relevant item is moved to the bottom of the ranking while holding all other positions fixed. We measure the precision and recall orientation of a metric  $\mu$  by the difference between  $\mu(\pi, \mathcal{R}) - \mu(\pi', \mathcal{R})$ .

precision valence, the ranking with a relevant item at the first position and the remaining  $m - 1$  relevant items at the bottom of the ranking. Precision orientation, then, measures sensitivity for a user interested in one relevant item. We depict this graphically in Figure 4(a). Similarly, to define the *recall orientation* of a metric, we measure the difference in the best-case recall valence and worst-case recall valence for a given metric. Although there is only one arrangement of positions of relevant items where the bottom-ranked item is at position  $m$ , there are  $\binom{n-1}{m-1}$  arrangements of positions of relevant items where the bottom-ranked item is at position  $n$ . To control for the contribution of lower recall levels, we can consider, for the worst-case recall valence, the ranking with a relevant item at position  $n$  and the remaining  $m - 1$  relevant items at the top of the ranking. Recall orientation, then, measures sensitivity for a user interested in all relevant items. We depict this graphically in Figure 4(b).

To understand the intuition behind this definition of metric orientation, we can think about the recall requirements of precision-oriented or recall-oriented users. The prototypical precision-oriented user is satisfied by a single relevant item. Precision orientation quantifies how sensitive a metric is at measuring the best case and worst case for this precision-oriented user. The prototypical recall-oriented user is only satisfied when they find all of the relevant items. Recall orientation quantifies how sensitive a metric is at measuring the best case and worst case for this recall-oriented user. These prototypical users intentionally represent extremes to characterize existing metrics and control for any contribution from other recall requirements (e.g., those greater than one for precision and less than  $m$  for recall).

Figure 5 shows the recall and precision orientation for several well-known evaluation metrics for a ranking of  $n = 10^5$  items. Because both precision and recall orientation are functions of the number of relevant items, we plot values for  $m \in [1 \dots 15]$ .

In terms of precision orientation, the ordering of metrics follows conventional wisdom. RR is often used for known-item or other high-precision tasks where the user is satisfied by the first relevant item. Across all values of  $m$ , RR dominates other metrics. NDCG, often used for evaluating both top- $N$  recommendations and web search results, is the next most precision-oriented metric.<sup>8</sup>

<sup>8</sup>We note that the classification of NDCG as precision oriented could be influenced by different discounting functions, as suggested by Jeunen et al. [54].

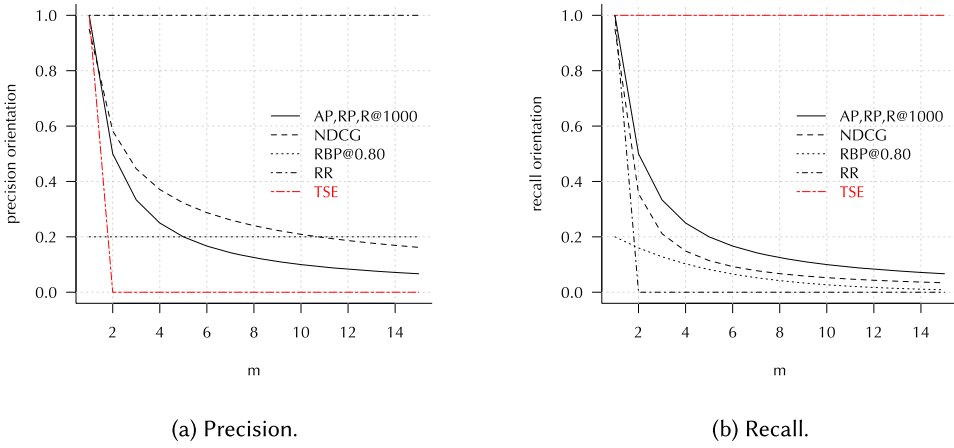


Fig. 5. Metric orientation of ranking metrics of  $10^5$  items with  $m \in [1 \dots 15]$  relevant items. The vertical axis reflects, for  $m \in [1, 15]$ , the change in metric value when (a) moving top-ranked item from position 1 to position  $n - m + 1$  or (b) moving bottom-ranked item from position  $m$  to position  $n$ . The values for TSE (Equation (3)) are scaled by the lower and upper bound given a fixed  $m$  and therefore apply to any exposure model. See Figure 4 for details. *This figure best rendered in color.*

AP, “recall at 1000” ( $R_{1000}$ ), and RP all have the same precision orientation and are dominated by NDCG. RBP dominates NDCG and AP once we reach a modest number of relevant items. Both RR and RBP are not sensitive to the number of relevant items because neither is a function of  $m$ .

In terms of recall orientation, the ordering of metrics follows conventional wisdom in the information retrieval and recommender systems community.  $R_{1000}$ , AP, and RP all dominate other metrics for all values of  $m$ . This family of metrics is followed by metrics often used for precision tasks, NDCG and RBP. RR, the least recall-oriented metric, only considers the top-ranked relevant item and shows *no* recall orientation unless there is only one relevant item. We should note that, except for RR, the recall orientation decreases with the number of relevant items because all of these metrics aggregate an increasing number of positions as  $m$  increases. When evaluating over a set of requests with varying values of  $m$ , mis-calibrated recall valences may result in requests with lower values of  $m$  dominating any averaging.

In this analysis, RR is a precision-oriented “basis” metric insofar as it is only dependent on the position of the highest-ranked relevant item. We are interested in designing a symmetric recall-oriented “basis” metric that only depends on the position of the lowest-ranked relevant item. We depict this desired metric as the red line in Figure 5. Traditional recall-oriented metrics do not satisfy this since they depend on the position of the higher-ranked relevant items, especially as  $m$  grows. In this article, we identify the missing metric that captures recall orientation while being well-calibrated across values of  $m$ .

### 3.2 Total Search Efficiency

Although metrics such as AP are often referred to as “recall oriented,” in this section, we focus on metrics that explicitly define recall as a construct. Such recall metrics for ranking systems come in two flavors.<sup>9</sup> The first flavor of recall metrics measures the fraction of relevant items found after a

<sup>9</sup>We exclude set-based metrics used in set-based retrieval or some technology-assisted review evaluation [23, 59], since they require systems to provide a cutoff in addition to a ranking.

user terminates their scan of the ranked list. Metrics like  $R_{1000}$  and RP use a model of search depth to simulate how deep a user will scan. We can define exposure and normalization functions for  $R_k$  and RP,

$$\begin{aligned} e_{R_k}(i) &= \mathbf{I}(i \leq k) & z_{R_k}(j, m) &= \frac{1}{m} \\ e_{RP}(i) &= \mathbf{I}(i \leq m) & z_{RP}(j, m) &= \frac{1}{m}. \end{aligned}$$

Note that, although we decompose these metrics using the notation of recall-level metrics, neither of these exposure functions strictly monotonically decrease in rank, so they are not recall-level metrics.

The second flavor of recall metrics measures the effort to find all  $m$  relevant items. Cooper [22] refers to this as the *Type 3 search length* and is measured by

$$\begin{aligned} \text{SL}_3(\pi, \mathcal{R}) &= p_m - m \\ &\stackrel{\text{rank}}{=} p_m. \end{aligned}$$

Similarly, Zou and Kanoulas [131] use “position of the last relevant item” to evaluate high-recall tasks. By contrast, Rocchio [84] proposed *recall error*, a metric based on the average rank of relevant items,

$$\begin{aligned} \text{RE}(\pi, \mathcal{R}) &= \frac{1}{m} \sum_{i=1}^m p_i - \frac{m+1}{2} \\ &\stackrel{\text{rank}}{=} \sum_{i=1}^m p_i. \end{aligned}$$

Recall error can be sensitive to outliers at very low ranks, which occur frequently in even moderately sized corpora [65].

Inspired by Cooper’s  $\text{SL}_3$ , we define a new recall-oriented top-heavy recall-level metric by looking at exposure of relevant items at highest recall level (i.e.,  $i = m$  in Equation (1)). We can define a recall-oriented metric based on any top-heavy recall-level metric by replacing its normalization function with the following:

$$z_{\text{SL}_3}(i, m) = \begin{cases} 1 & \text{if } i = m \\ 0 & \text{otherwise} \end{cases}.$$

We refer to this as the efficiency of finding *all* relevant items or the *total search efficiency*, defined as

$$\begin{aligned} \text{TSE}_e(\pi, \mathcal{R}) &= \sum_{i=1}^m e(p_i) z_{\text{SL}_3}(i, m) \\ &= e(p_m), \end{aligned} \tag{3}$$

where the specific exposure function depends on base top-heavy recall-level metric (e.g., AP and NDCG). TSE, then, is a family of metrics parameterized by a specific exposure function with properties defined in Section 2.3.1. Although computing  $\text{TSE}_e$  depends on the exposure model  $e$ , unless necessary, we will drop the subscript for clarity. We demonstrate the precision and recall orientation of TSE in Figure 5. Since TSE with an AP base behaves identically to RR, except from the perspective of recall orientation, we consider it RR’s recall-oriented counterpart. In the next section, we will connect this notion of recall to concepts of robustness and fairness.

Table 1. Ambiguity in Requests

Task	Request		Relevance	
	Implicit	Explicit	User 1	User 2
web retrieval	—	query='salsa'	recipe	dance
map retrieval	location='NYC'	query='restaurant'	vegetarian	Korean
movie recommendation	watched=['Jaws', 'Piranha']	—	nautical horror	1970s horror
music recommendation	saved=['Black Sabbath', 'Bach']	query='study music'	metal	classical

In both retrieval and recommendation tasks, multiple users or the same user at different times may consider different items relevance.

#### 4 Robustness

In the context of a single ranking, we are interested in measuring its robustness in terms of its effectiveness for different possible users who might have issued the same request.<sup>10</sup> In retrieval, this might occur when multiple users issue the same text-based query. In recommendation, this might occur when multiple users share the same engagement or session history, for example in cold-start or low-data situations. In both retrieval and recommendation, even a single user may have multiple intents depending on the context (e.g., an item may be relevant to a user when they are studying versus when they are relaxing). We present additional examples in Table 1 that demonstrate situations where data sparsity in the request can conceal two users with different intents. The concept of robustness across users or user-contexts is related to work in search engine auditing that empirically studies how effectiveness varies across different searchers issuing the same request [67]. In our work, we define robustness as the effectiveness of a ranking for the worst-off user (or user-context) who might have issued a request.

Underlying our notion of robustness is a population-based perspective on ranking evaluation. Classic effectiveness measures can be interpreted as expected values over different user populations defined by different browsing behavior [12, 83, 92]. For example, Robertson [83] demonstrated that AP can be interpreted as the expected precision over a population of users with different recall requirements. More generally, Carterette [12] demonstrated this for a large class of metrics. We can contrast this with online evaluation production environments where systems observe individual user behavior and do not need to resort to statistical models to capture different user behavior. So, just as recent work in the fairness literature disaggregates evaluation metrics to understand how performance varies across groups [35, 37, 67, 76], we can disaggregate traditional evaluation metrics to understand how performance varies across implicit subpopulations of users or providers.

From an ethical perspective, when considered the expected value over a population of users, traditional metrics make assumptions aligned with average utilitarianism, where the expected utility over some population is used to make decisions [99]. While this reduces, in production environments, to averaging a performance metric across all logged requests, in offline evaluation, this is captured by the distribution underlying the metric, as suggested by Robertson [83] and Carterette [12]. This means that if there are certain types of user behavior that are over-represented in the data (online evaluation) or the user model (offline evaluation), then they

<sup>10</sup>Robustness in the information retrieval community has traditionally emphasized slightly different notions from our ranking-based perspective. For example, the TREC Robust track emphasized robustness of *searcher effectiveness across information needs*, focusing evaluation on difficult queries [112]. Similarly, risk-based robust evaluation seeks to ensure that *performance improvements across information needs* are robust with respect to a baseline [18, 118]. Meanwhile, Goren et al. [43] proposed robustness as the *stability of rankings across adversarial document manipulations*. In the context of recommender systems, robustness has analogously focused on robustness of *utility across users* [120, 122] and *stability of rankings across adversarial content providers* [70].

will dominate the expectation. Users whose behaviors or needs have low probability in the data or the user model will be overwhelmed and effectively be obscured from measurement.

For robustness, instead of measuring the effectiveness of a system by adopting average utilitarianism and computing the expected performance over users, we can summarize the distribution of performance over users using alternative traditions based on distributive justice. This follows recent literature in value-sensitive, normative design of evaluation metrics [39, 115–117]. Specifically, inspired by related work in fair classification [48, 49, 68, 98], we can adopt Rawls’ difference principle that evaluates a decision based on its value to the worst-off individual [81]. In the context of a single ranking, this means the worst-off user or provider. As such, our worst-case analysis is aligned with Rawlsian versions of (i) equality of information access (for users) and (ii) fairness of the distribution of exposure (for providers). Even from a utilitarian perspective, systematic underperformance can cost ranking system providers as a result of user attrition [48, 69, 125] or negative impacts to a system’s brand [104].

Although motivated by similar societal goals (e.g., equity, justice), existing methods of measuring fairness in ranking are normatively very different from worst-case robustness. First, the majority of fair ranking measures emphasize equal exposure amongst providers [35] and is based on strict egalitarianism, a different ethical foundation than Rawls’ difference principle [80]. Pragmatically, to satisfy this within a single ranking, authors restrict analysis to stochastic ranking algorithms [30, 100] or amortized evaluation [4, 5]. Second, fair ranking analyses that focus on users tend to be restricted to disaggregated evaluation, without reaggregating [37, 67]. This is different from our focus on disaggregating and then summarizing the distribution of effectiveness with the worst-off user. Most importantly, while most fairness work looks at *either* users or providers, in our analysis, we demonstrate that both worst-case user *and* provider robustness are simultaneously captured by recall, as measured by TSE.

#### 4.1 User Robustness

Given a ranking  $\pi$ , we would like to measure the worst-case effectiveness over a population of possible users.

**4.1.1 Possible Information Needs.** In this section, we describe how relevance, as traditionally used, reflects the set of *possible user information needs*. To see why, consider the notion of relevance described in Section 2.1. This is often referred to as *topical relevance*, the match between an item and the general topic of the user [7, 21]. By contrast, Harter [45] uses the expression *psychological relevance* to refer to the extent to which, in the course of an information access session, an item changes the user’s cognitive state with respect to their information need. Otherwise relevant items may stop being useful as a user’s anomalous state of knowledge changes [3]. Moreover, as Harter [46] notes, because users approach a system from a variety of backgrounds (i.e., states of knowledge), the same request might find quite different utility from two topically relevant items in the corpus. Indeed, researchers have observed variation in utility in controlled experiments [113, 114] as well as production environments [31, 106].

In the context of information retrieval, the notion of topical relevance can be interpreted as the potential utility of a document to the user. A topically relevant document may be psychologically nonrelevant for a number of reasons. First, a user may already be familiar with a judged relevant item in the ranking, which, in some cases, will, for that user, make the item nonrelevant. For example, in the context of a literature review, a previously read relevant article may not be useful [6, 7]. Second, even if editorial relevance labels are accurate (i.e., all users would consider labeled items as relevant), the *utility* of items may be isolated to a subset of  $\mathcal{R}$ . For example, in the context of decision support, including legal discovery and systematic review, topical relevance is the first

step in finding critical information [20]. In some cases, there will be a single useful “smoking gun” document amongst the larger set of relevant content. In other cases, a single subset of relevant documents will allow one to “connect the dots.” In a patent context, Trippe and Ruthven [110] describe situations where there is risk to missing items that may turn out to be critical to assessing the validity of a patent. So, while topical relevance is important, it only reflects the *possible usefulness* to the user [107].

Similarly, in the context of recommendation, the notion of topical relevance can be interpreted as the potential utility—but not the psychological relevance or desirability—of an item to the user. So, while a particular item may be of interest to a user in general, at any one point in time, it may be undesired for any number of reasons. First, the context of a user can affect whether a particular item is desired. For example, in music recommendation, a user may be interested in bluegrass music in many contexts but, while studying, may prefer ambient music. Second, as with information retrieval, the desirability of a previously consumed relevant item may degrade due to order effects. For example, in music recommendation, consider a user who listened to a specific relevant song  $X$ . There are two possible effects. *Satiation* [58] means that the user may not want to listen to  $X$  immediately again, making it no longer relevant to that user. *Obsessive listening* [19, 119] means that the user may want to listen to  $X$  over again, making other otherwise relevant songs no longer relevant to that user.

So, from the perspective of relevance,  $\mathcal{R}$  should be considered the union of psychologically relevant items over all possible states of knowledge a user might have when approaching the system. Indeed, multiple authors describe topical relevance as necessary but not sufficient for psychological relevance [7, 21, 87]. These papers suggest that rather than seeing a ranking system as acting to directly provide psychologically relevant items, it provides topically relevant items that are candidates to be scanned for psychologically relevant items by the user [7]. In light of this discussion, instead of considering any item in  $\mathcal{R}$  as definitely satisfying the user’s information need, we consider it as only having a nonzero probability of satisfying the user’s information need.

While the concept of diversity [15, 94], common to both recommendation and retrieval tasks, intends to support multiple information needs, they do not capture granular, item- or document-level needs. For example, measuring book recommendation diversity along the language dimension may detect poor performance for a group of users interested in Spanish language books, it will not be effective at detecting poor performance for Spanish language books unfamiliar to a particular user.

Given that binary relevance reflects the *possibility of psychological relevance*, we are interested in considering all users such that the union of their relevance criteria is  $\mathcal{R}$ . We can enumerate all such users over items as  $\mathcal{U} = \mathcal{R}^+$ , the power set of  $\mathcal{R}$  excluding the empty set. This means that, for a given request, we have  $m^+ = |\mathcal{U}| = 2^m - 1$  possible users interested in at least one relevant item (Figure 6). This conservative definition of  $\mathcal{U}$  captures all possible satisfiable users.

**4.1.2 Robustness across Possible Information Needs.** Given a set of possible information needs  $\mathcal{U}$  based on  $\mathcal{R}$ , we define the robustness of a ranking  $\pi$  as the effectiveness of the ranking for the worst-off user,

$$WC_\mu(\pi, \mathcal{R}) = \min_{u \in \mathcal{U}} \mu(\pi, u). \quad (4)$$

This is close to the notion of robustness proposed by Memarrast et al. [68], who consider a worst-case user for whom relevant items have a marginal distribution of features that matches the distribution in the full training set. In comparison, our analysis considers the *full* set of worst-case situations, including those that do not match the training data.

