Evaluation of IR Applications with Constrained Real Estate

Yuanhua Lv, Ariel Fuxman, and Ashok K. Chandra

Microsoft Research,
Mountain View, CA USA, 94043
{yuanhual, arielf, achandra}@microsoft.com

Abstract. Traditional IR applications assume that there is always enough space ("real estate") available to display as many results as the system returns. Consequently, traditional evaluation metrics were typically designed to take a length cutoff k of the result list as a parameter. For example, one computes DCG@k, Prec@k, etc., based on the top-k results in the ranking list. However, there are important modern ranking applications where the result real estate is constrained to a small fixed space, such as the search verticals aggregated in the Web search results and the recommendation systems. For such applications, the following tradeoff arises: given a fixed amount of real estate, shall we show a small number of results with rich captions and details, or a larger number of results with less informative captions? In other words, there is a tradeoff between the length of the result list (i.e., quantity) and the informativeness of the results (i.e., quality). This tradeoff has important implications for evaluation metrics, since it leads the length cutoff k hard to be determined a priori. In order to tackle this problem, we propose two desirable formal constraints to capture the heuristics of regulating the quantity-quality tradeoff, inspired by the axiomatic approach to IR. We then present a general method to normalize the well-known Discounted Cumulative Gain (DCG) metric for balancing the quantity-quality tradeoff, yielding a new metric, that we call Length-adjusted Discounted Cumulative Gain (LDCG). LDCG is shown to be able to automatically balance the length and the informativeness of a ranking list without requiring an explicit parameter k, while still preserving the good properties of DCG.

Keywords: Evaluation, Aggregated Search, Constrained Real Estate, Quantity-Quality Tradeoff, LDCG, LNDCG

1 Introduction

Evaluation metrics play a critical role in the field of information retrieval (IR). Traditional IR applications assume that there is always enough space ("real estate") available to display as many results as the system returns. To evaluate such systems, traditional evaluation metrics were typically designed to take a length cutoff k of the result list as a parameter. For example, one computes DCG@k [8], Prec@k, etc., based on the top-k results in the ranking list.

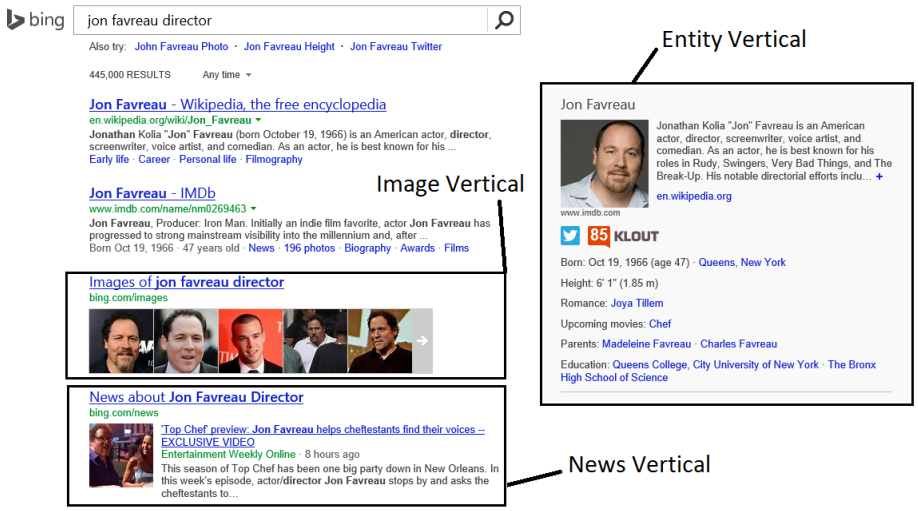

Fig. 1. Search verticals as examples of applications with constrained real estate.

