User-Oriented Metrics for Search Engine Deterministic Sort Orders

Alistair Moffat (ORCiD: 0000-0002-6638-0232)

School of Computing and Information Systems,
The University of Melbourne, Australia

Abstract

A recent paper proposes the "buying power" (bp) metric for assessing the quality of the product rankings generated by e-commerce sites such as Amazon and eBay. Focusing on the "ordered by price" type of product listing that is often viewed after a keyword search, bp is offered as a way of differentiating between helpful rankings (high bp scores) and unhelpful rankings (low bp scores), with those bp scores intended to reflect both the quality of the product match and also the relative pricing of the items that are listed. In this paper we adopt a user-centric viewpoint from which to evaluate the merits of bp as a scoring mechanism for product rankings, and provide an example that shows bp acting in opposition to likely user reactions. We then describe an alternative product ranking effectiveness metric, *price biased gain* (PBG), arguing that since it embeds a more plausible user model, it is more likely to reflect the opinions of the user viewing any given product ranking. We give a number of scenarios and motivating examples in support of our alternative proposal, and also discuss its limitations.

Keywords: information retrieval evaluation, e-commerce search, product ranking
2023 MSC: 00-01, 99-99

1. Introduction and Background

Search engine result pages (SERPs) are usually abstracted as ordered sequences of individual items, the i th of which is either relevant or not relevant to the user's information need (binary relevance, $r_i \in \{0, 1\}$); or has a relevance gain value over a continuous range (graded relevance, typically $0 \le r_i \le 1$). Given that context, a wide range of *effectiveness metrics* have been developed, taking as input a *gain vector* $\hat{r} = [r_i \mid 1 \le i \le k]$ and computing from it a numeric metric score, where k is the length of the ranking provided by the system, or is, at least, the length of the prefix of the ranking that is provided to the evaluation process.

For example, the well-known *precision at depth k* metric is computed as $\mathsf{Prec@}k(\hat{r}) = (\sum_{i=1}^{k} r_i)/k$, and the equally well-known *reciprocal rank* is defined for binary relevance as $\mathsf{RR@}k(\hat{r}) = 1/m(1, k)$, where $m(1, k)$ is the position in the ranking of the first relevant item, or infinity if there are no relevant items amongst the first k in the SERP.[1] Precision has the advantage of even-handedly weighing up the ranking as a whole, all the way through to the kth element; RR has the complementary advantage of tightly focusing on the effort the user must make to find a single relevant item. Between these two extremes a range of other top-weighted effectiveness metrics have been proposed, including *average precision*, AP [4]; *normalized discounted cumulative gain*, NDCG [10]; *rank-biased precision*, RBP [14]; *expected reciprocal rank*, ERR [7]; *time-biased gain*, TBG [20]; and metrics based around the C/W/L/A framework [16]. Moffat [13] discusses the overall context of evaluation in the context of search engine rankings, and the interpretation placed on effectiveness measurement.

Given that search engines are measured and compared via a combination of one or more of these metrics, when presented with a keyword query the search software faces the challenge of somehow estimating a *similarity score* s_d for each item d within its purview. It hopes (to the extent possible) to estimate an s_d that is correlated with the unknown value r_d, that is, the goal is for $s_d \approx f(r_d)$, where $f()$ is a monotonic relationship. The search system then creates a SERP by reverse-sorting the available elements according to the s_d estimates, and presenting a k-element prefix. Once that is done, the evaluation process measures the SERP using r_i values developed post-hoc, to obtain a score that is an assessment of the quality – or *usefulness* – of the search system's estimation algorithm.

In the case of *product search*, each possible item d also has a purchase price c_d associated with it. Similarly, there may be an "average review rating" attribute to be considered in the case of hotel search; a "distance from current location" attribute in the case of restaurant search; a "total journey duration" in the case of air-travel search; and so on. Moreover, product search services often allow users to "sort by attribute" as part of the search interface. When that option is requested, the elements in SERP are expected to comply, even though the ordering is algorithmically more complex [17, 21]. Taking "sort by increasing price" as a canonical "sort by" requirement, that means that in a k-item product SERP the computed similarity score s_d is no longer the only value used by the system when preparing the SERP, because the k items placed into the SERP must also comply with $c_i \leq c_{i+1}$ for $1 \leq i < k$.