One problem with this definition of robustness is that, because  $m^+$  is exponential in  $m$ , computing the minimum is impractical even for modest  $m$ . Fortunately, using the properties of top-heavy



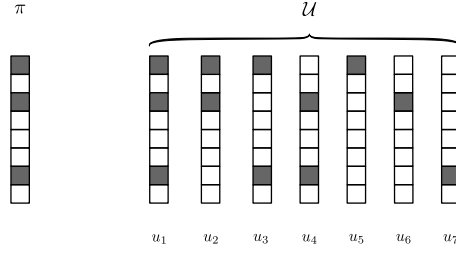


Fig. 6. Population of possible users  $\mathcal{U}$  based on all combinations of relevant items from  $\mathcal{R}$  for a system ranking  $\pi$ .

recall-level metrics, we can prove that

$$\text{WC}_\mu(\pi, \mathcal{R}) = \text{TSE}(\pi, \mathcal{R}). \quad (5)$$

In other words, the worst-off user is the one associated with  $p_m$ , the lowest-ranked relevant item. We present a proof in Appendix C.1. This result implies that recall orientation captures the utility of a ranking for the worst-off user.

## 4.2 Provider Robustness

Providers are individuals who contribute content to the ranking system's catalog  $\mathcal{D}$ . Given a ranking  $\pi$ , we would like to measure the worst-case effectiveness over a population of possible providers.

**4.2.1 Possible Provider Preferences.** Just as with information needs, each ranking consists of exposure of multiple possible providers. Consider the domains like job applicant ranking systems or dating platforms, where each item in the catalog is associated with an individual person. We assume that each relevant provider  $i \in \mathcal{R}$  is interested in its cumulative positive exposure in the ranking (Section 2.3.2),  $\eta(\pi, \mathcal{R}, \{i\})$ . In the more general case, providers can possibly be associated with multiple relevant items in  $\mathcal{R}$ . This might occur if a creator contributes multiple items to the catalog (e.g., multiple songs, videos, documents), or a provider may aggregate content from multiple individual creators (e.g., publishers, labels). Even if we have metadata attributing groups of items to specific providers or creators, their *preferences* for exposure of those items may be unobserved. Provider preferences can themselves be complex, covering a broad set of commercial, artistic, and societal values [33, 44].

Given the uncertainty and ambiguity over providers and their preferences, as with information needs, we can consider the full set of latent providers and their preferences,  $\mathcal{V} = \mathcal{R}^+$  to reflect the set of *possible* provider preferences.

**4.2.2 Robustness across Possible Provider Preferences.** Just as with users, we are interested in the utility of the worst-off provider. In the simple case where each provider is associated with a single item in  $\mathcal{R}$ , because exposure monotonically decreases with rank position, we know that the worst-off provider will be the one at the lowest rank; this is exactly  $\text{TSE}(\pi, \mathcal{R})$ . This is similar to earlier provider fairness definition [128]. More generally, if, like information needs, we consider  $\mathcal{V} = \mathcal{R}^+$ , then we define the worst-off provider as

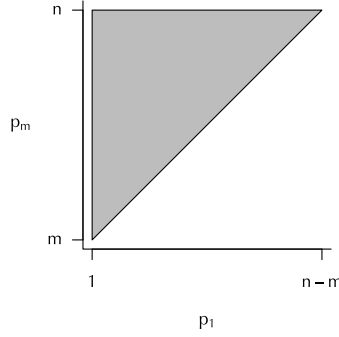
$$\text{WC}_\eta(\pi, \mathcal{R}) = \min_{v \in \mathcal{V}} \eta(\pi, \mathcal{R}, v). \quad (6)$$



Table 2. Agreement with  $WC_\mu(\pi, \mathcal{R}) < WC_\mu(\pi', \mathcal{R})$ 

$n$	tied	TSE	R@1000	RP	AP	NDCG	random
$10^3$	0.012	1.000	0.000	0.285	0.541	0.535	0.492
$10^4$	0.001	1.000	0.420	0.077	0.552	0.549	0.497
$10^5$	0.000	1.000	0.179	0.008	0.554	0.555	0.498
$10^6$	0.000	1.000	0.026	0.001	0.547	0.554	0.499

Probability of agreement over 10,000 simulated queries and pairs of random rankings for various corpus sizes. For each query, we selected  $m \sim U(5, 50)$  relevant items. We include the fraction of rankings tied under  $WC_\mu(\pi, \mathcal{R}) = WC_\mu(\pi', \mathcal{R})$ . The values for TSE apply to any exposure model.

Fig. 7. Dependence between  $p_1$  and  $p_m$ . The gray region indicates the possible values of each.

Given this definition, we can show that TSE is equal to the utility of the worst-case provider (proof in Appendix C.1).

Together, the theoretical results in Sections 4.1 and 4.2 provide a new interpretation of recall from the perspective of potential subpopulations of content consumers and providers participating on the platform. Worst-case analysis specifically connects naturally to Zobel et al.'s notion of totality [130].

### 4.3 Robustness and Existing Metrics

We have demonstrated that whenever  $WC_\mu(\pi, \mathcal{R}) < WC_\mu(\pi', \mathcal{R})$ , then  $TSE(\pi, \mathcal{R}) < TSE(\pi', \mathcal{R})$ . To understand the relationship between other metrics and worst-case performance, we simulate 10,000 requests by sampling 10,000 pairs of rankings  $\pi$  and  $\pi'$ . We can then compute the evaluation metric for each ranking and compare  $\mu(\pi, \mathcal{R}) < \mu(\pi', \mathcal{R})$  with  $WC_\mu(\pi, \mathcal{R}) < WC_\mu(\pi', \mathcal{R})$ . We conduct this simulation for  $n \in \{10^3, 10^4, 10^5, 10^6\}$  and present results in Table 2. We observe that, while TSE has perfect sign agreement with  $WC_\mu$ , other recall-oriented metrics have worse agreement than random, largely because they *only* look at a prefix of  $p$ . The sign agreement of AP and NDCG is slightly better than random for two reasons. First, their aggregation (Equation (1)) includes  $p_m$  and will subtly affect the value, despite making a small contribution to the total sum. Second,  $p_i$  values depend on each other because  $\pi$  is a permutation. In Figure 7, we compare the possible joint values of  $p_1$  and  $p_m$ . This means that we should expect there to be some dependence between purely precision-oriented metrics (e.g., RR) and purely recall-oriented metrics (e.g., TSE). That said, AP and NDCG agree less than TSE because their aggregation includes the positions of relevant items above  $p_m$ . In whole, this result suggests that, even compared to traditional recall metrics, TSE is better able to measure the robustness of a ranking.

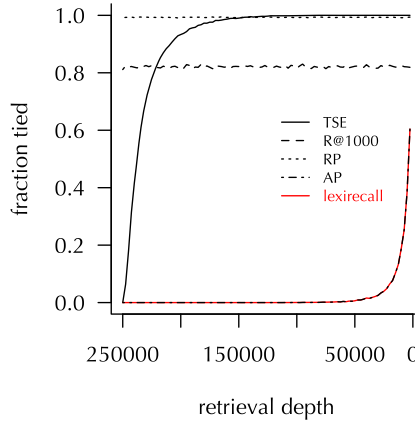


Fig. 8. Fraction tied rankings as retrieval depth  $\tilde{n}$  decreases for  $n = 250,000$  and 25 requests. For each query, we selected  $m \sim U(5, 50)$  relevant items. This figure best rendered in color.

## 5 Lexicographic Evaluation

In Section 3, we defined recall orientation from the perspective of users interested in finding the totality of relevant items and then proposed a new metric, TSE, based on this interpretation. In Section 4, we demonstrated how TSE measures the worst-case utility for multiple definitions of users and providers, connecting it to notions of robustness and fairness, through Rawls' difference principle. In this section, we will further develop the fairness perspective by combining recent work in preference-based evaluation with classic work in social choice theory, improving the nuance in worst-case analysis and allowing it to be useful as an evaluation tool. We will begin by discussing the practical limitations of TSE for evaluation (Section 5.1) before developing a preference-based evaluation method derived from social choice theory that generalizes TSE and improves its practical use (5.2).

### 5.1 Low Sensitivity of Total Search Efficiency

Although measuring worst-case performance and, as a result, Rawlsian fairness, TSE may not satisfy our desiderata for an evaluation method (Section 2.4). To understand why, consider two rankings  $\pi$  and  $\pi'$  for the same request. When comparing a pair of systems, we are interested in defining a preference relation  $\pi > \pi'$ . The worst-case preference  $\pi >_{\text{WC}} \pi'$ , is defined as

$$\pi >_{\text{WC}} \pi' \leftrightarrow \min_{u \in \mathcal{U}} \mu(\pi, u) > \min_{u \in \mathcal{U}} \mu(\pi', u). \quad (7)$$

We know from Section 4.1.2 that this can be efficiently computed as  $\text{TSE}(\pi) > \text{TSE}(\pi')$ . Unfortunately, in situations where the worst-off user is tied (i.e.,  $\min_{u \in \mathcal{U}} \mu(\pi, u) = \min_{u \in \mathcal{U}} \mu(\pi', u)$ ), we cannot derive a preference between  $\pi$  and  $\pi'$ . Because we assume that unretrieved items occur at the bottom of the ranking and because most runs do not return all of the relevant items, an evaluation based on TSE will observe many ties between  $\pi$  and  $\pi'$ , limiting its effectiveness at distinguishing runs and use for system development [11]. In Figure 8, we simulated random pairs of rankings of 250,000 items and computed the number of metric ties for a variety of retrieval cutoffs. We observe that  $R_{1000}$  and RP both have a large number of ties across all retrieval depths. Despite having few ties for very deep retrievals, TSE quickly observes many ties. This is due to our conservative permutation imputation method (Section 2.2). We can compare all of these measures to AP,

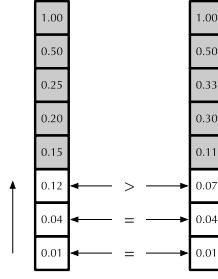


Fig. 9. Lexicographic relation between two sorted vectors with the same dimension. In leximin ordering, we compare pairs of elements from the bottom up and define the preference between vectors based on the first difference in values. In this example, the last element in both vectors is 0.01 and the second to last element is 0.04 in both. The third to last elements differ and we say that we prefer the ranking on the left because  $0.12 > 0.07$ .

which exhibits high sensitivity across most cutoffs. In this section, we will improve the sensitivity of TSE to be comparable to AP using methods from social choice theory.

## 5.2 Lexicographic Recall

We can address the lack of sensitivity of TSE by turning to recent work on *preference-based evaluation* [17, 28, 29]. As mentioned earlier, in many evaluation scenarios, our objective is to compute  $\pi > \pi'$ . In *metric-based evaluation*, we compute this preference by first computing the value of an evaluation metric for each ranking. That is,

$$\mu(\pi) > \mu(\pi') \implies \pi > \pi'. \quad (8)$$

Preference-based evaluation [28, 29] is a quantitative evaluation method that directly computes the preference between two rankings  $\pi$  and  $\pi'$  without first computing an evaluation metric. Diaz and Ferraro [29] and Diaz [28] show that preference-based evaluation can achieve much higher statistical sensitivity compared to standard metric-based evaluation.

We can convert TSE into a much more sensitive preference-based evaluation by returning to our discussion of fairness and robustness. In the context of social choice theory, the number of ties in Rawlsian fairness can be addressed by adopting a recursive procedure known as *leximin*, originally proposed by Sen [95]. Consider the problem of distributing a resource to  $m$  individuals, in our case users or providers. Further consider two different allocations  $x$  and  $y$  represented by two  $m \times 1$  vectors where  $x_i$  is the amount of resource allocated to the  $i$ th-highest ranked individual, similarly for  $y$ . In other words,  $x$  and  $y$  are the resource allocations in decreasing order. Given these two allocations, we begin by inspecting the allocation to the lowest-ranked items, as we did with Rawlsian fairness. If  $x_m > y_m$ , then the bottom-ranked item of  $x$  is better off and  $x > y$ ; if  $x_m < y_m$ , then the bottom-ranked item of  $y$  is better off and  $x < y$ ; otherwise, the bottom-ranked items are equally well-off and we inspect the position of the next lowest item,  $m - 1$ . If  $x_{m-1} > y_{m-1}$ , then we say  $x > y$ ; if  $x_{m-1} < y_{m-1}$ , then we say  $x < y$ ; otherwise, we inspect the position of the next lowest item  $m - 2$ . We continue this procedure until we return a preference or, if we exhaust all  $m$  positions, we say that we are indifferent between the two rankings. Formally,

$$x >_{\text{leximin}} y \leftrightarrow x_i > y_i, \quad (9)$$

where  $i = \max\{j \in [1 \dots m] : x_j \neq y_j\}$ . We show an example of this process in Figure 9. This way of comparing rankings generates a total lexicographic ordering over vectors of the same

dimensionality and is often used in the fairness literature to address ties when adopting Rawls' difference principle.<sup>11</sup>

We can use leximin to define the *lexicographic recall* or *lexirecall* preference between  $\pi$  and  $\pi'$ . For a fixed request and ranking  $\pi$ , let  $\varrho$  be the  $m^+ \times 1$  vector of metric values for  $\mathcal{U}$  sorted in decreasing order. In other words,  $\varrho_i = \mu(\pi, u^i)$ , where  $u^i$  is the user with the  $i$ th-highest metric value. We define  $\varrho'$  equivalently for  $\pi'$ . Lexicographic recall is defined as

$$\pi \succ_{\text{LR}} \pi' \leftrightarrow \varrho \succ_{\text{leximin}} \varrho'. \quad (10)$$

Using lexirecall, we can define an ordering over unique rankings, which addresses the ties observed in TSE.

Although operating over  $\mathcal{U}$  grounds our evaluation in possible user information needs, scoring and ranking  $m^+$  subsets of  $\mathcal{R}$  can be computationally intractable. So, just as we demonstrated that we only need to inspect the position of the last relevant item to compute  $\succ_{\text{WC}}$ , we can demonstrate that we only need to compare the rank positions of the relevant items to compute  $\pi \succ_{\text{LR}} \pi'$  (proof in Appendix C.2) and, therefore,

$$\pi \succ_{\text{LR}} \pi' \leftrightarrow p_i < p'_i, \quad (11)$$

where  $i = \max\{j \in [1 \dots m] : p_j \neq p'_j\}$ . Moreover, although defined as a user-oriented metric, we can also demonstrate that this results in provider leximin as well (proof in Appendix C.2).

We can better understand lexirecall by returning to our discussion of robustness in Section 4. While TSE provided one way to distinguish robustness of two rankings, it is very insensitive and unlikely to be of practical use. To address this, we adopted leximin, a well-studied method for addressing insensitivity in applying Rawls' difference principle. That said, lexirecall is still a measure of robustness. A lexirecall preference is simply claiming that one ranking is more fair or more robust than another. Over a population of requests, then, we can compute the probability that one system's rankings are fairer or more robust than another.

### 5.3 Sensitivity of Lexicographic Recall

We can demonstrate the higher sensitivity of lexirecall through simulation and analysis of the total space of permutations. In Figure 8, we demonstrated that, for a set of random paired rankings, the number of ties was high for traditional metrics and grew quickly for TSE as the cutoff  $\tilde{n}$  decreased. Figure 8 also includes lexirecall, which exhibits substantially fewer ties than traditional metrics and TSE. Independent of simulation, we are also interested in the probability of a tie over for randomly sampled pairs of complete rankings  $\pi, \pi' \in S_n$  (i.e.,  $\tilde{n} = n$ ). We can derive these probabilities (see Appendix E) as functions of  $m, n$ , and any parameters of the metric (e.g.,  $k$ ),

$$\begin{aligned} \Pr(\pi =_{\text{TSE}} \pi') &= \binom{n}{m}^{-2} \sum_{i=m}^n \binom{i-1}{m-1}^2 \\ \Pr(\pi =_{\text{R}_k} \pi') &= \binom{n}{m}^{-2} \sum_{i=0}^m \binom{k}{i}^2 \binom{n-k}{m-i}^2 \\ \Pr(\pi =_{\text{RP}} \pi') &= \binom{n}{m}^{-2} \sum_{i=0}^m \binom{m}{i}^2 \binom{n-m}{m-i}^2 \\ \Pr(\pi =_{\text{LR}} \pi') &= \frac{m!(n-m)!}{n!}. \end{aligned}$$

<sup>11</sup>For further discussion of the connection between Rawls' difference principle and leximin, see References [26, 57, 75, 96, 97].

Table 3. Probability of a Metric Tie  $\Pr(\pi =_{\mu} \pi')$  for Randomly Sampled Permutations and  $m = 10$ 

$n$	TSE	$R_{1000}$	RP	AP	LR
$10^3$	0.005	1.000	0.825	0.000	0.000
$10^4$	0.001	0.313	0.980	0.000	0.000
$10^5$	0.000	0.826	0.998	0.000	0.000
$10^6$	0.000	0.980	1.000	0.000	0.000

Table 4. Probability of a Metric Tie  $\Pr(\pi =_{\mu} \pi')$  for Randomly Sampled Permutations and  $n = 10^6$ 

$m$	TSE	$R_{1000}$	RP	AP	LR
1	0.000	0.998	1.000	0.000	0.000
5	0.000	0.990	1.000	0.000	0.000
10	0.000	0.981	1.000	0.000	0.000
25	0.000	0.952	0.999	0.000	0.000
50	0.000	0.907	0.995	0.000	0.000

To better understand the relationship between these probabilities, we display probabilities of ties for several retrieval depths in Table 3. Although Figure 8 demonstrated that TSE exhibited poor sensitivity when  $\tilde{n} < n$ , it is much more sensitive for complete rankings, in part because random complete rankings are less likely to share  $p_m$  than imputed rankings. Both traditional recall metrics exhibit many more ties, especially as the retrieval depth grows. Among methods, lexirecall and AP demonstrate the few or no ties across corpus sizes. Table 4 presents the same results for varying numbers of relevant items. The results are consistent with Table 3, where traditional recall metrics exhibit a large number of ties, which decreases slowly with the number of relevant items. By comparison, TSE, lexirecall, and AP show higher sensitivity with a negligible number of ties.

## 6 Empirical Analysis

In this section, we empirically assess lexirecall with respect to the associated empirical desiderata from Section 2.4: (i) correlation with existing metrics, (ii) ability to distinguish between rankings, (iii) ability to distinguish between systems, and (iv) robustness to missing labels.

### 6.1 Methods and Materials

**6.1.1 Data.** We evaluated ranking system runs across a variety of conditions (Table 5). For each dataset, we have a set of evaluation requests and associated relevance judgments. In addition, each dataset involved a number of competing systems, each of which produced a ranking for every request. The movielens, libraryThing, and beerAdvocate datasets were downloaded from a public repository with splits and processing described in prior work [111].<sup>12</sup> Amazon datasets were prepared using LensKit 0.14.4 [34] to train a variety of recommendation models on five-core global split datasets from the 2023 Amazon Reviews data [51] and generate top-K recommendation runs for the users in the test splits, using an early version of the LensKit Codex.<sup>13</sup> For recommender systems datasets, consistent with Reference [111], we converted graded judgments to binary labels by considering any rating below 4 as nonrelevant and otherwise relevant for these datasets. All

<sup>12</sup><https://github.com/dvalcarce/evalMetrics>

<sup>13</sup><https://codex.lenskit.org>; runs will be provided as supplemental resources for this article when published.