However, there are important modern ranking applications where the real estate available for the results is constrained to a small fixed space. For example, search engines are no longer restricted to the classical ten blue links for Web results, and they now aggregate all kinds of information (shopping, weather, news, images, entities from a knowledge repository, etc.) on the search engine result page. As an illustration, consider the snapshot in Figure 1 for the query "jon favreau director". In addition to the Web links, we can see results corresponding to images (Image Vertical), news (News Vertical), and an entity card with rich information about the famous American director Jon Favreau (Entity Vertical). Each vertical has limited real estate; for instance, the Entity Vertical results are restricted to a fixed area on the right hand side of the screen.

For each such applications, the following tradeoff arises: given the constrained amount of real estate, shall we show a small number of results with rich captions and details, or a larger number of results with less informative captions? Continuing our example of "jon favreau director", if the search engine is confident enough about the person Jon Favreau being the intended entity result, it can show a single rich entity card for him, as in Figure 1; if the search engine is not confident enough, it may instead show two more impoverished candidates: the entity for Jon Favreau and the entity for "Iron Man 3" (a film directed by Jon Favreau), as shown in Figure 2. Then, how to evaluate which one is better?

On the one hand, the evaluation of these applications does share some requirements with Web search. First, ranking order matters. In the example of query "jon favreau director", we do not want to rank "Iron Man 3" higher than the director himself. Second, we need to deal with graded relevance. In our example, the director Jon Favreau should be a "perfect" answer, and "Iron Man 3" should still be a good answer because Jon Favreau is its director. But the entity

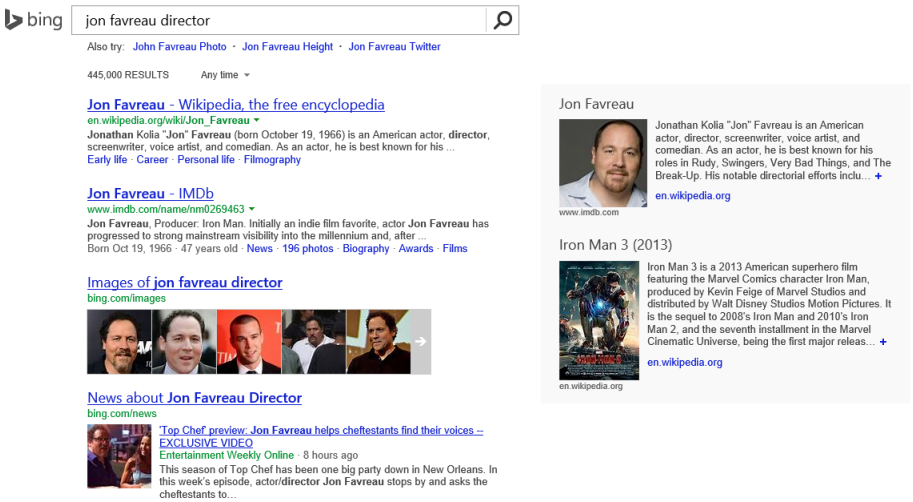

Fig. 2. Alternative entity result for "jon favreau director".

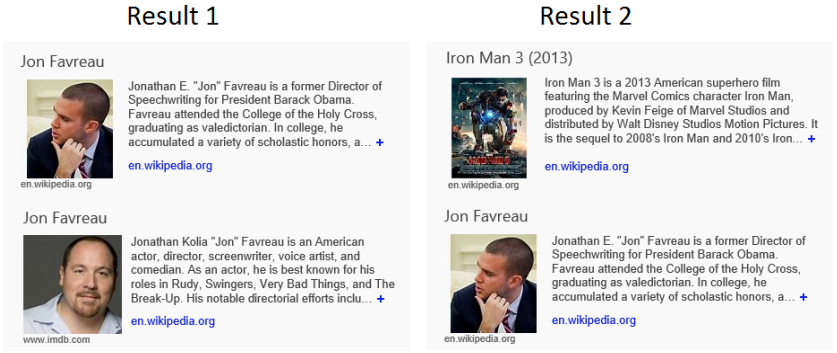

Fig. 3. Comparison of entity results for "jon favreau director".

for another person who also happens to be called Jon Favreau (i.e., Obama's speechwriter) is clearly a bad answer. This would suggest the use of DCG [8] or other existing IR metrics (e.g., [4, 10]) as an evaluation metric.