That difference raises several key research questions:

RQ1: How should "sort by" product rankings be evaluated?

RQ2: Do metrics such as RR@k, average precision, or rank-biased precision, still give good guidance in regard to user satisfaction?

RQ3: Can tailored measurement approaches that incorporate a user browsing model provide more informative measurement of "sorted by" results listings?

[1] The SERP argument "$(\hat{r})$" to effectiveness measures will be omitted in the development below when there is no risk of ambiguity.

In a recent paper Trotman and Kitchen [23] seek to answer the first two of these three questions, proposing a metric that they name "buying power at depth k", bp@k. Trotman and Kitchen also describe a multi-item variant of bp@k which we denote here as bp4T(T)@k, with T representing the number of products that are to be purchased; that is, with bp4T(1)@$k \equiv$ bp@k as one particular possibility, corresponding to the situation when a single matching item is being sought for purchase.

In this paper we first critique Trotman and Kitchen's proposal, identifying anomalies in the behavior of bp@k and bp4T@k that suggest – at least in some cases – that they may not correlate well with user satisfaction. We then address **RQ3**, describing an alternative approach to product ranking effectiveness measurement that we argue avoids the issues that affect bp and bp4T, and corresponds to a more plausible user behavior model; and hence is likely to lead to product search evaluations that more accurately reflect the search experience of users.

2. Buying Power

This section first defines the bp@k product ranking effectiveness metric of Trotman and Kitchen [23]; and gives some examples of the computations that arise when using it. An alternative metric for scoring product search SERPs is then presented in Section 3.

2.1. An Example

As part of their presentation, Trotman and Kitchen provide the example shown in Table 1, assuming that the r_k relevance values in the second column are binary. Trotman and Kitchen suggest that users be regarded as "consuming" the elements in the ranking until they find one that is relevant, at which point they stop, in the same way that RR@k models the user as stopping at the first relevant answer they encounter. In the case of RR, the "effort" spent by the user to achieve that one relevant outcome is counted via the number of documents (or snippets) inspected before they exit the ranking [3]. Trotman and Kitchen propose that users be thought of as actually *purchasing* every item in the ranking down to, and including, the first one that is relevant. In Table 1, for example, Trotman and Kitchen argue that inspecting (and hence purchasing) the first three elements in the ranking "costs" the user $\$1 + \$2 + \$5 = \$8$. Moreover, given the external knowledge that the item exists and is available (somewhere) for \$2.50, the user would have ideally only needed to spend \$2.50. The ratio between these two, \$2.50/\$8, gives the listed bp@k score of 0.3125 shown in Table 1 for $k \geq 3$. Prior to depth $k = 3$ the user has spent money on unwanted goods, and not yet not acquired the product they were searching for; and hence bp@$1 =$ bp@$2 = 0$.

Rank k	r_k	c_k	$\sum_{i=1}^{k} c_i$	$m(1,k)$	$\mathsf{bp}@k$
1	0	~~\$1~~	\$1	∞	0
2	0	~~\$2~~	\$3	∞	0
3	1	\$5	\$8	3	0.3125
4	0	~~\$9~~	\$17	3	0.3125
5	1	\$11	\$28	3	0.3125
6	0	~~\$12~~	\$40	3	0.3125

Table 1: Example of the $\mathsf{bp}@k$ "buying power" metric, adapted from Trotman and Kitchen [23, Table 2]. A set of six products are presented in a SERP in "sort by price" order, with two of the products (shown without strike-through at ranks 3 and 5) regarded by the user as matching their purchase criteria, that is, as being relevant. In this example it is assumed that $c_{\min} = \$2.50$ is the best available price for the item being sought, but that the vendor offering that price did not get included in the SERP.