Table 5. Datasets Used in Empirical Analysis

	Requests	Runs	Rel/request	Docs/request	Label
<b>recommendation</b>					
amzn-cds-vinyl-100	6,875	10	2.76	100	shallow
amzn-cds-vinyl-1000	6,875	10	2.76	1,000	standard
amzn-cds-vinyl-2000	6,875	10	2.76	2,000	deep
amzn-musical-instruments-100	11,648	10	3.08	100	shallow
amzn-musical-instruments-1000	11,648	10	3.08	1,000	standard
amzn-musical-instruments-2000	11,648	10	3.08	2,000	deep
amzn-software-100	6,345	10	2.28	100	shallow
amzn-software-1000	6,345	10	2.28	1,000	standard
amzn-software-2000	6,345	10	2.28	2,000	deep
amzn-video-games-100	11,554	10	3.08	100	shallow
amzn-video-games-1000	11,554	10	3.08	1,000	standard
amzn-video-games-2000	11,554	10	3.08	2,000	deep
beerAdvocate	17,564	21	13.66	99.39	shallow
libraryThing	7,227	21	13.15	100.00	shallow
movielens	6,005	21	18.87	100.00	shallow
<b>retrieval</b>					
core (2017)	50	75	180.04	8853.11	deep
core (2018)	50	72	78.96	7102.61	deep
deep-docs (2019)	43	38	153.42	623.77	standard
deep-docs (2020)	45	64	39.27	99.55	shallow
deep-docs (2021)	57	66	189.63	98.83	shallow
deep-docs (2022)	76	42	1245.62	98.86	shallow
deep-docs (2023)	82	5	75.10	100.00	shallow
deep-pass (2019)	43	37	95.40	892.51	standard
deep-pass (2020)	54	59	66.78	978.01	standard
deep-pass (2021)	53	63	191.96	99.95	shallow
deep-pass (2022)	76	100	628.145	97.50	shallow
deep-pass (2023)	82	35	49.87	99.90	shallow
legal (2006)	39	34	110.85	4835.07	deep
legal (2007)	43	68	101.023	22240.30	deep
robust (2004)	249	110	69.93	913.82	standard
web (2009)	50	48	129.98	925.31	standard
web (2010)	48	32	187.63	7013.21	deep
web (2011)	50	61	167.56	8325.07	deep
web (2012)	50	48	187.36	6719.53	deep
web (2013)	50	61	182.42	7174.38	deep
web (2014)	50	30	212.58	6313.98	deep

Runs submitted to the associated TREC track or recommendation task. Datasets are labeled according to the depth of runs: “deep” ( $\tilde{n} > 1000$ ), “standard” ( $\tilde{n} \in (100 \dots 1000]$ ), and “shallow” ( $\tilde{n} \leq 100$ ).

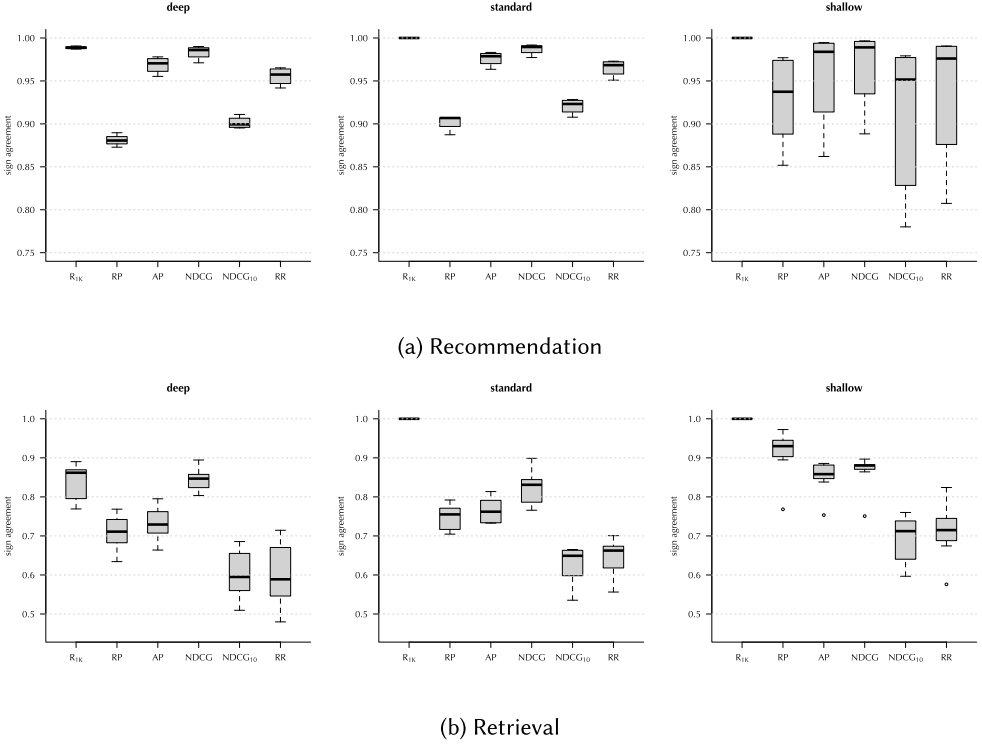


Fig. 10. Agreement between lexirecall and traditional metrics over rankings  $\pi, \pi'$  in our datasets. Fraction of ranking pairs where the lexirecall preference agrees with the sign of the metric difference.

retrieval datasets were downloaded from NIST. To analyze results for different ranking depths, we categorized datasets as deep ( $\tilde{n} > 1000$ ), standard ( $\tilde{n} \in (100 \dots 1000]$ ), or shallow ( $\tilde{n} \leq 100$ ).

**6.1.2 Evaluation Methods.** We computed lexirecall using pessimistic imputation. We compare LR with two traditional recall metrics ( $R_{1000}$  and RP) and two metrics that combine recall and precision (AP and NDCG). Definitions for metrics can be found in Section 2. An implementation can be found at [https://github.com/diazf/pref\\_eval](https://github.com/diazf/pref_eval).

## 6.2 Results

**6.2.1 Agreement with Existing Metrics.** To understand the similarity of lexirecall and traditional metrics, we measured its preference agreement with traditional metrics. Specifically, given an observed metric difference,  $\mu(\pi) \neq \mu(\pi')$ , for a traditional metric in our datasets, we computed how often lexirecall agreed with the ordering of  $\pi$  and  $\pi'$ ,

$$\frac{\sum_{\pi, \pi' \in \tilde{S}_n} \mathbb{I}(\pi \succ_{\mu} \pi' \wedge \pi \succ_{\text{LR}} \pi')}{\sum_{\pi, \pi' \in \tilde{S}_n} \mathbb{I}(\pi \succ_{\mu} \pi')},$$

where  $\tilde{S}_n$  is the set of rankings in our dataset. We present results in Figure 10. For reference, we include high precision metrics RR and NDCG<sub>10</sub>, which expectedly have the weakest agreement with lexirecall across all retrieval depths. Similarly, across all depths, we observed highest sign agreement with  $R_{1000}$ , indicating an alignment between lexirecall and traditional notions of recall.



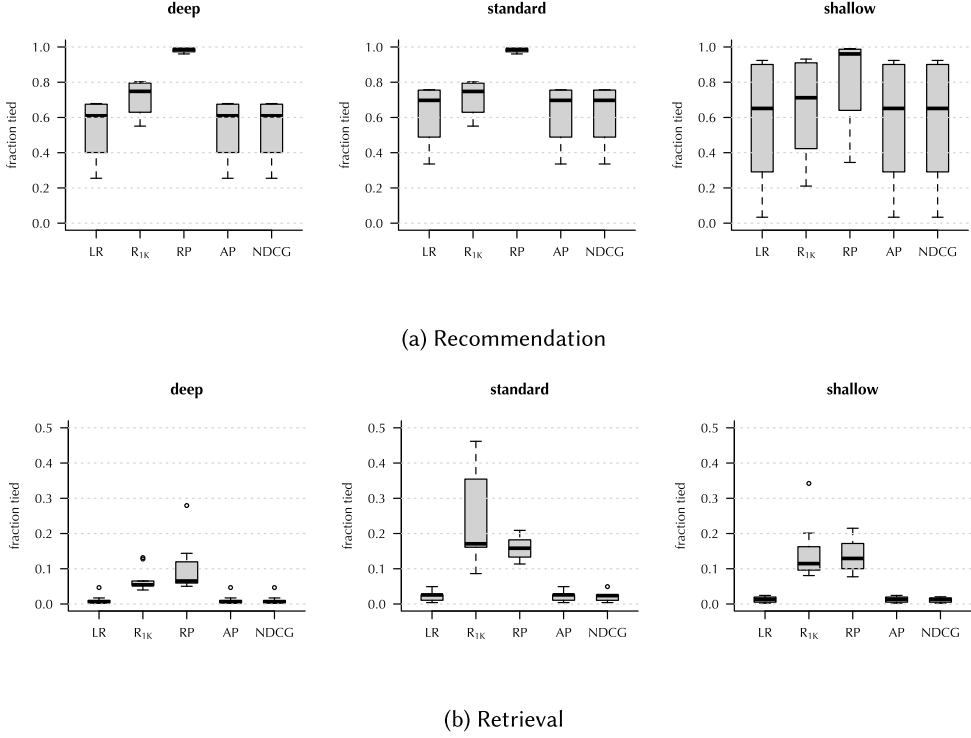


Fig. 11. Fraction of tied comparisons of rankings  $\pi, \pi'$  in recommendation and retrieval datasets.

Note that in the standard and shallow conditions, where  $\tilde{n} \leq 1000$ , if there is a difference in  $R_{1000}$ , then there is a difference in lexirecall due to pessimistic imputation; the converse is not true since lexirecall can distinguish rankings that are tied under  $R_{1000}$ . The agreement with NDCG increases to match that of  $R_{1000}$  with increased depth, perhaps due to the weaker position discounting in NDCG and higher likelihood of including a value based on the lowest ranked relevant item as depth increases (see Section 4.3). Both RP and AP show comparable agreement higher relative to RR and  $NDCG_{10}$ .

**6.2.2 Detection of Difference between Rankings.** In Section 5.3, we observed that, for pairs of rankings  $\pi, \pi' \in S_n$  sampled *uniformly at random*, lexirecall resulted in fewer ties than RP and  $R_{1000}$ . Figure 11 presents the empirical fraction of ties when sampling from rankings in our dataset (i.e.,  $\tilde{S}_n$ ). First, consider traditional recall metrics  $R_{1000}$  and RP. The empirical fraction of ties is substantially lower than suggested by Figure 8 (different  $\tilde{n}$ ) and Table 3 (different  $n$ ). This can be explained by the concentration of relevant items in the top positions  $\tilde{S}_n$  compared to  $S_n$ , resulting in fewer ties. That said, the fraction of ties for both of these metrics is substantially higher than observed for lexirecall, something consistent with results in Section 4.3. Although AP and NDCG capture both precision and recall valance, we can see that lexirecall is comparable in fraction of ties.

Although the general trends are consistent across recommendation and retrieval contexts, recommendation metrics have substantially higher variance. This results from the sparser set of labels in recommendation, where the total number of unique vectors,  $\binom{n}{m}$ , is lower and therefore there is a higher probability of ties.

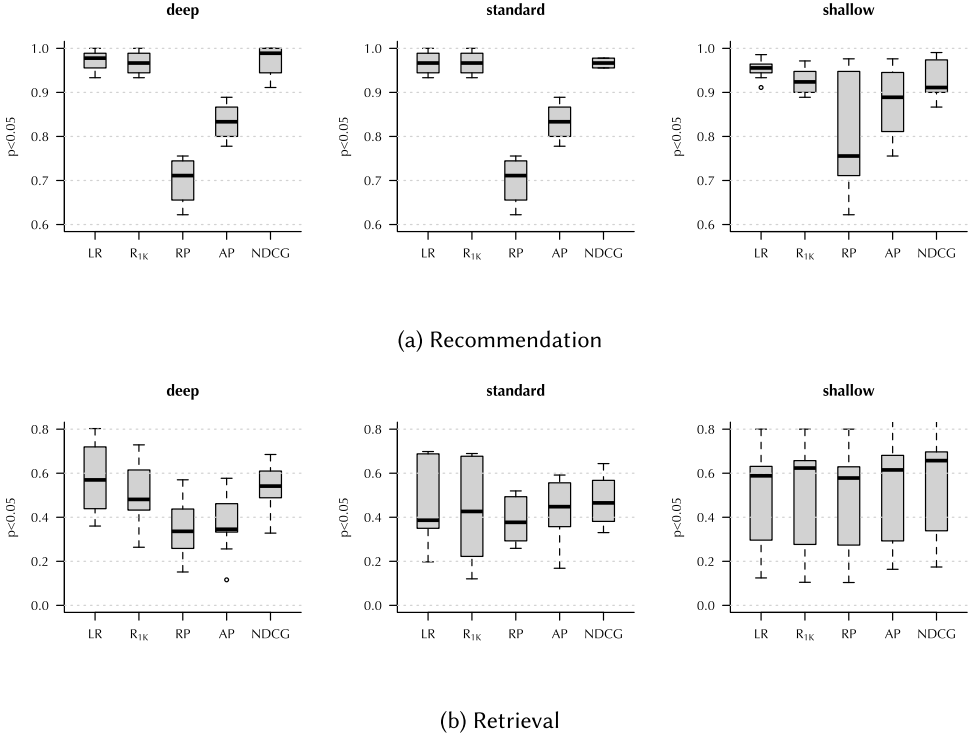


Fig. 12. Statistical sensitivity. Fraction of run pairs where we observe a statistically significant difference (i.e.,  $p < 0.05$ ) using a paired test.

**6.2.3 Statistical Sensitivity.** We are also interested in the ability of lexirecall to detect statistical differences between pairs of *runs* (i.e., sets of rankings generated by a single system for a shared set of queries). To do so, we adopted Sakai’s method of measuring the discriminative power of a metric [90]. This approach measures the fraction of pairs of systems that the method detects statistical differences with  $p < 0.05$ . We use two methods to compute  $p$  values. In the first, we compute a standard statistical test and, correcting for multiple comparisons, measure the fraction of  $p$  values below 0.05. For lexirecall, we adopt a binomial test since we have binary outcomes. For other metrics, we adopt a Student’s  $t$ -test, as recommended in the literature [101]. We also conducted experiments using incorrect-but-consistent statistical tests with similar outcomes. To correct for multiple comparisons for all tests, we use the conservative Holm–Bonferroni method [8]. Our second method of computing  $p$  values uses Tukey’s **honestly significant difference (HSD)** test as proposed by Carterette [14]. This method is considered a more appropriate approach to addressing multiple comparisons compared to our first approach. The goal of this analysis is to understand the statistical sensitivity of lexirecall compared to other recall-oriented metrics, while presenting nonrecall metrics for reference.

We present the results of this analysis in Figures 12 and 13. When using standard tests (Figure 12), lexirecall is slightly better at detecting significant differences compared to existing recall metrics at deeper retrievals. We can refine this analysis by inspecting the HSD results (Figure 13). In this case, the sensitivity of lexirecall manifests more strongly, clearly more discriminative than existing recall metrics for deep retrievals, although losing this power as retrieval depth decreases. This is consistent with previous observations for preference-based evaluation [28, 29].

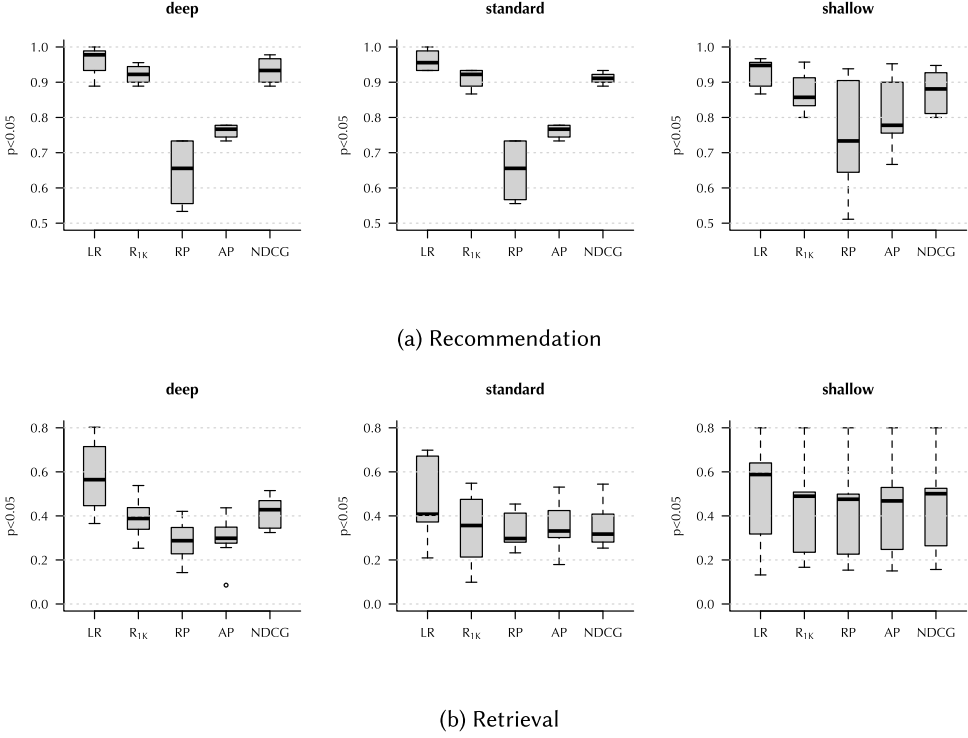


Fig. 13. Statistical sensitivity. Fraction of run pairs where we observe a statistically significant difference (i.e.,  $p < 0.05$ ) using Tukey's honestly significant difference test.

**6.2.4 Label Degradation.** Effective evaluation methods are stable in the presence of missing relevance labels. In this analysis, we held the number of queries fixed and randomly removed a fraction of relevant items per query, leaving at least one relevant item per query. We uniformly sample from all relevant items  $\mathcal{R}$  to remove. We present results for the effect of label degradation on the fraction of ties (we expect more ties with fewer labels) and preference agreement with full data (we expect lower agreement with fewer labels). As with the previous section, the goal of this analysis is to compare lexirecall to other recall-oriented metrics, while presenting nonrecall metrics for reference.

In terms of fraction of ties (Figure 14), lexirecall degrades comparably to metrics like AP and NDCG and substantially more gracefully compared to existing recall metrics  $R_{1000}$  and RP. While the importance of relevance labels for recall-oriented evaluation is important, this result suggests that existing metrics are extremely brittle when labels are missing. All methods observed more ties at shallower retrieval depths with degradation more pronounced for traditional recall-oriented metrics. As with our analysis of the number of ties (Section 6.2.2), because of the sparsity of labels in recommendation, we observe much higher variance compared to retrieval, where labels are more complete due to pooling [103, 129]. As such, the recommendation tasks can be seen as operating in the far right-hand portion of retrieval experiments.