On the other hand, the real estate constraints have important implications for evaluation metrics. As discussed above, we need to decide carefully if we want to show a small number of informative answers with more details (if we are confident); or give a few candidates with fewer details for each (otherwise). Both options have their pros and cons: showing a more informative result would surely delight the users if the result is correct, but it would upset them if it is incorrect; showing more results would be safer, but the more impoverished descriptions would not please users as much as the former. Therefore, to best use the constrained real estate, we need to adaptively choose an option to optimize the tradeoff between the length of the result list (quantity) and the informa-

Entity	Relevance
Jon Favreau (director)	perfect
Iron Man 3	good
Jon Favreau (speechwriter)	bad

Table 1. All relevance judgments for query "jon favreau director" (faked).

tiveness of the results (quality) for each individual query. This poses significant challenges to traditional IR evaluation metrics. More specifically:

- No appropriate value for the length cutoff k can be easily determined a priori because the length of such a ranking list varies significantly in different alternative outputs of the ranking algorithm. In other words, a fixed cutoff parameter k could lead to poor evaluation results. Consider the examples of Figure 3. For the same query, "jon favreau director", suppose Table 1 contains all the relevance judgments. When we compute DCG@1 or NDCG@1, the result list 2 would be preferred, but when we compute DCG@2 or NDCG@2, the result list 1 would be preferred. That is, DCG@k and NDCG@k lead to inconsistent decisions depending on the value of k that we choose. Other traditional metrics share similar problems.
- The "additive" nature of DCG/NDCG and other IR metrics is not suitable in this application. To see why, compare Figure 1 and 2 again. For query "jon favreau director", the entity vertical result of Figure 1 (which shows exclusively the film director) seems more desirable than the entity vertical results of Figure 2 that show the director in the first position and the film in the second position. However, taking DCG/NDCG as an example, although both results have the same DCG@1 and NDCG@1 scores, the entity result in Figure 2 has higher DCG@2 and NDCG@2 scores than that of Figure 1.

To address these problems, we first propose two desirable formal constraints to capture the heuristics of regulating the quantity-quality tradeoff to properly evaluate IR applications with constrained result real estate. Inspired by the axiomatic approach to IR [7], we then present a general method to normalize the popular DCG metric [8] for balancing the quantity-quality tradeoff, yielding a new metric, namely Length-adjusted Discounted Cumulative Gain (LDCG). LDCG is shown to be able to automatically balance the length and the informativeness of a ranking list without requiring an explicit parameter k, while still preserving the good properties of DCG.

2 Related Work

The evaluation of IR applications with constrained real estate is a novel problem, and to the best of our knowledge, no previous work has addressed it to date. We briefly discuss some research efforts in the general IR evaluation literature that connect to our work.

Evaluation metrics play a critical role in the field of IR, and various metrics have been proposed. For example, average precision (AP) has been used extensively in TREC and other IR literature, and an extension GAP has recently

proposed to extend AP to incorporate multi-graded relevance [10]; DCG and NDCG [8] have been accepted as major metrics for Web search [2]; and Expected Reciprocal Rank (ERR), an extension of the classical reciprocal rank to the graded relevance, has recently attracted much attention in Web search [4].

Since the metrics that we propose in this paper are extensions of DCG and NDCG, we now provide a brief overview of them. DCG at rank k for a ranking list is computed as [2]:

$$DCG@k = \sum_{i=1}^{k} \frac{2^{r(i)} - 1}{\log_2^{(i+1)}} \quad (1)$$

where $r(i)$ is the relevance level of the result at rank i. However, DCG is often incomparable, and should be normalized. This can be done by sorting the relevance judgments of a query by relevance and producing the maximum possible DCG up to position k, called the ideal DCG (IDCG) at position k, as the normalization factor. The normalized DCG (NDCG) is computed as: $NDCG@k = \frac{DCG@k}{IDCG@k}$.