2.2. Definition

More precisely, suppose that a k-element SERP being evaluated is now abstracted as an ordered sequence of k "(relevance, cost)" pairs, $\hat{rc} = [(r_i, c_i) \mid 1 \le i \le k]$, and that

$$c_{\min} = \min\{c_d \mid r_d = 1\} \tag{1}$$

is the smallest cost of any relevant item across the whole collection (that is, including items that do not appear in the SERP being measured). Note that we do not include the product description, or a photograph, or other meta-data such as color in this abstraction, and that for SERP evaluation purposes we are interested solely in whether or not the user will regard the corresponding product as being what they searched for (represented by the binary indicator r_i), and its cost (represented by the strictly positive value c_i). Then

$$\mathsf{bp}@k = \begin{cases} c_{\min} / \left(\sum_{i=1}^{m(1,k)} c_i \right) & \text{if } 1 \le m(1,k) \le k \\ 0 & \text{if } m(1,k) = \infty\,, \end{cases} \tag{2}$$

where

$$m(1,k) = \min(\{i \mid 1 \le i \le k \wedge r_i = 1\} \cup \{\infty\}) \tag{3}$$

is, as was anticipated in connection with RR in Section 1, the rank position of the first relevant document amongst the first k items in the SERP, or infinity if there are no relevant documents in the first k. The two cases in Equation 2 represent, respectively, the situation in which the user exits the ranking having encountered a relevant item (the case $m(1,k) \le k$), and the situation where they reach the end of the ranking at depth k without finding a relevant item, in which case the sum $\sum_{i=1}^{m(1,k)} c_i$ is taken to be ∞, resulting in $\mathsf{bp}@k = 0$. The second to last

k	r_k	c_k	$\sum_{i=1}^{k} c_i$	$m(1,k)$	bp@k
1	0	~~$1~~	$1	∞	0
2	0	~~$1~~	$2	∞	0
3	0	~~$1~~	$3	∞	0
4	0	~~$1~~	$4	∞	0
5	1	$110	$114	5	0.8772

(a) Product search SERP from System A

k	r_k	c_k	$\sum_{i=1}^{k} c_i$	$m(1,k)$	bp@k
1	0	~~$100~~	$100	∞	0
2	1	$105	$205	2	0.4878
3	0	~~$110~~	$315	2	0.4878
4	0	~~$110~~	$425	2	0.4878
5	1	$110	$535	2	0.4878

(b) Product search SERP from System B

Table 2: Further examples of the buying power metric bp@k being used to score product search rankings. Two SERPs of length $k = 5$ are shown, both in "sort by price" order. In this example it is assumed that $c_{\min} = \$100$ is the best available price for the item being sought, and that Systems A (left) and B (right) are being compared via the bp@5 scores in the last row.

column in Table 1 shows $m(1, k)$ for each evaluation depth $1 \leq k \leq 6$; the last column then shows the corresponding bp@k values for the example SERP.

Trotman and Kitchen also mention a third stopping criteria, the case in which the user chooses to exit the ranking without finding a relevant item and also without reaching the end of k items presented in the SERP, but do not give an explanation of conditions (that is, a user model) under which that might happen. That third possibility does not affect the concerns presented in this section, and will be returned to in the next section.

2.3. A Surprising Comparison

The reader is now invited to consider the two rankings shown in Table 2, presented in the same format as was used in Table 1; and imagine that they come from two different search services that are being compared. Which is the better ranking? According to the bp@5 scores shown in the last row of the tables, System A on the left achieves a 0.8772 outcome, which makes it rather better than System B on the right, which only attains a score of 0.4878. But is that "System A > System B" outcome plausible from a user-experience point of view?

In our opinion the relationship illustrated in Table 2 is rather *im*plausible; and that in fact "System B > System A" is the conclusion that would be reached by almost all search system users considering those two SERPs to depth 5. In particular, the System A user must both look further through the ranking than the System B user before they find a relevant item, and if they do buy that first relevant item, the System A user is obliged to pay more for it. That is, *System A is worse than System B in terms of both price and search effort*, a "lose-lose" situation; and thus the suggestion that System A is better than System B seems untenable. Also of concern is that if the costs c_k are not included in the evaluation, and the two SERPs are compared purely based on the relevance column on each side of the table (headed "r_k"), any metric that preferred System A would be regarded as being axiomatically indefensible, see, for example, Busin and Mizzaro [5] and Moffat [12, 13].