In terms of agreement with preferences based on full data (Figure 15), lexirecall again degrades comparably to AP and NDCG. In recommendation experiments, RP and  $R_{1000}$  behave comparably or better than lexirecall, AP, and NDCG. However, this may be an artifact of the poor

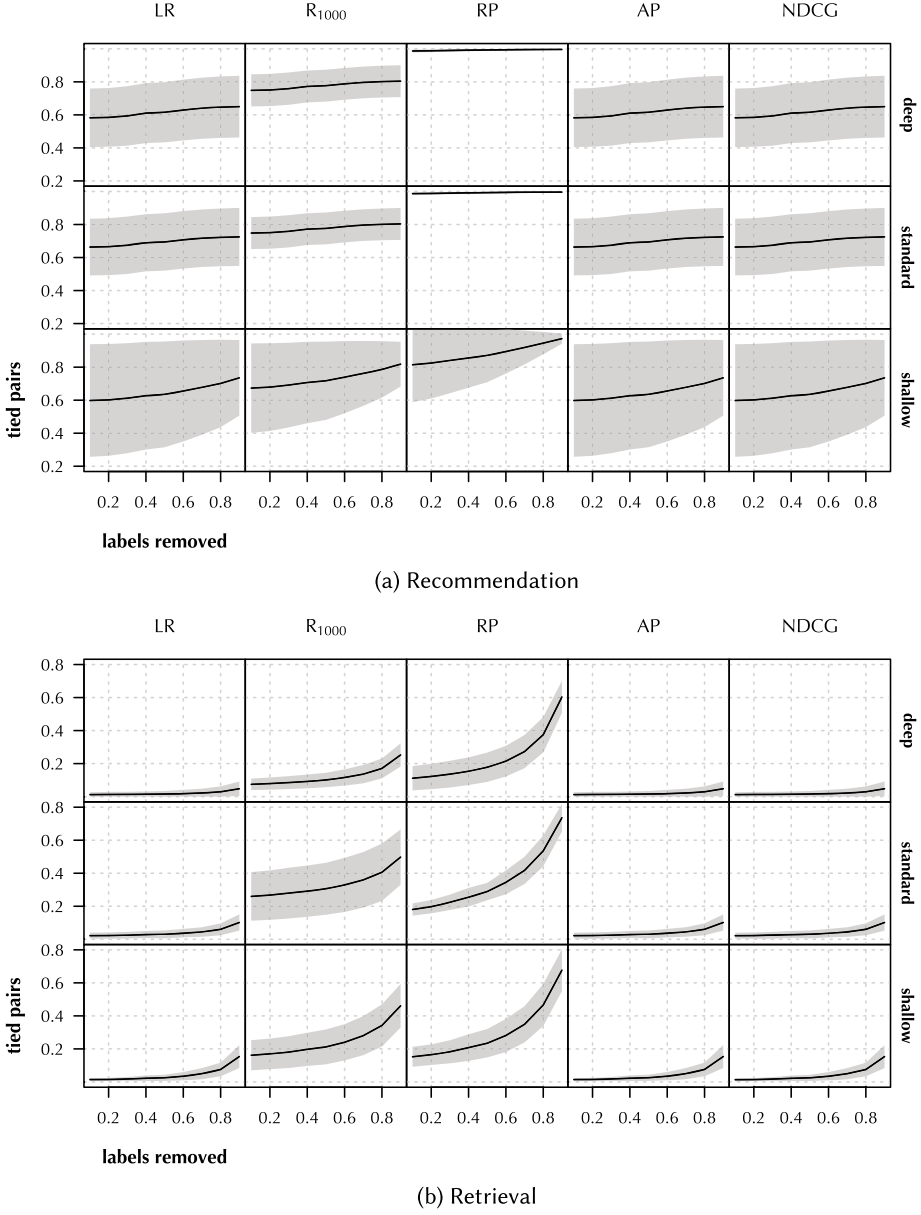
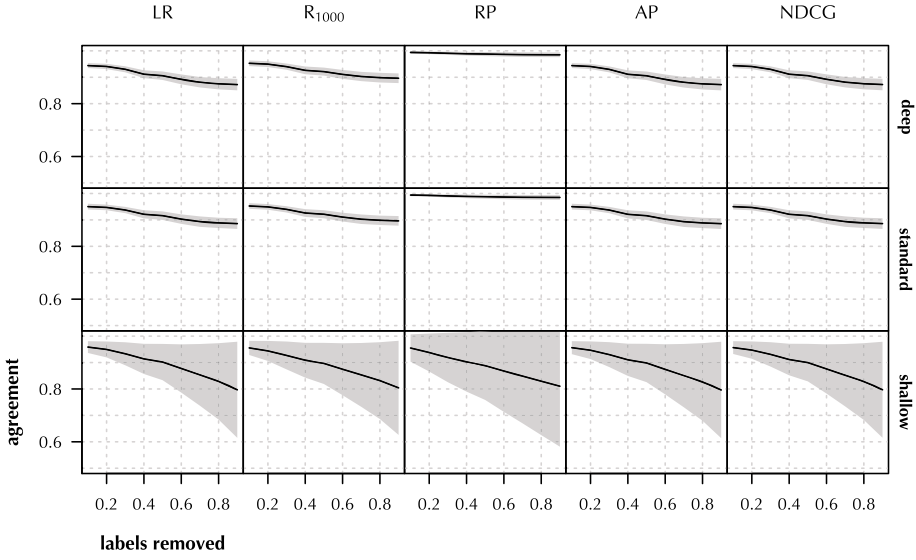
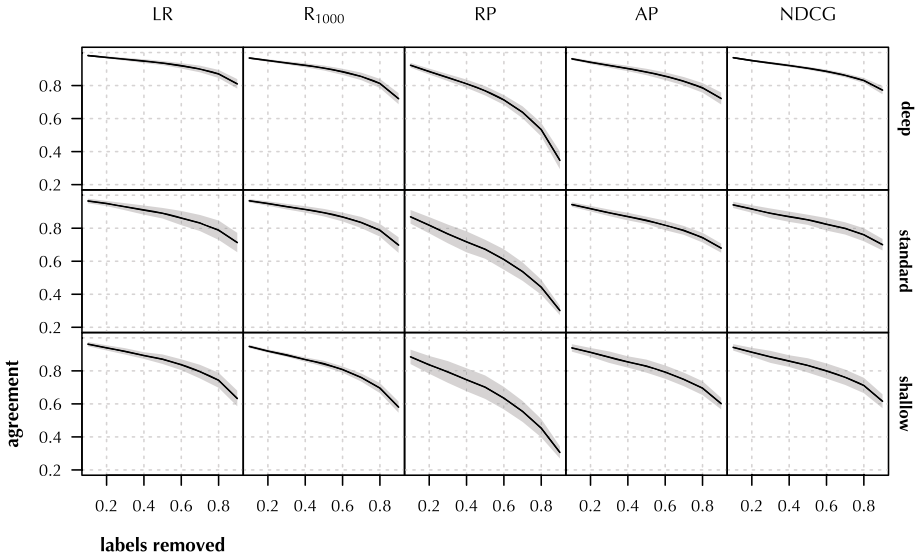


Fig. 14. Number of ties as labels removed. For each query, we removed a fraction of items from  $\mathcal{R}$  and counted the number of tied rankings. Relevant items were sampled uniformly at random. Average over 10 samples.

behavior of RP and  $R_{1000}$  under extremely sparse labels, as demonstrated in earlier experiments. That is, since these metrics observe so many ties and ties increase with label degradation, accurately predicting these preferences under “full data” will be easier with label degradation. We contrast this with retrieval experiments, where labels are more complete. In this condition, RP is much more sensitive to degradation, dropping in performance quickly. However,  $R_{1000}$  behaves similar to lexirecall when preserving most labels, but, for drastically sparse labels, the



(a) Recommendation



(b) Retrieval

Fig. 15. Preference agreement between metric based on degraded labels and complete labels. For each query, we removed a fraction of items from  $\mathcal{R}$  and measured the fraction of preferences  $>_{\mu}$  based on the incomplete relevance judgments that agreed with the preference when using the complete set of relevant items. Relevant items were sampled uniformly at random. Average over 10 samples.

agreement drops. We again see slightly worse degradations with shallower retrieval depths across all metrics. RP in particular demonstrates significantly worse degradation compared to all metrics, while  $R_{1000}$  shows worse degradation when removing relevant items based on ranking frequency.

## 7 Discussion

We begin our discussion by returning to the desiderata in Section 2.4. We originally sought to define and understand recall from a more formal grounding, allowing us to draw connections to recent literature in fairness and robustness, supporting our first desideratum. Moreover, our definition of recall orientation both directly implied the appropriate recall metric and differentiated it from existing metrics, supporting our second desideratum. Finally, our empirical analysis demonstrated that lexirecall captures many of the properties of existing metrics, while being substantially more sensitive and stable in the presence of missing labels, supporting our remaining desideratum. Collectively, we find strong support for investigating lexirecall as a method for assessing our robustness perspective of recall.

In light of our conceptual, theoretical, and empirical analysis, we can make a number of other observations about recall, robustness, and lexicographic evaluation.

### 7.1 Recall

**7.1.1 Labeling.** Although lexirecall appears more stable in the presence of missing labels than existing recall-oriented metrics, the performance of recall and robustness evaluation depends critically on having comprehensive relevance labels. This situation is exacerbated in recommender system environments where judgments, while often highly personalized and based on psychological relevance, can be extremely incomplete (see References [50] for a discussion). This suggests two possibilities. Sparsity in recommendation can arise from (i) very niche relevance or (ii) severe underlabeling of relevant items. In the first situation, our experiments suggest that lexirecall is substantially more sensitive than existing recall metrics. However, in the second situation is accurate, we can examine metric behavior under retrieval scenarios, where labels are more complete. Indeed, our experiments would suggest that using existing recall metrics for sparse recommendation tasks does not accurately reflect recall under more complete data. This echoes similar observations in the recommender system community for addressing data sparsity issues [36, 52, 108].

While time-consuming, we believe that, to develop robust and fair systems, new techniques for expanding labeled sets for recall evaluation are necessary. In retrieval contexts, initiatives like TREC adopt pooling as a strategy to achieve more complete judgments [103]. Unfortunately, in situations where relevance is derived from behavioral feedback (e.g., References [16, 56]), comprehensiveness of relevance is often not the focus. Moreover, the transience of information needs in production environments makes reliable detection of  $\mathcal{R}$  an open problem. This is compounded by the desire—in both retrieval and recommendation contexts—to estimate relevance for evaluation and optimization transparently to the user. As such, new methods for estimating relevance of a broader set of items without impacting the user are necessary.

**7.1.2 Depth Considerations for Recall-oriented Evaluation.** All of our recall-oriented evaluations (e.g., lexirecall, RP,  $R_{1000}$ ) suffer when operating within a shallow retrieval environment. We recommend that, especially for recall-oriented evaluation, retrieval depths be high, regardless of the specific evaluation method. Moreover, as labels become sparser, both RP and  $R_{1000}$  show substantially more ties (Figure 14) and poor stability in the presence of missing labels (Figure 15). We recommend that, for shallow retrieval with sparse labels, RP should be avoided altogether.

### 7.2 Robustness

**7.2.1 Number of Ties and Metric-based Evaluation.** The high number of ties from RP and  $R_{1000}$  arises when collapsing all permutations that share the same recall value. Top-heavy recall-level metrics that have nonzero weight over all relevant items effectively encode the  $\binom{n}{m}$  permutations onto the real line. We should expect more ties and lower statistical sensitivity for metrics that have

low cutoffs (e.g.,  $\tilde{n} < 100$ ). This includes RR and variants of top-heavy recall-level metrics with rank cutoffs (e.g.,  $\text{NDCG}_{10}$ ). Even for top-heavy recall-level metrics, we expect ties if  $e(i) \approx 0$  for unretrieved items or if the numerical precision limits the ability to represent all  $\binom{n}{m}$  positions of relevant items. Because of their top-heaviness, these ties are more likely to occur for differences at the lower ranks, precisely the positions worst-case performance emphasizes. As a result, even though some top-heavy metrics may theoretically *include* worst-case performance, they will not *emphasize* it in the metric value.

**7.2.2 Mixed Orientation Metrics.** We saw agreement between lexirecall and NDCG, even though the latter captures precision orientation (Figure 5). In Section 4.3, we explained that this may be due to either the inclusion of  $p_m$  in top-heavy recall-level metric computation (Equation (1)) or because of structural dependencies between rank positions of  $p_i$  and  $p_j$ . Alternatively, since we observed strong empirical agreement between lexirecall and NDCG, the position of the last ranked relevant item may be predictable because of systematic behavior in the model. For example, for many scenarios, performance higher in the ranking may be predictive of worst-case performance. Even if this is case for many systems or domains, we caution against presuming that performance at the top of the ranking is predictive of worst-case performance. If the worst-case performance is systematic and amongst smaller-sized groups (i.e., those unlikely to appear at the top), then the performance will not be well predicted by larger, systematically higher ranked items from dominant groups. We recommend lexirecall to detect worst-case performance in isolation of other criteria (e.g., precision).

**7.2.3 A Comment on Graded Metrics.** Although we have focused on binary relevance, many ranking scenarios use graded or ordinal relevance. Consider relevance labels represented as an ordinal scale, where higher grades reflect a higher probability of satisfying the information need according to the rater's subjective opinion, as happens in many retrieval scenarios. Under such a grading scheme, an item labeled with the minimum grade has a probability of relevance of 0 (i.e., no user would ever find the item relevant) and an item labeled with the maximum grade has a probability of relevance of  $1 - \epsilon$  (i.e., *almost* every user would find the item relevant). We can determine grades that reflect the probability of relevance through (i) labeling instructions (e.g., "an item with a high grade should satisfy many users; an item with a medium grade should satisfy some users; an item with a low grade should satisfy few users"), (ii) voting schemes [42], or (iii) aggregated behavioral data (e.g., clickthrough rate) [127]. No matter how grades are determined, for a fixed request, a user with a less popular intent may *not* be satisfied by an item relevant to a more popular intent. This implies that notions of optimality in graded ranking evaluation (i.e., that higher grades should be ranked above lower grades) explicitly values dominant group intents over minority intents. From the perspective of robustness, this means that, for graded judgments, we should consider  $\mathcal{R}$  to include all items with a nonzero chance of satisfying a user. This is precisely the approach adopted for TSE and lexirecall.

**7.2.4 Robustness of Optimal Rankings.** In the case of binary relevance, Robertson [82]'s Probability Ranking Principle suggests that an optimal ranking will place all relevant items above non-relevant items. If  $S_n^*$  is the set of optimal permutations, then it consists of permutations that rank all of the items in  $\mathcal{R}$  above  $\mathcal{D} - \mathcal{R}$ . Traditional ranking methods have largely been deterministic insofar as, given a request, they always return a fixed  $\pi \in S_n$ . The *optimal deterministic ranker*, then, is any ranker that selects a fixed  $\pi \in S_n^*$  and, therefore, for any optimal deterministic ranker, the worst-case performance is  $e(m)$ .

The situation changes if we consider stochastic rankers [9, 30, 77, 79, 100], systems that, in response to a request, sample a ranking  $\pi$  from some distribution over  $S_n$ . Such systems have been



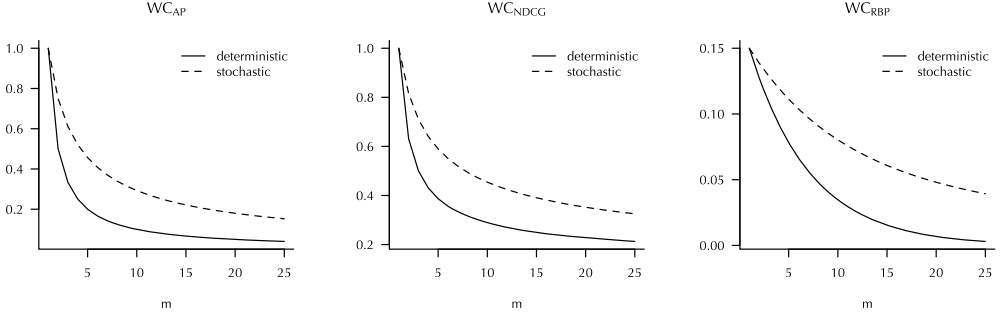


Fig. 16. Worst-case performance of optimal deterministic and stochastic rankings for  $m \in [1 \dots 25]$ .

proposed in the context of online learning [79] and fair ranking [30]. An *optimal stochastic ranker* would, in response to a request, uniformly sample a ranking from  $S_n^*$ .

We can show that the worst-case expected performance of the optimal stochastic ranker is better than the worst-case expected performance for *any* optimal deterministic ranker,

$$\min_{u \in \mathcal{U}} \mathbb{E}_{\pi \sim S_n^*} [\mu(\pi, u)] \geq \min_{u \in \mathcal{U}} \mu(\pi^*, u), \forall \pi^* \in S_n^*.$$

We present a proof in Appendix D. Figure 16 displays the difference in worst-case performance between optimal deterministic and stochastic rankings for  $m \in [1 \dots 25]$ . This result provides evidence from a robustness perspective that ranking system design should explore the design space of stochastic rankers.

### 7.3 Lexicographic Evaluation

**7.3.1 Recovering an Evaluation Metric.** In contrast with preference-based evaluation like lexirecall, metric-based evaluation can be performed efficiently for each ranking independently, moving the complexity from  $O(|\hat{S}_n| \log |\hat{S}_n|)$  to  $O(|\hat{S}_n|)$ . Fortunately, existing results in the computation of leximin point to how to design such a metric.

Yager [123] demonstrates one can construct a leximin representation of a ranking such that  $\text{leximin}(x) > \text{leximin}(y) \leftrightarrow x \succ_{\text{leximin}} y$ . Specifically, if  $x$  is a  $m \times 1$  allocation vector sorted in decreasing order, then

$$\text{leximin}(x) = \sum_{i=1}^m w_i x_i, \quad (12)$$

where  $w$  is a *bottom-heavy* weight vector such that  $w_1 \ll w_2 \ll \dots \ll w_m$ . In our situation, a system “allocates” exposure but, because  $e(p_i)$  is a monotonically decreasing function of  $p_i$ , we only need to compare the rank positions  $p$ . As such, we can define our allocation vectors as  $x_i = \frac{n-p_i}{n}$ . We can use this to define the recall-level weight vector  $w$  as

$$w_i = \begin{cases} \frac{\Delta^{m-1}}{(1+\Delta)^{m-1}} & i = 1 \\ \frac{\Delta^{m-i}}{(1+\Delta)^{m+1-i}} & i > 1 \end{cases},$$

where  $\Delta = (n + \epsilon)^{-1}$  and  $\epsilon \in (0, 1)$  is a free parameter. We can then define *metric lexirecall* as

$$\text{LR}(p) = \text{leximin}(x)$$

$$\propto - \sum_{i=1}^m w_i p_i.$$

We can then compare rankings directly using  $\text{leximin}(p)$ . Since  $\sum_i w_i = 1$ , this can be interpreted as the bottom-heavy weighted average of the positions of relevant items. We contrast this with uniform weighting found in the recall error metric [84] or top-heavy weighting found in precision-oriented metrics.