We can see that an explicit cutoff parameter k is required to compute DCG and NDCG. In fact, a similar cutoff parameter is also required by many other metrics, for example, average precision, precision, recall, etc. The purpose of this work is to eliminate this requirement by modeling the cutoff parameter in a "soft" way inside the metric.

One of the IR applications with constrained real estate is the search verticals in aggregated search. The evaluation of aggregated search has recently attracted attention, e.g., [1, 11, 12, 5]. However, existing work does not address the issue of constrained result real estate and the quantity-quality tradeoff. Also the proposed metrics in our paper are more general and are not restricted to vertical search: we can potentially apply the proposed metrics to many IR applications with constrained real estate, e.g., recommendation systems.

Our work is also related to the literature on user models. An accurate user model is essential for developing a good relevance metric. In general, there are two main types of user models: position models (e.g., [6]) and cascade models (e.g., [4]). Both types of models attempt to capture the position bias of search result presentation. In contrast, our proposed length-variant user expectation is designed to approximate a threshold value for an IR metric according to the length of the result list.

The axiomatic approach has been used to develop effective IR models [7, 9]. Our work adopts similar ideas to formalize the requirement of an IR evaluation scenario as formal constraints. Interestingly, a recent work [3] also uses the axiomatic approach to study IR metrics, but their work falls short when dealing with applications with constrained real estate.

3 Formal Constraints on Regulating the Quantity-Quality Tradeoff

A critical question is the following: how we can regulate the interactions between the quantity (i.e., the length of result list) and the quality (i.e., the informativeness of each result) so that we can balance the two factors? To answer this

question, we first propose two desirable heuristics that a reasonable evaluation metric should implement to properly evaluate IR applications with constrained result real estate:

- First, *showing fewer results of higher relevance with more details should be preferred over showing more results, some of similar relevance and some others of lower relevance, with fewer details for each.* For example, the entity result in Figure 1 should be preferred over the entity result in Figure 2. This heuristic is used to reward the quality (informativeness) of the results. In traditional IR systems, each search result is usually displayed in a fixed space of (almost) the same size, and thus the informativeness of every result is also similar. In contrast, in the applications with constrained real estate, each result is allowed to show more or fewer details dynamically, so a relevant result that is more detailed/informative should be rewarded. To further understand this heuristic, we conducted a user study via crowdsourcing to ask the crowd to do a side-by-side comparison of a set of 174 pairs of entity search ranking lists (all with the same constrained real estate). A preliminary analysis of the user preference suggests that 82.4% people agree that one single "perfect" result is better than one "perfect" result followed by another "good" result, and that 75.8% people agree that one single "good" result is better than one "good" result followed by another "bad" result. This verifies that the proposed heuristic is desirable.
- Second, *showing more relevant results should be preferred over showing fewer results of similar relevance.* This heuristic is actually not entirely novel; it is easy to show that many current IR metrics have already implemented it. However, we still emphasize it explicitly because (1) it is widely-accepted and implemented in existing IR metrics as shown later, (2) it can be used to prevent the first heuristic from overly-penalizing the length of result list, and (3) it presumably makes sense in our applications since the former covers more diverse information with no relevance degradation.

Next, in order to analytically diagnose the limitations of current IR metrics, we propose two formal constraints to capture the above two heuristics of regulating the quantity-quality tradeoff so that it is possible to apply them to any IR metrics analytically.

We first define some key notations. Let L_1 and L_2 be two ranking lists. Assume $|L_1| = |L_2|$, where $|L_1|$ and $|L_2|$ are the sizes of the space taken to show L_1 and L_2 respectively, i.e., the real estate. And assume $L_1 =< d_1 >$, and $L_2 =< d_2, d_3 >$, where d_1, d_2, and d_3 are three results with relevance degrees $R(d_1)$, $R(d_2)$, and $R(d_3)$, respectively. We denote $E(\cdot)$ as a reasonable evaluation metric. Then the two constraints are defined as follows:

- **C1:** If $R(d_1) = R(d_2) > R(d_3)$, then $E(L_1) > E(L_2)$.
- **C2:** If $R(d_1) = R(d_2) = R(d_3) > 0$ [1], then $E(L_1) < E(L_2)$.