The mismatch between the bp scores and anticipated user reaction is caused by the summation in Equation 2, and the assumption that as the user steps past non-relevant items they will purchase each of them, even though they are immediately identified as being not wanted. Trotman and Kitchen [23] defend this structure, writing[2] (in their Section 4.1):

> *imagine a user interacting with a digital assistant and saying "send me your cheapest box of cornflakes" and being sent . . . tuna. The postage price of returning the tuna is higher than the purchase price of the tuna so our customer does not return it, they simply return to their digital assistant and say "that isn't cornflakes, send me your cheapest box of cornflakes". Each time this happens they accumulate a loss equal to the cost of the item that was shipped.*

We suggest that the "too cheap to bother returning it" rationale might conceivably apply in regard to one tin of tuna, and might conceivably apply once, but is implausible as a general expectation. For example, in the scenario shown in Table 2(b) it seems unlikely that the $100 first item could be erroneously ordered, and then simply written off as being bad luck. Moreover, the "too cheap to bother returning it" assumption would also be unlikely to be tolerated indefinitely – would a user really continue to trust their digital assistant through the sequence of steps shown in Table 2(a), even if it were "merely" four tins of tuna that had arrived in four consecutive deliveries? (Or might the user, regardless of how cheap or expensive those first four purchases were, be tempted to make fifth request "Hey, digital assistant, send me a better digital assistant, and then switch yourself off"?)

In further explanation of their proposal, Trotman and Kitchen also write (their Section 9):

> *If the cost is distance . . . then this is equivalent to saying that the user must travel to [the restaurant] in order to discover that it is closed . . .*

But this argument also seems to be unrealistic. When looking at a "sorted by distance" list of restaurants, most users are likely to spend a minute or two considering each of the suggestions in the SERP, to (at a minimum) check that they are open. That is, broadly speaking, they will spend *constant* time evaluating each item in the ranking as they engage in the user-oriented process of making relevance decisions. They do *not* just walk to the closest suggestion, and if it is closed, sigh philosophically, walk back to their starting point, and then start walking to the option that was originally the second-closest location in the original SERP. At a minimum, they will search again using their new current location, as is also noted

[2] Noting that this example was motivated by a shopping SERP observed by Trotman and Kitchen [23] from a supermarket web site, which did indeed place flaked tuna ahead of cornflakes.

by Trotman and Kitchen, to minimize the extent of any backtracking that might be required. The additional suggestion made here is that, if they did indeed make the mistake of walking to a closed restaurant, they would learn from that experience, and would reformulate their query too, adding a "still open" requirement.

Finally in a "sort by travel duration" flight booking scenario, suppose that a search for flights from (say) Singapore to London offered a flight from Singapore to Melbourne at the top of the SERP, because of its shorter flight time. There is no question that the presence of a non-relevant item in that first position should be penalized. But it seems inconceivable to suggest that the traveler should actually "experience" the non-relevant flight from Singapore to Melbourne. Rather, they spend a constant (and, compared to the flight time, mercifully brief) time evaluating that option, form their relevance assessment (perhaps at the same time thinking, "well, that was weird, this service is a bit flaky"), and then move on to the next item in the SERP.

2.4. Finding T Relevant Items

Trotman and Kitchen [23] go on to consider the issue of scoring e-commerce SERPs in which the user wishes to find more than a single relevant product. They extend $\mathsf{bp}@k$ to suppose that T items are to be purchased,[3] defining:

$$\mathsf{bp4T}@k = \begin{cases} (T \cdot c_{\min,T}) \,/\, \left(\sum_{i=1}^{m(T,k)} c_i\right) & \text{if } 1 \le m(T,k) \le k \\ 0 & \text{if } m(T,k) = \infty \,, \end{cases} \tag{4}$$

where (again, assuming that all relevance values are binary, $r_i \in \{0, 1\}$)

$$m(T,k) = \min(\{i \mid 1 \le i \le k \wedge \left(\sum_{j=1}^{i} r_i\right) \ge T\} \cup \{\infty\}) \tag{5}$$

is the rank within the k products proposed in the SERP of the Tth relevant listing, or ∞ if there are fewer than T acceptable products within the first k, and where $c_{\min,T}$ is the average price of the cheapest T instances of a relevant product in the collection, and might be greater than $c_{\min}$. Given this formulation, it is clear that $\mathsf{bp}@k \equiv \mathsf{bp4T}@k$ in the case $T = 1$.