While providing interesting theoretical connection to existing top-heavy recall-level metrics, in practice, due to the large values of  $n$  and  $p_i$ , computing the metric  $\text{lexirecall}$  can suffer from numerical precision issues. When  $n$  is unknown—for example in dynamic or extremely large corpora—metric  $\text{lexirecall}$  cannot be calculated at all. In these situations, computing  $\text{lexirecall}$  is feasible due to our imputation procedure and the fact that we only care about relative positions.

**7.3.2 Optimization.** Although our focus has been on evaluation, optimizing for lexicographic criteria may be an alternative method for designing recall-oriented algorithms, for example for technology-assisted review or candidate generation. One way to accomplish this is to optimize for metric  $\text{lexirecall}$  discussed in the previous section. Since learning to rank methods often optimize for functions of positions of relevant items (e.g., Reference [78]), standard approaches may suffice. Alternatively, in Section 7.2.4, we observed that optimal stochastic rankers outperformed optimal deterministic rankers in terms of worst-case performance. This suggests that stochastic ranking techniques similar to those developed in the context of other fairness notions (e.g., Reference [30]) can be used for recall-oriented tasks.

## 8 Conclusion

Our analysis indicates that recall, when interpreted as retrieving the totality of relevant items, has conceptual and theoretical connections to worst-case robustness of retrieval effectiveness across possible users. By providing a clear definition of recall orientation, we have directly captured the recall orientation of existing metrics and recognized a missing “basis metric” for recall. Moreover, by developing TSE as the counterpart to RR, we directly connected recall orientation to robustness and Rawlsian notions of fairness, providing a normative argument for improving techniques for gathering complete relevance judgments. Doing so helps ensure the effective computation of recall and, in turn, address potential unfairness. To effectively deploy TSE, we adopted lexicographic evaluation techniques and introduce the  $\text{lexirecall}$  preference-based evaluation method, which we empirically demonstrated was preferable to existing recall metrics. We anticipate that variants of lexicographic evaluation can be applied for other constructs. These three themes of recall, robustness, and lexicographic evaluation, while each individually potentially being interesting areas of theoretical analysis, work collectively to substantially improve our understanding of recall, a concept that may be as old as the field of information retrieval and recommender systems.

## References

- [1] Enrique Amigó, Julio Gonzalo, and Felisa Verdejo. 2013. A general evaluation measure for document organization tasks. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. Association for Computing Machinery, New York, NY, 643–652.
- [2] Salvador Barberà and Matthew Jackson. 1988. Maximin, leximin, and the protective criterion: Characterizations and comparisons. *J. Econ. Theory* 46, 1 (1988), 34–44. [https://doi.org/10.1016/0022-0531\(88\)90148-2](https://doi.org/10.1016/0022-0531(88)90148-2)
- [3] N. J. Belkin, R. N. Oddy, and H. M. Brooks. 1982. ASK for information retrieval: Part I. background and theory. *J. Document.* 38, 2 (Jun. 1982), 61–71.
- [4] Asia J. Biega, Fernando Diaz, Michael D. Ekstrand, and Sebastian Kohlmeier. 2019. Overview of the TREC 2019 fair ranking track. In *Proceedings of the 28th Text REtrieval Conference (TREC'19)*.
- [5] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'18)*. ACM, New York, NY, 405–414.
- [6] Abraham Bookstein. 1979. Relevance. *J. Am. Soc. Inf. Sci.* 30, 5 (1979), 269–273. <https://doi.org/10.1002/asi.4630300505>

- [7] Bert Boyce. 1982. Beyond topicality: A two stage view of relevance and the retrieval process. *Inf. Process. Manage.* 18, 3 (1982), 105–109. [https://doi.org/10.1016/0306-4573\(82\)90033-4](https://doi.org/10.1016/0306-4573(82)90033-4)
- [8] Leonid Boytsov, Anna Belova, and Peter Westfall. 2013. Deciding on an adjustment for multiplicity in IR experiments. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'13)*. Association for Computing Machinery, New York, NY, 403–412. <https://doi.org/10.1145/2484028.2484034>
- [9] Sebastian Bruch, Shuguang Han, Mike Bendersky, and Marc Najork. 2020. A stochastic treatment of learning to rank scoring functions. In *Proceedings of the 13th ACM International Conference on Web Search and Data Mining (WSDM'20)*.
- [10] Chris Buckley and Ellen M. Voorhees. 2004. Retrieval evaluation with incomplete information. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)*. ACM, New York, NY, 25–32.
- [11] Rocío Cañamares and Pablo Castells. 2020. On target item sampling in offline recommender system evaluation. In *Proceedings of the 14th ACM Conference on Recommender Systems (RecSys'20)*. Association for Computing Machinery, New York, NY, 259–268. <https://doi.org/10.1145/3383313.3412259>
- [12] Ben Carterette. 2011. System effectiveness, user models, and user utility: A conceptual framework for investigation. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'11)*. ACM, New York, NY, 903–912.
- [13] Ben Carterette, Evangelos Kanoulas, and Emine Yilmaz. 2012. Incorporating variability in user behavior into systems based evaluation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management (CIKM '12)*. ACM, New York, NY, 135–144.
- [14] Benjamin A. Carterette. 2012. Multiple testing in statistical analysis of systems-based information retrieval experiments. *ACM Trans. Inf. Syst.* 30, 1 (Mar. 2012), 4:1–4:34. <https://doi.org/10.1145/2094072.2094076>
- [15] Pablo Castells, Neil Hurley, and Saúl Vargas. 2022. *Novelty and Diversity in Recommender Systems*. Springer US, New York, NY, 603–646.
- [16] Praveen Chandar, Fernando Diaz, and Brian St. Thomas. 2020. Beyond accuracy: Grounding evaluation metrics for human-machine learning systems. In *Advances in Neural Information Processing Systems*.
- [17] Charles L. A. Clarke, Fernando Diaz, and Negar Arabzadeh. 2023. Preference-based offline evaluation. In *Proceedings of the 16th ACM International Conference on Web Search and Data Mining (WSDM'23)*. Association for Computing Machinery, New York, NY, 1248–1251.
- [18] Kevyn Collins-Thompson, Paul Bennett, Fernando Diaz, Charlie Clarke, and Ellen Voorhees. 2013. TREC 2013 web track overview. In *Proceedings of the 22nd Text Retrieval Conference Proceedings (TREC'13)*. NIST.
- [19] Frederick Conrad, Jason Corey, Samantha Goldstein, Joseph Ostrow, and Michael Sadowsky. 2019. Extreme re-listening: Songs people love . . . and continue to love. *Psychol. Music* 47, 2 (2019), 158–172.
- [20] Harris M. Cooper. 2016. *Research Synthesis and Meta-analysis: A Step-by-step Approach*. SAGE, Thousand Oaks, CA.
- [21] W. S. Cooper. 1971. A definition of relevance for information retrieval. *Inf. Stor. Retriev.* 7, 1 (1971), 19–37. [https://doi.org/10.1016/0020-0271\(71\)90024-6](https://doi.org/10.1016/0020-0271(71)90024-6)
- [22] William S. Cooper. 1968. Expected search length: A single measure of retrieval effectiveness based on the weak ordering action of retrieval systems. *Am. Document.* 19, 1 (1968), 30–41. <https://doi.org/10.1002/asi.5090190108>
- [23] Gordon V. Cormack and Maura R. Grossman. 2016. Engineering quality and reliability in technology-assisted review. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'16)*. Association for Computing Machinery, New York, NY, 75–84. <https://doi.org/10.1145/2911451.2911510>
- [24] Gordon V. Cormack and Maura R. Grossman. 2017. Navigating imprecision in relevance assessments on the road to total recall: Roger and me. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'17)*. Association for Computing Machinery, New York, NY, 5–14.
- [25] Zhuyun Dai, Yubin Kim, and Jamie Callan. 2017. Learning to rank resources. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'17)*. Association for Computing Machinery, New York, NY, 837–840.
- [26] Claude d'Aspremont and Louis Gevers. 1977. Equity and the informational basis of collective choice. *Rev. Econ. Stud.* 44, 2 (1977), 199–209.
- [27] Fernando Diaz. 2015. Condensed list relevance models. In *Proceedings of the International Conference on The Theory of Information Retrieval (ICTIR'15)*. ACM, New York, NY, 313–316.
- [28] Fernando Diaz. 2023. Best-case retrieval evaluation: Improving the sensitivity of reciprocal rank with lexicographic precision. In *Proceedings of the 10th International Workshop on Evaluating Information Access co-located with the 17th NTCIR Conference on the Evaluation of Information Access Technologies (NTCIR'23)*.
- [29] Fernando Diaz and Andres Ferraro. 2022. Offline retrieval evaluation without evaluation metrics. In *Proceedings of the 45th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*.

- [30] Fernando Diaz, Bhaskar Mitra, Michael D. Ekstrand, Asia J. Biega, and Ben Carterette. 2020. Evaluating stochastic rankings with expected exposure. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM'20)*. Association for Computing Machinery, New York, NY, 275–284. <https://doi.org/10.1145/3340531.3411962>
- [31] Zhicheng Dou, Ruihua Song, and Ji-Rong Wen. 2007. A large-scale evaluation and analysis of personalized search strategies. In *Proceedings of the 16th International Conference on World Wide Web (WWW'07)*. ACM, New York, NY, 581–590. <https://doi.org/10.1145/1242572.1242651>
- [32] Nathan Drenkow, Numair Sani, Ilya Shpitser, and Mathias Unberath. 2021. Robustness in deep learning for computer vision: Mind the gap? *CoRR* abs/2112.00639.
- [33] Doris Ruth Eikhof and Axel Haunschild. 2006. Lifestyle meets market: Bohemian entrepreneurs in creative industries. *Creativ. Innov. Manage.* 15, 3 (2006), 234–241. <https://doi.org/10.1111/j.1467-8691.2006.00392.x>
- [34] Michael D. Ekstrand. 2020. LensKit for Python: Next-generation software for recommender system experiments. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*. ACM, 2999–3006. <https://doi.org/10.1145/3340531.3412778>
- [35] Michael D. Ekstrand, Anubrata Das, Robin Burke, and Fernando Diaz. 2022. *Fairness and Discrimination in Information Access Systems*. Foundations and Trends in Information Retrieval.
- [36] Michael D. Ekstrand and Vaibhav Mahant. 2017. Sturgeon and the cool kids: Problems with random decoys for Top-N recommender evaluation. In *Proceedings of the 30th International Florida Artificial Intelligence Research Society Conference*.
- [37] Michael D. Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D. Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. 2018. All the cool kids, how do they fit in?: Popularity and demographic biases in recommender evaluation and effectiveness. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (Proceedings of Machine Learning Research)*, Sorelle A. Friedler and Christo Wilson (Eds.), Vol. 81. PMLR, New York, NY, 172–186.
- [38] Marco Ferrante, Nicola Ferro, and Maria Maistro. 2015. Towards a formal framework for utility-oriented measurements of retrieval effectiveness. In *Proceedings of the International Conference on the Theory of Information Retrieval (ICTIR'15)*. Association for Computing Machinery, New York, NY, 21–30.
- [39] Andres Ferraro, Gustavo Ferreira, Fernando Diaz, and Georgina Born. 2022. Measuring commonality in recommendation of cultural content: Recommender systems to enhance cultural citizenship. In *Proceedings of the 16th ACM Conference on Recommender Systems*.
- [40] Peter Flach and Meelis Kull. 2015. Precision-recall-gain curves: PR analysis done right. In *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (Eds.), Vol. 28. Curran Associates, Inc.
- [41] W. B. Gallie. 1955. Essentially contested concepts. *Proc. Arist. Soc.* 56 (1955), 167–198.
- [42] Mitchell L. Gordon, Kaitlyn Zhou, Kayur Patel, Tatsunori Hashimoto, and Michael S. Bernstein. 2021. *The Disagreement Deconvolution: Bringing Machine Learning Performance Metrics in Line with Reality*. Association for Computing Machinery, New York, NY. <https://doi.org/10.1145/3411764.3445423>
- [43] Gregory Goren, Oren Kurland, Moshe Tennenholtz, and Fiana Raiber. 2018. Ranking robustness under adversarial document manipulations. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'18)*. Association for Computing Machinery, New York, NY, 395–404. <https://doi.org/10.1145/3209978.3210012>
- [44] Allègre Hadida. 2015. Performance in the creative industries. In *The Oxford Handbook of Creative Industries*. Oxford University Press. <https://doi.org/10.1093/oxfordhb/9780199603510.013.018>
- [45] Stephen P. Harter. 1992. Psychological relevance and information science. *J. Am. Soc. Inf. Sci.* 43, 9 (1992), 602–615.
- [46] Stephen P. Harter. 1996. Variations in relevance assessments and the measurement of retrieval effectiveness. *J. Am. Soc. Inf. Sci.* 47, 1 (1996), 37–49.
- [47] Helia Hashemi, Mohammad Aliannejadi, Hamed Zamani, and W. Bruce Croft. 2020. ANTIQUE: A non-factoid question answering benchmark. In *Advances in Information Retrieval*, Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins (Eds.). Springer International Publishing, Cham, 166–173.
- [48] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. 2018. Fairness without demographics in repeated loss minimization. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research)*, Jennifer Dy and Andreas Krause (Eds.), Vol. 80. PMLR, Stockholm, Sweden, 1929–1938.
- [49] Hoda Heidari, Michele Loi, Krishna P. Gummadi, and Andreas Krause. 2019. A moral framework for understanding fair ML through economic models of equality of opportunity. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT\*19)*. Association for Computing Machinery, New York, NY, 181–190.

- [50] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* 22, 1 (Jan. 2004), 5–53. <https://doi.org/10.1145/963770.963772>
- [51] Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian McAuley. 2024. Bridging language and items for retrieval and recommendation. arXiv:2403.03952. Retrieved from <https://arxiv.org/abs/2403.03952>
- [52] Ngozi Ihemelandu and Michael D. Ekstrand. 2023. Candidate set sampling for evaluating Top-N recommendation. In *Proceedings of the IEEE/WIC International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'23)*. 88–94.
- [53] Abigail Z. Jacobs and Hanna Wallach. 2021. Measurement and fairness. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (FAccT'21)*. Association for Computing Machinery, New York, NY, 375–385.
- [54] Olivier Jeunen, Ivan Potapov, and Aleksei Ustimenko. 2024. On (normalised) discounted cumulative gain as an off-policy evaluation metric for top-n recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'24)*. Association for Computing Machinery, New York, NY, 1222–1233. <https://doi.org/10.1145/3637528.3671687>
- [55] Gabriella Kazai and Mounia Lalmas. 2006. EXtended cumulated gain measures for the evaluation of content-oriented XML retrieval. *ACM Trans. Inf. Syst.* 24, 4 (Oct. 2006), 503–542.
- [56] Diane Kelly and Jaime Teevan. 2003. Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum* 37, 2 (Sep. 2003), 18–28. <https://doi.org/10.1145/959258.959260>
- [57] Serge-Christophe Kolm. 2002. *Justice and Equity*. MIT Press.
- [58] Liu Leqi, Fatma Kilinc-Karzan, Zachary C. Lipton, and Alan L. Montgomery. 2021. Rebounding bandits for modeling satiation effects. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*. ACM, New York, NY.
- [59] David D. Lewis, Eugene Yang, and Ophir Frieder. 2021. *Certifying One-Phase Technology-Assisted Reviews*. Association for Computing Machinery, New York, NY, 893–902. <https://doi.org/10.1145/3459637.3482415>
- [60] Cheng Li, Yue Wang, Paul Resnick, and Qiaozhu Mei. 2014. ReQ-ReC: High recall retrieval with query pooling and interactive classification. In *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'14)*. Association for Computing Machinery, New York, NY, 163–172.
- [61] Malte Ludewig and Dietmar Jannach. 2018. Evaluation of session-based recommendation algorithms. *User Model. User-Adapt. Interact.* 28, 4–5 (Dec. 2018), 331–390.
- [62] Mihai Lupu, Katja Mayer, Noriko Kando, and Anthony J. Trippe. 2017. *Current Challenges in Patent Information Retrieval*. Springer, Berlin.
- [63] Craig Macdonald, Rodrygo L. T. Santos, and Iadh Ounis. 2013. The whens and hows of learning to rank for web search. *Inf. Retrieval* 16, 5 (2013), 584–628.
- [64] Iain Mackie, Shubham Chatterjee, and Jeffrey Dalton. 2023. Generative and pseudo-relevant feedback for sparse, dense and learned sparse retrieval. arXiv:cs.LR/2305.07477. Retrieved from <https://arxiv.org/abs/2305.07477>
- [65] Walid Magdy and Gareth J. F. Jones. 2010. PRES: A score metric for evaluating recall-oriented information retrieval applications. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. Association for Computing Machinery, New York, NY, 611–618. <https://doi.org/10.1145/1835449.1835551>
- [66] Walid Magdy and Gareth J. F. Jones. 2010. Examining the robustness of evaluation metrics for patent retrieval with incomplete relevance judgements. In *Multilingual and Multimodal Information Access Evaluation*, Maristella Agosti, Nicola Ferro, Carol Peters, Maarten de Rijke, and Alan Smeaton (Eds.). Springer, Berlin, 82–93.
- [67] Rishabh Mehrotra, Amit Sharma, Ashton Anderson, Fernando Diaz, Hanna Wallach, and Emine Yilmaz. 2017. Auditing search engines for differential satisfaction across demographics. In *Proceedings of the 26th International Conference on World Wide Web*.
- [68] Omid Memarrast, Ashkan Rezaei, Rizal Fathony, and Brian D. Ziebart. 2021. Fairness for robust learning to rank. arXiv:2112.06288. Retrieved from <https://arxiv.org/abs/2112.06288>
- [69] Martin Mladenov, Elliot Creager, Omer Ben-Porat, Kevin Swersky, Richard S. Zemel, and Craig Boutilier. 2020. Optimizing long-term social welfare in recommender systems: A constrained matching approach. In *Proceedings of the 37th International Conference on Machine Learning (ICML'20)*, Vol. 119. PMLR, 6987–6998.
- [70] Bamshad Mobasher, Robin Burke, Runa Bhaumik, and Chad Williams. 2007. Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Trans. Internet Technol.* 7, 4 (Oct. 2007), 23–38. <https://doi.org/10.1145/1278366.1278372>
- [71] Alistair Moffat. 2013. Seven numeric properties of effectiveness metrics. In *Information Retrieval Technology*, Rafael E. Banchs, Fabrizio Silvestri, Tie-Yan Liu, Min Zhang, Sheng Gao, and Jun Lang (Eds.). Springer, Berlin, 1–12.
- [72] Alistair Moffat and Justin Zobel. 2008. Rank-biased precision for measurement of retrieval effectiveness. *ACM Trans. Inf. Syst.* 27, 1 (Dec. 2008), 2:1–2:27.