[1] We assume the relevance degree to be 0 for "bad" result.

	DCG	NDCG	GAP	ERR	LDCG	LNDCG
C1	No	No	No	No	Yes	Yes
C2	Cond	Cond	Cond	Cond	Yes	Yes

Table 2. Constraint analysis results for different IR metrics.

The proposed two constraints are useful to ensure the quantity-quality tradeoff. When either one is violated, the metric would likely not perform fairly for the evaluation of IR applications with constrained real estate $|L|$, and there should be room to improve the metric through improving its ability of satisfying the corresponding constraint. We analyze some typical traditional IR metrics that support graded relevance, including DCG/NDCG [8], ERR [4], and GAP [10].

- Under the condition in C1: Due to the additive nature of DCG, ERR, and GAP, it is easy to show that, DCG@$k(L1)$ <= DCG@$k(L2)$, NDCG@$k(L1)$ <= NDCG@$k(L2)$, GAP@$k(L1)$ <= GAP@$k(L2)$, and ERR@$k(L1)$ <= ERR@$k(L2)$. This shows that none of these existing evaluation metrics can satisfy C1, no matter what value the length cutoff parameter k is.
- Under the condition in C2: Similarly due to their additive nature, when $k > 1$, we can also see that DCG@$k(L1)$ < DCG@$k(L2)$, NDCG@$k(L1)$ < NDCG@$k(L2)$, GAP@$k(L1)$ < GAP@$k(L2)$, and ERR@$k(L1)$ < ERR@$k(L2)$, which is consistent with C2. However, when $k = 1$, it is not surprising that DCG@$k(L1)$ = DCG@$k(L2)$, NDCG@$k(L1)$ = NDCG@$k(L2)$, GAP@$k(L1)$ = GAP@$k(L2)$, and ERR@$k(L1)$ = ERR@$k(L2)$, which is inconsistent with C2. These analysis results show that the existing evaluation metrics can only satisfy C2 conditionally when an appropriate length cutoff parameter k is chosen a prior which itself is nontrivial.

Due to space limitations, we cannot show all the analysis details in this paper, but the results are presented in Table 2.

4 A Quantity-Quality Balanced Metric

The analysis above shows that traditional IR metrics do not satisfy the proposed quantity-quality regulation constraints. In order to properly evaluate applications with constrained real estate, we need to add the quantity-quality tradeoff into the evaluation metric. However, we do not want the addition of this property to change other desirable properties (e.g., ranking order and multi-graded relevance awareness) possessed by these widely-accepted metrics.

We propose a general approach to achieve this goal by adopting the current additive metrics to measure the quality of a ranking list, while developing a length (quantity) normalization component to normalize the quality score. In this paper, we choose the popular DCG [8, 2] as the quality measure, and propose a novel method for length-normalization based on an intuition of length-variant user expectation.

4.1 Length-Variant User Expectation

User experience relies not only on the quality of the results themselves, but also on the a priori expectation that the users have on those results. For example, when user expectation is low, a ranking list could be satisfactory even if its quality is not so good; in contrast, when user expectation is high, a result list could still be unsatisfactory even if its quality is already very good.

In this section, we explore this notion, and approximate it as a threshold score for the quality of a ranking list: failing to achieve this threshold renders the results unsatisfactory to the user. This can also be regarded as some sort of a "lower-bound" on the user expectation.

Intuitively, user expectation depends on the length of a ranking list: users would generally have higher expectations when they are shown a longer list of results. That is, user expectation increases with the length of the result list. On the other hand, user expectation would become less sensitive to the length of the result list as the list becomes longer.

Based on this intuition, we approximate a length-variant user expectation for a ranking list. Specifically, we propose a model that relies on a single assumption: a necessary condition for a user to be satisfied with the results is that *there must be at least one relevant result appearing in the list, and that this relevant result can be ranked at any position according to some given position priors* (which may reflect the fact that the user would expect the relevant result to appear at a high position.)