- [73] Hafeezul Rahman Mohammad, Keyang Xu, Jamie Callan, and J. Shane Culpepper. 2018. Dynamic shard cutoff prediction for selective search. In *Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval (SIGIR'18)*. Association for Computing Machinery, New York, NY, 85–94.
- [74] Ali MontazerAlghaem, Hamed Zamani, and James Allan. 2020. A reinforcement learning framework for relevance feedback. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'20)*. Association for Computing Machinery, New York, NY, 59–68.
- [75] Hervé Moulin. 2003. *Fair Division and Collective Welfare*. MIT Press.
- [76] Nicola Neophytou, Bhaskar Mitra, and Catherine Stinson. 2022. Revisiting popularity and demographic biases in recommender evaluation and effectiveness. In *Advances in Information Retrieval*, Matthias Hagen, Suzan Verberne, Craig Macdonald, Christin Seifert, Krisztian Balog, Kjetil Nørvåg, and Vinay Setty (Eds.). Springer International Publishing, Cham, 641–654.
- [77] Harrie Oosterhuis and Maarten de Rijke. 2020. Policy-aware unbiased learning to rank for Top-k rankings. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'20)*. Association for Computing Machinery, New York, NY, 489–498.
- [78] Tao Qin, Tie-Yan Liu, and Hang Li. 2010. A general approximation framework for direct optimization of information retrieval measures. *Inf. Retrieval* 13, 4 (2010), 375–397. <https://doi.org/10.1007/s10791-009-9124-x>
- [79] Filip Radlinski and Thorsten Joachims. 2007. Active exploration for learning rankings from clickthrough data. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, New York, NY, 570–579.
- [80] John Rawls. 1974. Some reasons for the maximin criterion. *Am. Econ. Rev.* 64, 2 (1974), 141–146.
- [81] John Rawls. 2001. *Justice as Fairness: A Restatement*. Harvard University Press.
- [82] Stephen Robertson. 1977. The probability ranking principle. *J. Document.* (1977).
- [83] Stephen Robertson. 2008. A new interpretation of average precision. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*. Association for Computing Machinery, New York, NY, 689–690.
- [84] Joseph John Rocchio. 1964. Performance indices for document retrieval systems. In *Information Storage and Retrieval*, Gerald Salton (Ed.). Number ISR-8. Harvard University, Chapter III.
- [85] Adam Roegiest, Gordon V. Cormack, Charles L. A. Clarke, and Maura R. Grossman. 2015. TREC 2015 total recall track overview. In *Proceedings of the 24th Text REtrieval Conference TREC'15*, Ellen M. Voorhees and Angela Ellis (Eds.), Vol. 500-319. National Institute of Standards and Technology (NIST).
- [86] Rachel Rudinger, Pushpendre Rastogi, Francis Ferraro, and Benjamin Van Durme. 2015. Script induction as language modeling. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 1681–1686.
- [87] Ian Ruthven. 2014. Relevance behaviour in TREC. *J. Document.* 70, 6 (2022/12/02 2014), 1098–1117. <https://doi.org/10.1108/JD-02-2014-0031>
- [88] Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. 2018. Assessing generative models via precision and recall. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS'18)*. Curran Associates Inc., Red Hook, NY, 5234–5243.
- [89] Tetsuya Sakai. 2006. *A Further Note on Evaluation Metrics for the Task of Finding One Highly Relevant Document*. Technical Report 33(2006-DD-054). Toshiba Corporate R&D Center.
- [90] Tetsuya Sakai. 2014. Metrics, statistics, tests. In *Bridging between Information Retrieval and Databases—PROMISE Winter School 2013, Revised Tutorial Lectures (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics))*. Springer Verlag, Berlin, 116–163. [https://doi.org/10.1007/978-3-642-54798-0\\_6](https://doi.org/10.1007/978-3-642-54798-0_6). 2013
- [91] Tetsuya Sakai and Noriko Kando. 2008. On information retrieval metrics designed for evaluation with incomplete relevance assessments. *Inf. Retrieval* 11, 5 (2008), 447–470.
- [92] Tetsuya Sakai and Stephen Robertson. 2008. Modelling a user population for designing information retrieval metrics. In *Proceedings of the 2nd International Workshop on Evaluating Information Access (EVIA'08)*.
- [93] Mark Sanderson, Monica Lestari Paramita, Paul Clough, and Evangelos Kanoulas. 2010. Do user preferences and evaluation measures line up? In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*. Association for Computing Machinery, New York, NY, 555–562.
- [94] Rodrygo L. T. Santos, Craig Macdonald, and Iadh Ounis. 2015. *Search Result Diversification*. Foundations and Trends in Information Retrieval, Vol. 9. Now Publishers Inc., Hanover, MA.
- [95] Amartya Sen. 1970. *Collective Choice and Social Welfare*. Holden-Day.
- [96] Amartya Sen. 1976. Welfare inequalities and Rawlsian axiomatics. *Theory Decis.* 7, 4 (1976), 243–262.
- [97] Amartya Sen. 1977. On weights and measures: Informational constraints in social welfare analysis. *Econometrica* 45, 7 (1977), 1539–1572.

- [98] Kulin Shah, Pooja Gupta, Amit Deshpande, and Chiranjib Bhattacharyya. 2021. Rawlsian fair adaptation of deep learning classifiers. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AI/ES'21)*. Association for Computing Machinery, New York, NY, 936–945.
- [99] Henry Sidgwick. 2011. *The Methods of Ethics*. Cambridge University Press.
- [100] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'18)*. ACM, New York, NY, 2219–2228. <https://doi.org/10.1145/3219819.3220088>
- [101] Mark D. Smucker, James Allan, and Ben Carterette. 2007. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM Conference on Conference on Information and Knowledge Management (CIKM'07)*. Association for Computing Machinery, New York, NY, 623–632. <https://doi.org/10.1145/1321440.1321528>
- [102] David Sontag, Kevyn Collins-Thompson, Paul N. Bennett, Ryan W. White, Susan Dumais, and Bodo Billerbeck. 2012. Probabilistic models for personalizing web search. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining (WSDM'12)*. Association for Computing Machinery, New York, NY, 433–442.
- [103] Karen Spärck Jones and Cornelis J. van Rijsbergen. 1975. *Report on the Need for and Provision for an 'IDEAL' Information Retrieval Test Collection*. British Library Research and Development Report 5266. Computer Laboratory, University of Cambridge.
- [104] Raji Srinivasan and Gülen Sarial-Abi. 2021. When algorithms fail: Consumers' responses to brand harm crises caused by algorithm errors. *J. Market.* 85, 5 (2021), 74–91.
- [105] Yan-Martin Tamm, Rinchin Damdinov, and Alexey Vasilev. 2021. Quality metrics in recommender systems: Do we calculate metrics consistently? In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'21)*. Association for Computing Machinery, New York, NY, 708–713.
- [106] Jaime Teevan, Susan T. Dumais, and Eric Horvitz. 2010. Potential for personalization. *ACM Trans. Comput.-Hum. Interact.* 17, 1 (Apr. 2010), 4:1–4:31. <https://doi.org/10.1145/1721831.1721835>
- [107] The Sedona Conference. 2009. Commentary on achieving quality in the E-Discovery process. *Sedona Conf. J.* 10 (2009).
- [108] Mucun Tian and Michael D. Ekstrand. 2020. Estimating error and bias in offline evaluation results. In *Proceedings of the Conference on Human Information Interaction and Retrieval (CHIIR'20)*. Association for Computing Machinery, New York, NY, 392–396.
- [109] Stephen Tomlinson and Bruce Hedin. 2017. *Measuring Effectiveness in the TREC Legal Track*. Springer, Berlin, 163–182.
- [110] Anthony Trippe and Ian Ruthven. 2017. *Evaluating Real Patent Retrieval Effectiveness*. Springer, Berlin, 143–162.
- [111] Daniel Valcarce, Alejandro Bellogin, Javier Parapar, and Pablo Castells. 2020. Assessing ranking metrics in top-N recommendation. *Inf. Retrieval J.* 23, 4 (2020), 411–448.
- [112] E. M. Voorhees. 2004. Overview of the TREC 2004 robust track. In *Proceedings of the 13th Text REtrieval Conference (TREC'04)*.
- [113] Ellen M. Voorhees. 1998. Variations in relevance judgments and the measurement of retrieval effectiveness. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*. Association for Computing Machinery, New York, NY, 315–323. <https://doi.org/10.1145/290941.291017>
- [114] Ellen M. Voorhees and Donna K. Harman. 1997. Overview of the fifth text retrieval conference (TREC-5). In *Proceedings of the 5th Text REtrieval Conference (TREC-5)*.
- [115] Sanne Vrijenhoek, Gabriel Bénédict, Mateo Gutierrez Granada, Daan Odijk, and Maarten De Rijke. 2022. RADio – rank-aware divergence metrics to measure normative diversity in news recommendations. In *Proceedings of the 16th ACM Conference on Recommender Systems (RecSys'22)*. Association for Computing Machinery, New York, NY, 208–219.
- [116] Sanne Vrijenhoek, Mesut Kaya, Nadia Metoui, Judith Möller, Daan Odijk, and Natali Helberger. 2021. Recommenders with a mission: Assessing diversity in news recommendations. In *Proceedings of the Conference on Human Information Interaction and Retrieval (CHIIR'21)*. Association for Computing Machinery, New York, NY, 173–183.
- [117] Sanne Vrijenhoek, Lien Michiels, Johannes Kruse, Jordi Viader Guerrero, Alain Starke, and Nava Tintarev (Eds.). 2023. *Proceedings of the 1st Workshop on Normative Design and Evaluation of Recommender Systems*.
- [118] Lidan Wang, Paul N. Bennett, and Kevyn Collins-Thompson. 2012. Robust ranking models via risk-sensitive optimization. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*. Association for Computing Machinery, New York, NY, 761–770. <https://doi.org/10.1145/2348283.2348385>
- [119] Morgan K. Ward, Joseph K. Goodman, and Julie R. Irwin. 2014. The same old song: The power of familiarity in music choice. *Market. Lett.* 25, 1 (2014), 1–11.



- [120] Hongyi Wen, Xinyang Yi, Tiansheng Yao, Jiayi Tang, Lichan Hong, and Ed H. Chi. 2022. Distributionally-robust recommendations for improving worst-case user experience. In *Proceedings of the ACM Web Conference 2022 (WWW'22)*. Association for Computing Machinery, New York, NY, 3606–3610. <https://doi.org/10.1145/3485447.3512255>
- [121] Ziang Xiao, Susu Zhang, Vivian Lai, and Q. Vera Liao. 2023. Evaluating evaluation metrics: A framework for analyzing NLG evaluation metrics using measurement theory. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 10967–10982.
- [122] Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. 2020. Adversarial counterfactual learning and evaluation for recommender system. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20)*. Curran Associates Inc., Red Hook, NY.
- [123] Ronald R. Yager. 1997. On the analytic representation of the Leximin ordering and its application to flexible constraint propagation. *Eur. J. Operat. Res.* 102, 1 (1997), 176–192. [https://doi.org/10.1016/S0377-2217\(96\)00217-2](https://doi.org/10.1016/S0377-2217(96)00217-2)
- [124] Hamed Zamani and W. Bruce Croft. 2020. Learning a joint search and recommendation model from user-item interactions. In *Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM'20)*. Association for Computing Machinery, New York, NY, 717–725.
- [125] Xueru Zhang, Mohammadmahdi Khaliligarekani, Cem Tekin, and mingyan liu. 2019. Group retention when using machine learning in sequential decision making: The interplay between user dynamics and fairness. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 15269–15278.
- [126] Wayne Xin Zhao, Zihan Lin, Zhichao Feng, Pengfei Wang, and Ji-Rong Wen. 2022. A revisiting study of appropriate offline evaluation for Top-N recommendation algorithms. *ACM Trans. Inf. Syst.* 41, 2, Article 32 (Dec. 2022), 41 pages. <https://doi.org/10.1145/3545796>
- [127] Hua Zheng, Dong Wang, Qi Zhang, Hang Li, and Tinghao Yang. 2010. Do clicks measure recommendation relevancy? An empirical user study. In *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys'10)*. Association for Computing Machinery, New York, NY, 249–252. <https://doi.org/10.1145/1864708.1864759>
- [128] Ziwei Zhu, Jingu Kim, Trung Nguyen, Aish Fenton, and James Caverlee. 2021. Fairness among new items in cold start recommender systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'21)*. Association for Computing Machinery, New York, NY, 767–776. <https://doi.org/10.1145/3404835.3462948>
- [129] Justin Zobel. 1998. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*. Association for Computing Machinery, New York, NY, 307–314.
- [130] Justin Zobel, Alistair Moffat, and Laurence A. F. Park. 2009. Against recall: Is it persistence, cardinality, density, coverage, or totality? *SIGIR Forum* 43, 1 (Jun. 2009), 3–8. <https://doi.org/10.1145/1670598.1670600>
- [131] Jie Zou and Evangelos Kanoulas. 2020. Towards question-based high-recall information retrieval: Locating the last few relevant documents for technology-assisted reviews. *ACM Trans. Inf. Syst.* 38, 3 (May 2020). <https://doi.org/10.1145/3388640>

## Appendices

### A Full Papers Published at the ACM Conference on Recommender Systems That Measure Recall

We downloaded and reviewed all full papers published at the ACM Conference on Recommender Systems (2007–2024), omitting reproducibility, late-breaking results, short, demo, doctoral, or keynote papers. A paper was identified if recall was quantitatively measured as part of experimentation. Conceptual discussions of recall (e.g., in comparison to experimentation metrics) were not considered to use recall. We identified only those papers that specifically name recall (e.g.,  $R_k$ ), although some operationalizations of hit rate may be equivalent to recall [105]. We present the full papers identified in Table 7.

Table 6. Notation

$\mathbb{Z}^+$	positive integers
$\mathbb{Z}^*$	non-negative integers
$\mathbb{R}^+$	positive reals
$\mathbb{R}^*$	non-negative reals
$S_k$	set of all permutations of $k$ items
$\mathcal{A}^+$	non-empty subsets of $\mathcal{A}$
$\mathbf{I} : X \rightarrow \{0, 1\}$	indicator function (i.e., returns 1 if $X$ is true; 0 otherwise)
$x_{-i}$	reverse index (i.e., $x_{k-i+1}$ for the $k$ -dimensional vector $x$ )
$\mathcal{D}$	corpus
$\mathcal{R}$	relevant set
$n$	size of corpus (i.e., $ \mathcal{D} $ )
$m$	size of relevant set (i.e., $ \mathcal{R} $ )
$\tilde{n}$	number of items retrieved
$S_n$	set of all permutations of $\mathcal{D}$
$\tilde{S}_n$	subset of $S_n$ generated by multiple systems for a fixed request
$\pi$	permutation of $\mathcal{D}$
$p$	sorted positions of relevant items
$\bar{p}$	item ids of relevant items sorted by position
$C$	number of unique permutations in $S_n$ for a given $p$ (i.e., $m!(n-m)!$ )
$\mu : S_n \times \mathcal{D}^+ \rightarrow \mathbb{R}^*$	user evaluation metric
$\eta : S_n \times \mathcal{D}^+ \rightarrow \mathbb{R}^*$	provider evaluation metric
$\mu(\pi, \mathcal{R}) \stackrel{\text{rank}}{=} \mu'(\pi, \mathcal{R})$	$\mu$ and $\mu'$ rank $S_n$ identically
$e : \mathbb{Z}^+ \rightarrow \mathbb{R}^*$	exposure of position
$z : \mathbb{Z}^+ \times \mathbb{Z}^+ \rightarrow \mathbb{R}^*$	normalization function
$\mathcal{U}$	set of possible user information needs for a request
$\mathcal{V}$	set of possible providers for a request

Table 7. DOIs of Full Papers Published at the ACM Conference on Recommender Systems (2007–2024) That Measure Recall in Experiments

year	total	recall	DOIs
2007	16	3	10.1145/1297231.12972{48, 50, 51}
2008	20	3	10.1145/1454008.14540{17, 32, 34}
2009	25	4	10.1145/1639714.16397{24, 26, 37, 45}
2010	25	8	10.1145/1864708.18647{21, 28, 31, 32, 41, 44, 45, 47}
2011	22	5	10.1145/2043932.20439{41, 45, 47, 57, 65}
2012	24	11	10.1145/2365952.23659{62, 63, 67, 68, 69, 72, 73, 76, 79, 82, 89}
2013	32	11	10.1145/2507157.2507{160, 170, 171, 172, 182, 184, 186, 209, 210, 211, 215}
2014	35	8	10.1145/2645710.26457{21, 23, 29, 34, 38, 40, 43, 46}
2015	28	7	10.1145/2792838.28001{70, 72, 74, 76, 85, 89, 95}
2016	29	12	10.1145/2959100.29591{33, 37, 46, 47, 49, 51, 57, 67, 70, 78, 80, 82}
2017	26	6	10.1145/3109859.31098{77, 879, 887, 896, 900, 903}
2018	32	11	10.1145/3240323.3240{343, 347, 349, 355, 363, 371, 372, 374, 381, 391, 405}
2019	36	11	10.1145/3298689.334{6996, 7002, 7007, 7009, 7012, 7013, 7026, 7036, 7044, 7058, 7065}
2020	39	12	10.1145/3383313.341{1476, 2232, 2235, 2243, 2247, 2248, 2249, 2258, 2259, 2262, 2265, 2268}
2021	49	19	10.1145/3460231.3474{228, 230, 234, 238, 240, 242, 249, 252, 255, 257, 260, 263, 265, 266, 268, 270, 272, 273, 275}
2022	39	13	10.1145/3523227.3546{752, 754, 755, 760, 762, 763, 768, 770, 771, 775, 782, 784, 785}
2023	47	21	10.1145/3604915.3608{766, 771, 773, 781, 783, 784, 785, 786, 803, 804, 806, 809, 810, 811, 812, 815, 863, 868, 871, 878, 882}
2024	58	24	10.1145/3640457.3688{096, 098, 100, 104, 108, 109, 113, 117, 121, 122, 123, 124, 125, 127, 133, 137, 138, 139, 145, 146, 148, 149, 151, 153}

## B Metric Properties

We can connect our proposed metrics to prior work by leveraging several properties defined in the community. Drawing on fundamental contributions from Moffat [71], Ferrante et al. [38], and Amigó et al. [1], we produced a synthesized list of properties. These properties should be considered descriptive—not prescriptive—since (i) they can be in tension and (ii) there are valid metrics (e.g., high-precision, diversity, fairness) that do not satisfy all of them. Let  $\tilde{\pi}$  be a top- $\tilde{n}$  from which we impute a total ranking using pessimistic imputation (Section 2.2).

Consistent with our setup, we assume that the evaluator has access to a set of fixed relevance assessments  $\mathcal{R}$  over a fixed corpus  $\mathcal{D}$  and pessimistic imputation of a top- $\tilde{n}$  ranking  $\tilde{\pi}$ . Given a ranking  $\pi$ , let  $y_i$  refer to the relevance of  $\pi_i$ . Finally, we assume that the number of unretrieved  $n - \tilde{n}$  items is larger than the number of relevant items  $m$ , which is the case in most ranking tasks.

- (1) *Monotonicity in retrieval size* [1, 71]. This property refers to the behavior of  $\mu$  as we append items to  $\tilde{\pi}$ . Following Moffat [71], a metric is *monotonically increasing in retrieval size* if it is non-decreasing as  $\tilde{n}$  is increased by appending either relevant or nonrelevant items to  $\tilde{\pi}$ . Following Amigó et al. [1], a metric is *(strictly) monotonically decreasing in nonrelevance* if it (strictly) decreases as  $\tilde{n}$  is increased by appending nonrelevant items to  $\tilde{\pi}$ ; Amigó et al. [1] refer to the strict version of this as “Confidence.” Note that the Moffat [71] and Amigó et al. [1] properties are in tension. For completeness, we refer to a metric as *(strictly)*

*monotonically increasing in relevance* if it (strictly) increases as  $\tilde{n}$  is increased by appending relevant items to  $\tilde{\pi}$ .

- (2) *Monotonicity in swapping up* [1, 38, 71]. A metric is (strictly) *monotonically increasing in swapping up* if, when  $i < j$  and  $y_i < y_j$ , we observe a (strictly) monotonic increase in  $\mu$  when we swap the items at positions  $i$  and  $j$ . Ferrante et al. [38] refers to the non-decreasing property as “Swap.” When  $j > \tilde{n}$ , Moffat [71] refers to the strictly increasing property as “Convergence” and Ferrante et al. [38] refers to the non-decreasing property as “Replacement.” When  $j \leq \tilde{n}$ , Moffat [71] refers to the strictly increasing property as “Top-weightedness.” For contiguous swaps (i.e.,  $j = i + 1$ ), Amigó et al. [1] refer to the strictly increasing property as “Priority.”
- (3) *Concavity in contiguous swap depth* [1]. A metric is (strictly) *concave in contiguous swap depth* if, when  $i < j$  and  $y_i < y_{i+1}$  and  $y_j < y_{j+1}$ , swapping  $i$  and  $i + 1$  will lead to a (strictly) larger improvement in  $\mu$  compared to swapping  $j$  and  $j + 1$ . Amigó et al. [1] refer this to “Deepness.”
- (4) *Suffix Invariance* [1]. Given two rankings  $\pi$  and  $\pi'$  that have relevant items in same positions in the top- $\tilde{n}$  prefix, a metric is *suffix invariant* if, no matter the positions of the remaining relevant items, the metric values will be the same. Amigó et al. [1] refer this to “Deepness Threshold.”
- (5) *Prefix Invariance* [1]. Given two rankings  $\pi$  and  $\pi'$  that have relevant items in same positions in the bottom- $k$  suffix, a metric is *prefix invariant* if, no matter the positions of the remaining relevant items, the metric values will be the same. Amigó et al. [1] refer this to “Closeness Threshold.”
- (6) *Boundedness* [71]. A metric is *bounded from above* if there exists an  $U_\mu \in \mathfrak{R}$  such that,

$$\forall \mathcal{R} \in \mathcal{D}^+, \forall \pi \in S_n : \mu(\pi, \mathcal{R}) \leq U_\mu.$$

Similarly, a metric is *bounded from below* if there exists an  $L_\mu \in \mathfrak{R}$  such that,

$$\forall \mathcal{R} \in \mathcal{D}^+, \forall \pi \in S_n : \mu(\pi, \mathcal{R}) \geq L_\mu.$$

In general, a metric is *bounded* if it is bounded from above and below.

- (7) *Localization* [71]. A metric is *localized* if can be computed only with the information in the top- $\tilde{n}$ ; in other words, the metric value is suffix invariant *and* independent of the number of unretrieved relevant items.
- (8) *Completeness* [71]. A metric is *complete* if it is defined when  $m = 0$  (i.e., metrics are of the form  $\mu : S_n \times \mathcal{D}^* \rightarrow \mathfrak{R}$ ).
- (9) *Realizability* [71]. A metric is *realizable above* if it is bounded from above and

$$\forall \mathcal{R} \in \mathcal{D}^+, \exists \pi \in S_n : \mu(\pi, \mathcal{R}) = U_\mu.$$

A metric is *realizable below* if it is bounded from below and

$$\forall \mathcal{R} \in \mathcal{D}^+, \exists \pi \in S_n : \mu(\pi, \mathcal{R}) = L_\mu.$$

If  $m > 0$ , then Moffat [71] refers to metrics that are realizable above as simply “Realizable.”

In some cases, we have adopted a property name different from the original to help with clarity. In subsequent sections, we will demonstrate which of these properties are present for top-heavy recall-level metrics, TSE, and lexirecall.

Several properties were not present in any of our evaluation methods. None of our methods are strictly decreasing in nonrelevance. Amigó et al. [1] note also that, “[a]s far as we know, current evaluation measures do not consider this aspect.” None of our methods are prefix or suffix invariant. This is largely due to the fact that (i) exposure is strictly monotonically decreasing, and (ii) normalization is a function of recall level (and the number of relevant items). As a result, any position-based “flatness” in the computation is missing. None of our methods are guaranteed

Table 8. Metric Properties

	THRL	THRL $z(i, m) > 0$	TSE	LR
increasing in retrieval size [71]	✓	✓	✓	✓
decreasing in nonrelevance	✓	✓	✓	✓
strictly decreasing in nonrelevance [1]				
increasing in relevance	✓	✓	✓	✓
strictly increasing in relevance		✓		✓
increasing in swapping up [38]	✓	✓	✓	✓
strictly increasing in swapping up [1, 71]		✓		
concavity in contiguous swap depth	✓	✓	✓	NA
strict concavity in contiguous swap depth [1]		✓ <sup>1</sup>		NA
suffix invariance [1]				
prefix invariance [1]				
boundedness [71]				NA
localization [71]				NA
completeness [71]	✓ <sup>2</sup>	✓ <sup>2</sup>	✓ <sup>2</sup>	NA
realizability [71]				NA

1: if the second derivative of the exposure function is strictly positive. 2: if normalization function is defined for  $m = 0$ . The properties concavity, boundedness, localization, completeness, and realizability are specific to metric-based evaluation and therefore cannot be analyzed for preference-based evaluation like lexirecall.

to be bounded to allow maximal flexibility in our analysis; this also means that none of our methods are guaranteed to be realizable. None of our methods are localized because we explicitly use  $m$  and  $n$  in pessimistic imputation; while lexirecall does not need  $n$ , it does still require  $m$ .

We summarize the properties for top-heavy recall-level metrics, total search efficiency, and lexirecall in Table 8.

### B.1 Properties of Top-Heavy Recall-Level Metrics

Let  $\mu$  be a top-heavy recall-level metric with exposure function  $e$  and normalization function  $z$ .

**THEOREM B.1.**  $\mu$  is monotonically increasing in retrieval size.

**PROOF.** Let  $\tilde{\pi}'$  be  $\tilde{\pi}$  with another item appended, with pessimistically imputed rankings  $\pi'$  and  $\pi$ . If the new item is nonrelevant, then  $\forall i, p_i = p'_i$  and, therefore,  $\mu(\pi, \mathcal{R}) = \mu(\pi', \mathcal{R})$  and  $\mu$  is trivially non-decreasing. Next, consider the case where the new item is relevant. This can only happen if  $\tilde{\pi}$  includes  $\tilde{m} < m$  relevant items. As such, this is equivalent to swapping a relevant item in the imputed ranking  $\pi$  from position  $n - (m - \tilde{m} - 1)$  to position  $\tilde{n} + 1$ . Let  $\Delta_{i,j}\mu(\pi, \mathcal{R})$  be the difference in metric value from swapping a relevant item in position  $j$  to position  $i$ ,

$$\begin{aligned}
\Delta_{\tilde{n}+1, n-(m-\tilde{m}-1)}\mu(\pi, \mathcal{R}) &= \mu(\pi', \mathcal{R}) - \mu(\pi, \mathcal{R}) \\
&= \sum_{i=1}^m e(p'_i)z(i, m) - \sum_{i=1}^m e(p_i)z(i, m) \\
&= e(p'_{\tilde{m}+1})z(\tilde{m} + 1, m) - e(p_{\tilde{m}+1})z(\tilde{m} + 1, m) \\
&= z(\tilde{m} + 1, m)(e(\tilde{n} + 1) - e(n - (m - \tilde{m} - 1))). \tag{13}
\end{aligned}$$

Since  $z(i, m) \geq 0$  and  $e(\tilde{n} + 1) > e(n - (m - \tilde{m} - 1))$ , we know that Equation (13) will always be non-negative.  $\square$

**THEOREM B.2.**  $\mu$  is monotonically decreasing in nonrelevance.

**PROOF.** See the first case in the proof Theorem B.1.  $\square$

**THEOREM B.3.** If  $z(i, m)$  is (strictly) positive, then  $\mu$  is (strictly) monotonically increasing in relevance.

**PROOF.** See the second case in the proof Theorem B.1. Moreover, if  $z(i, m)$  is strictly positive, then Equation (13) will always be strictly positive.  $\square$

**THEOREM B.4.** If  $z(i, m)$  is (strictly) positive, then  $\mu$  is (strictly) monotonically increasing in swapping up.

**PROOF.** Given a ranking  $\pi \in S_n$ , let  $j$  be the  $j$ th relevant item and  $k < p_j$  the position of an arbitrary nonrelevant item ranked above it. Let  $\ell$  be the recall level of the first relevant item below position  $k$  (i.e.,  $\ell = \min\{i : p_i > k\}$ ).

$$\Delta_{p_j, k} \mu(\pi, \mathcal{R}) = z(\ell, m)(e(k) - e(p_\ell)) + \sum_{i=\ell}^{j-1} z(i+1, m)(e(p_i) - e(p_{i+1})).$$

Since  $z(i, m) \geq 0$  and  $e(p'_i) > e(p_i)$  for  $i \in [\ell, j]$ , we know that this difference will always be non-negative. Moreover, if  $z(i, m)$  is strictly positive, then the difference will always be strictly positive.  $\square$

**THEOREM B.5.** If  $z(i, m)$  strictly positive and the second derivative of  $e$  is (strictly) positive, then  $\mu$  is (strictly) concave in contiguous swap depth.

**PROOF.** Let  $\pi, \pi' \in S_n$  be two rankings whose relevant position vectors differ in one element  $j$  (i.e.,  $\forall i \neq j, p_i = p'_i$ ) and  $p_j < p'_j$ . The metric difference for moving the  $j$ th relevant item up one positions in  $\pi$  is

$$\Delta_{p_j, p_j-1} \mu(\pi, \mathcal{R}) = z(j, m)(e(p_j - 1) - e(p_j)).$$

If the second derivative of  $e$  is positive, since  $p_j < p'_j$ , then

$$(e(p_j - 1) - e(p_j)) \geq (e(p'_j - 1) - e(p'_j)).$$

Moreover, because  $z(i, m) > 0$ ,

$$\begin{aligned} z(j, m)(e(p_j - 1) - e(p_j)) &\geq z(j, m)(e(p'_j - 1) - e(p'_j)) \\ \Delta_{p_j, p_j-1} \mu(\pi, \mathcal{R}) &\geq \Delta_{p'_j, p'_j-1} \mu(\pi, \mathcal{R}), \end{aligned}$$

where the inequality is strict if the second derivative of  $e$  is strictly positive.  $\square$

**THEOREM B.6.** If  $z(i, m)$  is defined for  $m = 0$ , then  $\mu$  is complete.

**PROOF.** The only factor in Equation (1) that depends on  $m$  is  $z$ . If it is defined for  $m = 0$ , then  $\mu$  is defined as well.  $\square$

We note that, of the remaining properties, although we cannot prove every top-heavy recall-level metric will satisfy them, there are top-heavy recall-level metrics that do.

## B.2 Properties of Total Search Efficiency

Let  $\mu$  be TSE with an arbitrary exposure function  $e$  and normalization function  $z$ . Because TSE is a top-heavy recall-level metric, we know that it satisfies all of the properties in Section B.1 *except* those conditional on  $z(i, m) > 0$ , since  $z_{SL_3}(i, m) = 0$  when  $i < m$ .

## B.3 Properties of Lexirecall

**THEOREM B.7.** *lexirecall is monotonically increasing in retrieval size.*

**PROOF.** Let  $\tilde{\pi}'$  be  $\tilde{\pi}$  with another item appended, with pessimistically imputed rankings  $\pi'$  and  $\pi$ . If the new item is nonrelevant, then  $\forall i, p_i = p'_i$  and, therefore,  $\mu(\pi, \mathcal{R}) = \mu(\pi', \mathcal{R})$  and lexirecall is trivially non-decreasing. Next, consider the case where the new item is relevant. This can only happen if  $\tilde{\pi}$  includes  $\tilde{m} < m$  relevant items. As such, this is equivalent to swapping a relevant item in the imputed ranking  $\pi$  from position  $n - (m - \tilde{m} - 1)$  to position  $\tilde{n} + 1$ . Because of pessimistic imputation, the bottom  $m - \tilde{m} - 1$  relevant items will be tied. However,  $p'_{m+1} = \tilde{n} + 1$  and  $p_{m+1} = n - (m - \tilde{m} - 1)$ . If  $m - \tilde{m} < n - \tilde{n}$ , then lexirecall will be positive.  $\square$

**THEOREM B.8.** *lexirecall is monotonically decreasing in nonrelevance.*

**PROOF.** Let  $\tilde{\pi}'$  be  $\tilde{\pi}$  with a nonrelevant item appended. Since the new item is nonrelevant,  $p$  will be unchanged and lexirecall will be the same and is trivially non-increasing.  $\square$

**THEOREM B.9.** *lexirecall is strictly monotonically increasing in relevance.*

**PROOF.** See the second case in the proof Theorem B.7.  $\square$

**THEOREM B.10.**  *$\mu$  is monotonically increasing in swapping up.*

**PROOF.** Given a ranking  $\pi \in S_n$ , let  $j$  be the  $j$ th relevant item and  $k < p_j$  the position of an arbitrary nonrelevant item ranked above it. Since  $\forall i > j, p_i = p'_i$ , we only need to compare  $p_j$  and  $p'_j$ . If  $k > p_{j-1}$ , then, because  $k < p_j$ ,  $\pi' > \pi$ . If  $k < p_{j-1}$ , then  $p'_j = p_{j-1}$ . Because  $p_{j-1} < p_j$ ,  $\pi' > \pi$ .  $\square$

The properties concavity, boundedness, localization, completeness, and realizability are specific to metric-based evaluation and therefore cannot be analyzed for preference-based evaluation like lexirecall.

## C Robustness

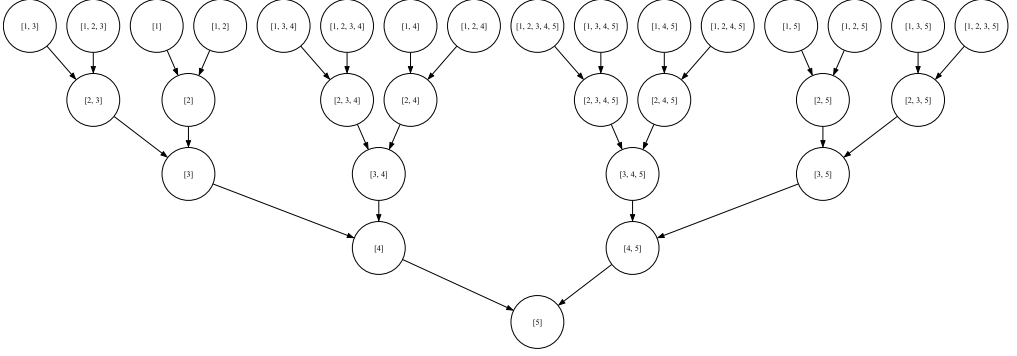
To demonstrate the relationship between the order of relevant items  $p$  and the order of either  $\mathcal{U}$  or  $\mathcal{V}$ , we first introduce a representation of subsets of positions of relevant items. Let  $\mathcal{W} = [1 \dots m]^+$  be the set of all non-empty sorted lists of integers between 1 and  $m$ . Moreover, let  $\mathcal{W}_{>i} = [i + 1 \dots m]^+$  and  $\mathcal{W}_{\geq i} = [i \dots m]^+$ . This is a way to represent each individual  $u \in \mathcal{U}$ , for example, in Figure 6. To see how, notice that, because both  $\mathcal{U}$  and  $\mathcal{V}$  are also power sets of  $m$  distinct integers, there is a one to one correspondence with  $\mathcal{W}$ . Specifically,

$$\begin{aligned} \forall w \in \mathcal{W}, u &= \{i \in w \mid \pi_{p_i}\} \\ \forall u \in \mathcal{U}, w &= \text{sort}(\{\pi_{p_j} \in u \mid j\}) \end{aligned}$$

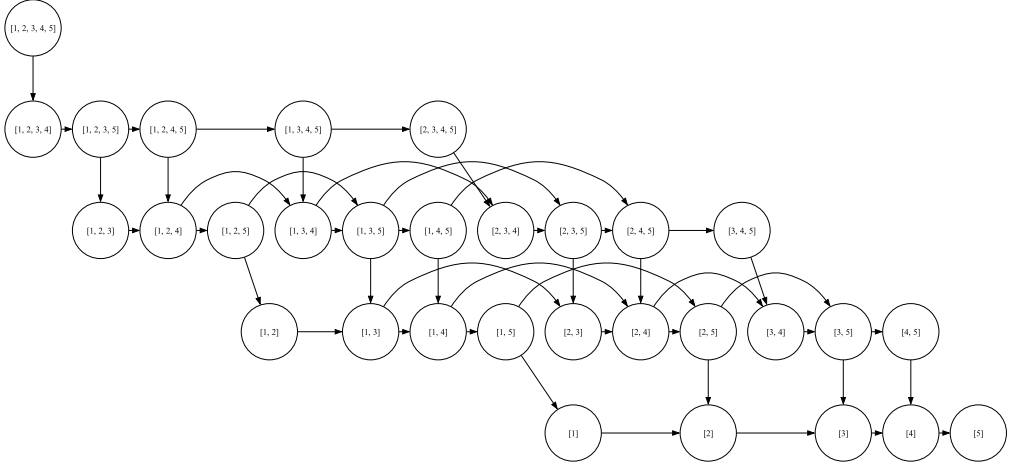
and similarly for  $\mathcal{V}$ .