4.2 Length Normalized DCG and NDCG

As we choose DCG as our quality measure, what would be the user expected DCG (i.e., the threshold DCG score), for a ranking list with N results? Based on the assumption articulated in the previous section, the threshold score can be met when there is at least one relevant result in the ranking list. When the relevant result occurs at position i, DCG can be calculated as:

$$DCG(i) = \frac{2^{rel} - 1}{\log_2^{(i+1)}} \tag{2}$$

where rel represents the relevance level of the relevant result. We can simply assume $rel = 1$, since it does not influence the relative score as we will show.

Let $p(i)$ be a function representing the "prior" belief of the user regarding what position holds the relevant result. For example, $p(1) = 0.5$ means that the user expects the relevant result to occur at the first position with a probability of 0.5. For the purposes of estimating the user's expected DCG, without loss of generality, we take a prior proportional to DCG's discounting factor when $i \leq M$, and take a prior of 0 when $i > M$, where M is the maximum number of results allowed to show in the result real estate. That is,

$$p(i) \propto \begin{cases} \frac{1}{\log_2^{(i+1)}} & \text{if } i \leq M \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

For a ranking list of length N, the user's expected DCG can be defined as the expected value of DCG(i) under $p(i)$:

$$E[DCG] = \sum_{i=1}^{N} p(i) \cdot DCG(i) = Z(M, rel) \cdot \sum_{i=1}^{N} \left(\frac{1}{\log_2^{(i+1)}} \right)^2 \tag{4}$$

where $Z(M, rel)$ is a normalization factor, which is a constant. More precisely,

$$Z(M, rel) = \frac{2^{rel} - 1}{\sum_{i=1}^{M} \frac{1}{\log_2^{(i+1)}}} = \frac{1}{\sum_{i=1}^{M} \frac{1}{\log_2^{(i+1)}}} \tag{5}$$

It is easy to see that $\frac{\partial E[DCG]}{\partial N} > 0$ and $\frac{\partial^2 E[DCG]}{\partial^2 N} < 0$, which shows that the user's expected DCG is monotonically increasing with N, but the increasing speed decreases with N. These characteristics are consistent with our intuitions that a user would expect more from a longer result list, and that if the result list is already long, the user's expectation would not be so sensitive to N.

In order to evaluate the user's satisfaction on a ranking list, we compare the standard DCG score against the user's expected DCG score. Specifically, we normalize the standard DCG by dividing it using the user's expected DCG, leading to a Length-adjusted DCG (LDCG). Formally,

$$LDCG = \frac{DCG}{E[DCG]} = \frac{DCG}{Z(M, rel) \cdot \sum_{i=1}^{N} \left(\frac{1}{\log_2^{(i+1)}} \right)^2} \propto \frac{DCG}{\sum_{i=1}^{N} \left(\frac{1}{\log_2^{(i+1)}} \right)^2} \tag{6}$$

which essentially normalizes DCG based on the square sum of the document discounting factors, but does not change other existing properties of DCG. One may notice that if we use a uniform probability for $p(i)$ in Formula 3, we will essentially use the sum of the document discounting factors as the normalization component; however, this will lead LDCG to violate the constraint C2 due to the over penalization of the list length. Nonetheless, using DCG's discounting factor or other decreasing functions as $p(i)$ is a more natural choice because $p(i)$ should represent the "prior" belief of the user regarding what position holds the relevant result.

Similar to NDCG, we can further rescale the score range of LDCG to $[0, 1]$ using the ideal LDCG (ILDCG) score: $ILDCG = \max\left\{ \frac{DCG}{E[DCG]} \right\}$. It is not hard to see that, if we use integer relevance labels (e.g., 0, 1, $\cdots$) and as long as the ranking list has no more than 25 results (which is arguably always the case for the constrained real estate), ILDCG would be obtained when the ranking list covers and only covers all of the highest relevant answers in the judgment set. For example, if we use relevance labels "perfect", "good", and "bad", each is associated with some integer value, the ILDCG will be obtained when all and only perfect results (or all good answers when there is no perfect answer) are included in the ranking list. Therefore, ILDCG can be calculated as:

$$ILDCG = \frac{\sum_{i=1}^{R} \frac{2^m - 1}{\log_2^{(i+1)}}}{Z(M, rel) \cdot \sum_{i=1}^{R} \left(\frac{1}{\log_2^{(i+1)}} \right)^2} \tag{7}$$

where R indicates the smaller one between M (the maximum number of results allowed) and the number of relevance judgments with the highest relevance degree m in the judgment set of the current query. Finally, with this ILDCG, the Length-adjusted NDCG is derived:

$$LNDCG = \frac{LDCG}{ILDCG} = \frac{DCG}{\sum_{i=1}^{N}\left(\frac{1}{\log_2^{(i+1)}}\right)^2} \cdot \frac{\sum_{i=1}^{R}\left(\frac{1}{\log_2^{(i+1)}}\right)^2}{\sum_{i=1}^{R}\frac{2^{rel_m}-1}{\log_2^{(i+1)}}} \tag{8}$$

where $Z(M, rel)$ can been mathematically eliminated in the calculation of LNDCG. It shows that *we do not really need to choose values for M and rel.*

From Formulas 6 and 8, we can see that, unlike DCG and NDCG, LDCG and LNDCG do not require any explicit length cutoff parameter, because they already model the length of the ranking list into the metric. Furthermore, it is not hard to verify that LDCG and LNDCG satisfy both formal constraints introduced in Section 3 unconditionally, as reported in Table 2.

5 Analysis

We use our running example query "jon favreau director" to compare the scores of LDCG/LNDCG and DCG/NDCG@k for several scenarios of interest. All the relevance judgments have been presented in Table 1, and we use the following settings of relevance labels: perfect= 2, good= 1, and bad= 0. We assume the maximum number of results allowed to show is $M = 3$ [2]. The comparison results have been shown in Table 3.

Our first observation is that the DCG/NDCG scores at different cutoff values vary significantly in the different cases, confirming our statement that it is hard to determine an appropriate parameter k in a constrained real estate application. Take Scenarios 6 and 7 as examples: when we compute DCG@2 or NDCG@2, Scenario 7 will be preferred, but when we compute DCG@3 or NDCG@3, Scenario 6 will be preferred, leading to inconsistent evaluation decisions. In contrast, the proposed LDCG/LNDCG, which can essentially be regarded as an aggregation of DCG/NDCG scores for different k values, produces a single score without relying on any explicit parameter k, suggesting that LDCG/LNDCG would be more stable and consistent for IR applications with constrained result real estate.

Second, LDCG/LNDCG is more effective than DCG/NDCG for length penalization. Comparing Scenarios 1 and 2, we can see that LDCG/LNDCG prefers cases where only perfect answers are shown in a clean way by means of length penalization. Another interesting comparison is between Scenario 4 and Scenario 5: Scenario 5 would lead to worse user experience than Scenario 4, because the former is exactly the latter with an additional bad result. DCG/NDCG does not

[2] We emphasize again that LDCG and LNDCG do not require any length cutoff parameter, as shown in Section 4.2. Yet we set M to some value just to help readers understand the LDCG calculation in Table 3 using the full-fledged Formula 6.