Given a way to represent each  $u \in \mathcal{U}$ , we need to sort these users according to their utility. We can use our metric definitions  $\mu(\pi, u)$  and  $\eta(\pi, \mathcal{R}, v)$  to define a partial ordering over  $\mathcal{W}$ . This naturally can be represented as a graph where edges reflect that the utility of one user is greater than another. We present an example based on  $m = 5$  of the transitive reduction of the partially ordered





(a) Users



(b) Providers

Fig. 17. Transitive reduction of the partially ordered set  $\mathcal{W}$  and  $m = 5$ . Nodes correspond to elements in  $\mathcal{W}$ . A directed edge from  $w$  to  $w'$  if  $\mu(\pi, b) > \mu(\pi, b')$  (Figure 17(a)) or  $\eta(\pi, \mathcal{R}, b) > \eta(\pi, \mathcal{R}, b')$  (Figure 17(b)) based on the properties of top-heavy recall-level metrics (Definition 2.2).

set for both  $\mathcal{U}$  and  $\mathcal{V}$  in Figure 17. Although we will present formal proofs, these visualizations help understand the utility structure behind these sets of users.

### C.1 Worst-case and Total Search Efficiency

Notice that, in our example, there is a single, unique minimal element for both  $\mathcal{U}$  and  $\mathcal{V}$ . In this section, we will prove that this element will always be associated with the lowest-ranked relevant item,  $\text{TSE}(\pi, \mathcal{R})$ . Throughout these proofs, we will use abbreviations for top-heavy recall-level metric properties defined in Section 2.

**THEOREM C.1.** *If  $\mu$  is a top-heavy recall-level metric and  $z(1, 1) = 1$ , then*

$$\text{WC}_\mu(\pi, \mathcal{R}) = \text{TSE}(\pi, \mathcal{R}).$$

PROOF We want to show that, for any  $u \in \mathcal{U}$ , the performance  $\mu(\pi, u)$  is greater than or equal to  $\text{TSE}(\pi, \mathcal{R})$ , with equality for the user associated with the lowest-ranked relevant item. Recall that each  $w \in \mathcal{W}$  is associated with a  $u \in \mathcal{U}$ ,

$$\begin{aligned} \mu(\pi, w) &\geq e(p_{w_{-1}})z(1, 1) && \text{top-heaviness} \\ &= e(p_{w_{-1}}) && z(1, 1) = 1 \\ &\geq e(p_m) && p_{w_{-1}} \leq p_m \text{ and } e(i) > e(i'), \forall i < i' \\ &= \text{TSE}(\pi, \mathcal{R}). \end{aligned}$$

□

THEOREM C.2. If  $\eta$  is associated with a top-heavy recall-level metric  $\mu$ , then

$$\text{WC}_\eta(\pi, \mathcal{R}) = \text{TSE}(\pi, \mathcal{R}).$$

PROOF. Because all summands of Equation (2) are positive, we know that the minimum  $v \in \mathcal{V}$  will correspond to the smallest summand. Moreover, because of the monotonically decreasing exposure, this is the exposure of the lowest ranked relevant item, which is exactly  $\text{TSE}(\pi, \mathcal{R})$ . □

## C.2 Leximin and Lexicographic Recall

Let  $q_i = e(p_i)$ . Since  $p$  is sorted in increasing order and  $e(x)$  is monotonically decreasing in  $x$ ,  $q$  is monotonically decreasing. Assuming we measure the performance of a ranking  $\pi$  for a user  $u$  as  $\mu(\pi, u)$ , then we define  $\varrho$  to be the  $m^+ \times 1$  vector containing the value of  $\mu(\pi, u)$  for each  $u \in \mathcal{U}$ . We will assume that  $\varrho$ , like  $q$ , is sorted in decreasing order.

LEMMA C.3. If  $q >_{\text{leximin}} q'$  and  $z(1, 1) = 1$ , then the minimum non-tied user is in  $\mathcal{W}_{\geq k} - \mathcal{W}_{> k}$ , where  $k = \min\{i \in [1 \dots m] | q_i \neq q'_i\}$ .

PROOF Let  $k = \min\{i \in [1 \dots m] | q_i \neq q'_i\}$ . Because each  $w \in \mathcal{W}_{> k}$  is comprised of indices  $i > k$  and because  $\forall i > k, p_i = p'_i$ , for each associated user  $u$ ,  $\mu(p, u) = \mu(p', u)$ . When computing leximin, by the axiom of the Independence of Identical Consequences [2], we can remove all of the elements from  $\varrho$  and  $\varrho'$  associated with the  $\mathcal{W}_{> k}$ . This means that the minimum non-tied pair will be in  $\mathcal{W} - \mathcal{W}_{> k}$ .

Now we need to show that  $\mathcal{W}_{\geq k} - \mathcal{W}_{> k}$  is the set of minimal elements of  $\mathcal{W} - \mathcal{W}_{> k}$ . This holds if we can show that for every element  $w \in \mathcal{W} - \mathcal{W}_{\geq k}$ , there exists  $w' \in \mathcal{W}_{\geq k} - \mathcal{W}_{> k}$ , such that  $\mu(\pi, w) \geq \mu(\pi, w')$ .

If  $w_{-1} \leq k$  (as depicted by the pink nodes Figure 17), then let  $w' = \{k\}$ ,

$$\begin{aligned} \mu(\pi, w) &\geq e(p_{w_{-1}})z(1, 1) && \text{top-heaviness} \\ &= e(p_{w_{-1}}) && z(1, 1) = 1 \\ &\geq e(p_k) && p_{w_{-1}} \leq p_k \text{ and } e(i) > e(i'), \forall i < i' \\ &= \mu(\pi, w'). \end{aligned}$$

If  $w_{-1} > k$  and  $k \in w$  (as depicted by the blue nodes Figure 17), then we can let  $w' = \{j \in w : j \geq k\}$ .

Let  $j = |w| - |w'|$ ,

$$\begin{aligned} \mu(\pi, w) &\geq \sum_{i=j+1}^{|w|} e(p_{w_i})z(i - j, |w| - j) && \text{top-heaviness} \\ &= \sum_{i=1}^{|w'|} e(p_{w'_i})z(i, |w'|) && \text{definition of } w' \\ &= \mu(\pi, w'). \end{aligned}$$

If  $w_{-1} > k$  and  $k \notin w$  (as depicted by the green nodes Figure 17), then  $w' = \{k\} \cup \{j \in w : j > k\}$ . Let  $j = |w| - |w'|$ ,

$$\begin{aligned}
 \mu(\pi, w) &\geq \sum_{i=j+1}^{|w|} e(p_{w_i})z(i-j, |w|-j) && \text{top-heaviness} \\
 &= e(p_{w_{j+1}})z(1, |w|-j) + \sum_{i=j+2}^{|w|} e(p_{w_i})z(i-j, |w|-j) \\
 &= e(p_{w_{j+1}})z(1, |w'|) + \sum_{i=2}^{|w'|} e(p_{w'_i})z(i, |w'|) && \text{definition of } j \text{ and } w' \\
 &> e(p_{w'_1})z(1, |w'|) + \sum_{i=2}^{|w'|} e(p_{w'_i})z(i, |w'|) && p_{w_{j+1}} < p_k \text{ and } e(i) > e(i'), \forall i < i' \\
 &= \sum_{i=1}^{|w'|} e(p_{w'_i})z(i, |w'|) \\
 &= \mu(\pi, w').
 \end{aligned}$$

□

THEOREM C.4.

$$q \succ_{\text{leximin}} q' \rightarrow \varrho \succ_{\text{leximin}} \varrho'.$$

PROOF

*Case 1* ( $q_m > q'_m$ ). Theorem C.1 means that  $\varrho_{m^+} = q_m$  and  $\varrho'_{m^+} = q'_m$  and the implication is true.

*Case 2* ( $q_m = q'_m$ ). Because  $q \succ_{\text{leximin}} q'$ , we know that there exists a  $k$  such that  $q_k > q'_k$  and  $\forall i \in [k+1 \dots m], q_i = q'_i$  (as depicted in Figure 19). We know from Lemma C.3 that the lowest-ranked, non-tied users are in  $\mathcal{W}_{\geq k} - \mathcal{W}_{> k}$ .

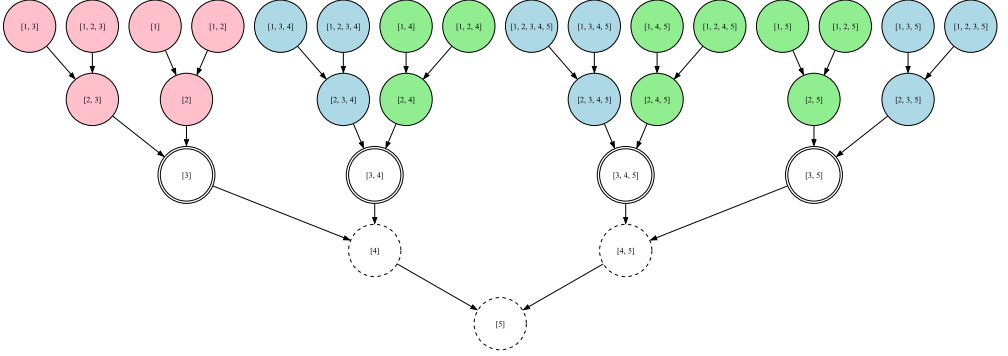
We want to show that, for every element  $w \in \mathcal{W}_{\geq k} - \mathcal{W}_{> k}$ ,  $\pi$  is preferred to  $\pi'$ ,

$$\begin{aligned}
 \mu(\pi, w) &= \sum_{i=1}^{|w|} e(p_{w_i})z(i, |w|) \\
 &= e(p_k)z(1, |w|) + \sum_{i=2}^{|w|} e(p_{w_i})z(i, |w|) \\
 &= e(p_k)z(1, |w|) + \sum_{i=2}^{|w|} e(p'_{w_i})z(i, |w|) && \forall i > k, p_i = p'_i \\
 &> e(p'_k)z(1, |w|) + \sum_{i=2}^{|w|} e(p'_{w_i})z(i, |w|) && p_k < p'_k \text{ and } e(i) > e(i'), \forall i < i' \\
 &= \sum_{i=1}^{|w|} e(p'_{w_i})z(i, |w|) \\
 &= \mu(\pi', w).
 \end{aligned}$$

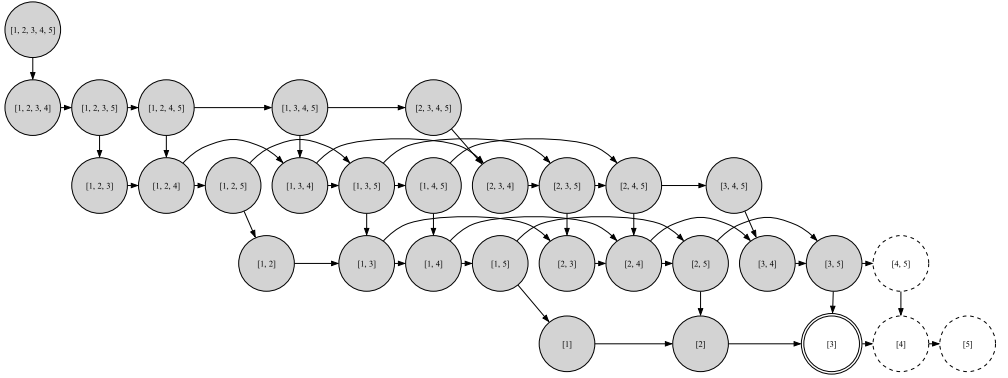
□

THEOREM C.5.

$$q =_{\text{leximin}} q' \rightarrow \varrho =_{\text{leximin}} \varrho'.$$



(a) Users. Partitioning  $\mathcal{W}$  when  $k = 3$ .  $\mathcal{W}$  is partitioned into  $\mathcal{W}_{\geq k}$  (double circle),  $\mathcal{W}_{> k}$  (dashed circle), and  $\mathcal{W} - \mathcal{W}_{\geq k}$  (colored circles, described in the proof of Lemma C.3). This figure best rendered in color.



(b) Providers. Assuming  $k = 3$ , then the provider with the minimum exposure is  $w = [k]$ .

Fig. 18. Leximin.

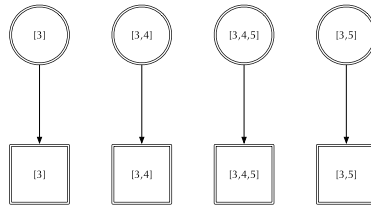


Fig. 19. Domination of  $\mathcal{W}_{\geq k} - \mathcal{W}_{> k}$ . Let  $\pi$  be represented by circles and  $\pi'$  be represented by squares.

PROOF. The only way that  $q =_{\text{leximin}} q'$  is when  $\forall i \in [1 \dots m], q_i = q'_i$ . If this is the case, then  $\forall u \in \mathcal{U}, \mu(\pi, u) = \mu(\pi', u)$  and, therefore,  $\varrho =_{\text{leximin}} \varrho'$ .  $\square$

THEOREM C.6.

$$q >_{\text{leximin}} q' \rightarrow \phi >_{\text{leximin}} \phi',$$

where, for  $v \in \mathcal{V}$ ,  $\phi_v = \eta(\pi, \mathcal{R}, v)$ .

PROOF.

*Case 1* ( $q_m > q'_m$ ). We know from the proof of Theorem C.2, the lowest ranked items in  $\phi$  and  $\phi'$  correspond to  $e(p_m) = q_m$  and  $e(p'_m) = q'_m$ , respectively, and so the proof holds in this case.

*Case 2* ( $q_m = q'_m$ ). Because  $q \succ_{\text{leximin}} q'$ , we know that there exists a  $k$  such that  $q_k > q'_k$  and  $\forall i \in [k+1, m]$ ,  $q_i = q'_i$ . The provider  $w = [k]$  must be the worst-off non-tied element. Assume that there exists a worse-off non-tied provider. Because  $w$  is a singleton and because of monotonically decreasing exposure, this worse off provider must be in  $\mathcal{W}_{>k}$ . However, we know that these providers are all tied and therefore we have a contradiction.  $\square$

## D Optimal Rankings

Let  $S_n^*$  be the set of permutations that rank all of the items in  $\mathcal{R}$  above  $\mathcal{D} - \mathcal{R}$ .

THEOREM D.1. Given  $\pi^* \in S_n^*$ ,

$$\min_{u \in \mathcal{U}} \mathbb{E}_{\pi \sim S_n^*} [\mu(\pi, u)] \geq \min_{u \in \mathcal{U}} \mu(\pi^*, u).$$

PROOF. Let  $\tilde{u} = \operatorname{argmin}_{u \in \mathcal{U}} \mathbb{E}_{\pi \sim S_n^*} [\mu(\pi, u)]$  be the worst-off user for a stochastic ranking and  $\check{\pi} = \operatorname{argmin}_{\pi \sim S_n^*} \mu(\pi, \tilde{u})$  the worst deterministic ranking in  $S_n^*$  for  $\tilde{u}$ ,

$$\begin{aligned} \mathbb{E}_{\pi \sim S_n^*} [\mu(\pi, \tilde{u})] &= \frac{1}{C} \sum_{\pi \sim S_n^*} \mu(\pi, \tilde{u}) \\ &\geq \mu(\check{\pi}, \tilde{u}) \\ &\geq \min_{u \in \mathcal{U}} \mu(\check{\pi}, u) \\ &= \min_{u \in \mathcal{U}} \mu(\pi^*, u), \end{aligned}$$

where the final equality follows because  $\check{\pi}, \pi^* \in S_n^*$  and, therefore, have isometric distributions of user utility.  $\square$

## E Number of Ties

THEOREM E.1.

$$\Pr(\pi =_{\text{LR}} \pi') = \frac{m!(n-m)!}{n!}.$$

PROOF. If we sample a ranking uniformly from  $S_n$ , then the probability of any specific  $p$  is

$$\begin{aligned} \Pr(p) &= \frac{C}{|S_n|} \\ &= \binom{n}{m}^{-1}. \end{aligned}$$

Let  $\mathcal{P}_m^n$  be the set of all size  $m$  samples of unique integers from  $[1 \dots n]$ ,

$$\begin{aligned}
 \Pr(\pi =_{\text{LR}} \pi') &= \Pr(p = p') \\
 &= \sum_{p, p' \in \mathcal{P}_m^n} \Pr(p) \Pr(p') \mathbb{I}(p = p') \\
 &= \sum_{p \in \mathcal{P}_m^n} \Pr(p)^2 \\
 &= \binom{n}{m} \times \frac{1}{\binom{n}{m}} \times \frac{1}{\binom{n}{m}} \\
 &= \frac{m!(n-m)!}{n!}.
 \end{aligned}
 \quad \square$$

THEOREM E.2.

$$\Pr(\pi =_{\text{TSE}} \pi') = \binom{n}{m}^{-2} \sum_{i=m}^n \binom{i-1}{m-1}^2.$$

PROOF. First, we will compute the probability of ranking  $\pi$  where the position of the last relevant item is  $i$ ,

$$\Pr(p_m = i) = \frac{\binom{i-1}{m-1}}{\binom{n}{m}}.$$

We can use this to compute the probability of a tie,

$$\begin{aligned}
 \Pr(\pi =_{\text{TSE}} \pi') &= \Pr(p_m = p'_m) \\
 &= \sum_{p, p' \in \mathcal{P}_m^n} \Pr(p) \Pr(p') \mathbb{I}(p_m = p'_m) \\
 &= \sum_{i=m}^n \Pr(p_m = i)^2 \\
 &= \sum_{i=m}^n \frac{\binom{i-1}{m-1}^2}{\binom{n}{m}^2}.
 \end{aligned}
 \quad \square$$

THEOREM E.3.

$$\Pr(\pi =_{R_k} \pi') = \binom{n}{m}^{-2} \sum_{i=0}^m \binom{k}{i}^2 \binom{n-k}{m-i}^2.$$

PROOF. Let  $\text{Rel}(p, k) = \sum_{i=1}^m \mathbb{I}(p_i \leq k)$ . First, we will compute the probability of ranking  $\pi$  where  $i$  items are ranked above position  $k$ ,

$$\Pr(\text{Rel}(p, k) = i) = \frac{\binom{k}{i} \binom{n-k}{m-i}}{\binom{n}{m}}.$$

We can use this to compute the probability of a tie,

$$\begin{aligned}
 \Pr(\pi =_{\text{TSE}} \pi') &= \Pr(\text{Rel}(p, k) = \text{Rel}(p', k)) \\
 &= \sum_{p, p' \in \mathcal{P}_m^n} \Pr(p) \Pr(p') \mathbb{I}(\text{Rel}(p, k) = \text{Rel}(p', k)) \\
 &= \sum_{i=0}^m \Pr(\text{Rel}(p, k) = i)^2 \\
 &= \sum_{i=0}^m \frac{\binom{k}{i}^2 \binom{n-k}{m-i}^2}{\binom{n}{m}^2}.
 \end{aligned}$$

□

THEOREM E.4.

$$\Pr(\pi =_{\text{RP}} \pi') = \binom{n}{m}^{-2} \sum_{i=0}^m \binom{m}{i}^2 \binom{n-m}{m-i}^2.$$

PROOF. The proof follows that of Theorem E.3, substituting  $m$  for  $k$ .

□

Received 7 March 2024; revised 30 November 2024; accepted 4 February 2025