	Ranking Lists	DCG			LDCG	NDCG			LNDCG
		@1	@2	@3		@1	@2	@3	
1	Jon Favreau (director)	3	3	3	6.40	1	0.82	0.82	1
2	Jon Favreau (director) Iron Man 3	3	3.63	3.63	5.54	1	1	1	0.87
3	Jon Favreau (director) Jon Favreau (speechwriter)	3	3	3	4.58	1	0.82	0.82	0.72
4	Iron Man 3 Jon Favreau (director)	1	2.89	2.89	4.41	0.33	0.79	0.79	0.69
5	Iron Man 3 Jon Favreau (director) Jon Favreau (speechwriter)	1	2.89	2.89	3.74	0.33	0.79	0.79	0.59
6	Iron Man 3 Jon Favreau (speechwriter) Jon Favreau (director)	1	1	2.5	3.23	0.33	0.27	0.69	0.51
7	Jon Favreau (speechwriter) Jon Favreau (director) Iron Man 3	0	1.89	2.39	3.09	0	0.52	0.66	0.48
8	Jon Favreau (speechwriter) Jon Favreau (director)	0	1.89	1.89	2.88	0	0.52	0.52	0.45
9	Jon Favreau (speechwriter) Iron Man 3 Jon Favreau (director)	0	0.63	2.13	2.76	0	0.17	0.59	0.43
10	Iron Man 3	1	1	1	2.13	0.33	0.27	0.27	0.33
11	Iron Man 3 Jon Favreau (speechwriter)	1	1	1	1.52	0.33	0.27	0.27	0.24
12	Jon Favreau (speechwriter) Iron Man 3	0	0.63	0.63	0.96	0	0.17	0.17	0.15

Table 3. Comparison of LDCG with DCG@k and LNDCG with NDCG@k for several scenarios of interest.

deal with such cases well by giving the same scores to both scenarios at all k values, but LDCG/LNDCG does a clearly better job to score Scenario 4 higher than Scenario 5.

6 Conclusions

We proposed new IR metrics, namely Length-adjusted DCG and NDCG (LDCG and LNDCG), to evaluate IR applications with constrained result real estate. LDCG/LNDCG extend DCG/NDCG to be able to automatically achieve length normalization without requiring a length cutoff parameter k that is typically required by traditional IR metrics, while still preserving the good properties of DCG/NDCG. Our preliminary analysis shows that LDCG and LNDCG work better than existing metrics for evaluating IR applications with constrained real estate. In the future, we will conduct a thorough empirical evaluation of the proposed LDCG and LNDCG based on user studies and large-scale query log analysis.

Acknowledgments We thank Jay He, Jiayuan Huang, Dhyanesh Narayanan, and Bo Zhao for their helpful discussions. We also thank the anonymous reviewers for their useful comments.

References

1. Jaime Arguello, Fernando Diaz, Jamie Callan, and Ben Carterette. A methodology for evaluating aggregated search results. In *Proceedings of the 33rd European conference on IR research*, ECIR'11, pages 141–152, 2011.
2. Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, ICML '05, pages 89–96, 2005.
3. Luca Busin and Stefano Mizzaro. Axiometrics: An axiomatic approach to information retrieval effectiveness metrics. In *Proceedings of the 4th International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '13, 2013.
4. Olivier Chapelle, Donald Metlzer, Ya Zhang, and Pierre Grinspan. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM conference on Information and knowledge management*, CIKM '09, pages 621–630, 2009.
5. Aleksandr Chuklin, Anne Schuth, Katja Hofmann, Pavel Serdyukov, and Maarten de Rijke. Evaluating aggregated search using interleaving. In *Proceedings of the 22nd ACM International Conference on Conference on Information and Knowledge Management*, CIKM '13, pages 669–678, 2013.
6. Nick Craswell, Onno Zoeter, Michael Taylor, and Bill Ramsey. An experimental comparison of click position-bias models. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, WSDM '08, pages 87–94, 2008.
7. Hui Fang and ChengXiang Zhai. An exploration of axiomatic approaches to information retrieval. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 480–487, 2005.
8. Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, October 2002.
9. Yuanhua Lv and ChengXiang Zhai. Lower-bounding term frequency normalization. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, CIKM '11, pages 7–16, 2011.
10. Stephen E. Robertson, Evangelos Kanoulas, and Emine Yilmaz. Extending average precision to graded relevance judgments. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 603–610, 2010.
11. Ke Zhou, Ronan Cummins, Mounia Lalmas, and Joemon Jose. Evaluating large-scale distributed vertical search. In *Proceedings of the 9th Workshop on Large-scale and Distributed Informational Retrieval*, LSDS-IR '11, pages 9–14, 2011.
12. Ke Zhou, Ronan Cummins, Mounia Lalmas, and Joemon M. Jose. Evaluating aggregated search pages. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 115–124, 2012.