Note that $\mathsf{bp4T}@k$ is zero if there are fewer than T acceptable products in the k that are shown in the SERP, and there is no "partial credit" awarded. Trotman and Kitchen write (their Section 4.2):

> *If there are fewer than T* [replacing K by T throughout] *relevant items in the results list then the search engine cannot fulfil the user's needs . . . consequently,* [we] *give a score of* 0 *(even if the search engine can fulfil a request for $T-1$ items)*

[3] Trotman and Kitchen use the symbol K for that same quantity, but we prefer to use T to avoid any risk of confusion against k, the evaluation depth; and to align with other work in which the user is anticipated as wanting T relevant documents [15].

This decision means that in Table 2 both System A and System B would be assigned bp4T@5 scores of zero if the user had intended to purchase $T = 3$ products at the time they issued their query, even though it might be reasonably argued that System B has come closer to meeting the request than has System A.

2.5. Availability Counts

Trotman and Kitchen [23] further observe that product search interfaces also often include a "quantity available" indicator. In this case, we can abstract a SERP S as an ordered k-sequence of *triples*, namely $\hat{rcn} = [(r_i, c_i, n_i) \mid 1 \leq i \leq k]$, with the third component n_i a bound of the number of instances of the product that may be purchased at this time.

The presence of n_i complicates the bp4T@k mechanism. If (say) $T = 10$ boxes of cornflakes are being sought, the digital assistant risks ordering 10 tins of tuna, increasing the total amount wasted by the user. To address this dilemma, Trotman and Kitchen write (their Section 4.2):

> *If an item is not relevant then it is not relevant for that group of items . . . That is, the penalty is only given once . . .*

But this calculation is inconsistent with Trotman and Kitchen's "total dollars spent" motivation that was summarized in Section 2.3. If the digital assistant selects the product that is listed first in the SERP and orders T tins of tuna, are we now to assume that $T - 1$ of them are returned if a mis-match is detected, but that one tin is retained? Or, if the suggestion is that the digital assistant only ordered one instance rather than T, are we to assume that it somehow divined that the tuna was not the desired product? If so, why did it even order one tin, when it could have bypassed that product completely and gone on to the next element in the SERP?

3. A C/W/L/A Model for Price-Ordered Product Retrieval

Having expressed concerns in regard to the proposal of Trotman and Kitchen, we now present a grounded metric for ordered retrieval. We begin by documenting the "all other things being equal" behaviors that a sorted-by metric might be argued to require; then take direction and guidance from the C/W/L/A framework of Moffat et al. [16], describing a model of user behavior as price-ordered SERPs are consumed; and then translate that into an effectiveness metric and provide an operational description. Finally, we provide a range of numeric examples and SERP pair comparisons, and argue for the plausibility of the relationships that they imply.

We note that Smucker and Clarke [20], Azzopardi [1], Zhang et al. [25], Azzopardi et al. [2], and Su et al. [22] have also considered ways in which search behaviors might be modeled, with Moffat and Zobel [14] providing an early description of a metric based upon anticipated user behavior.

3.1. Underlying Principles

Following the lead of Moffat et al. [15], who describe a set of monotonic desiderata for normal effectiveness metrics using "all other things being equal" as an equalizing device to allow the effect of individual factors to be isolated, we propose the following principles for "sorted-by" evaluation metrics, supposing that the user is seeking T "units" of relevance at the lowest overall "price":

D1 All other things being equal, the higher the initial value of T, the greater the depth the user is likely to reach in the SERP before exiting.

D2 All other things being equal, the greater the price of the item at rank d is relative to the known minimum possible purchase price, the more likely it is that the user will exit the SERP without considering the merits of that dth or any subsequent item.

D3 All other things being equal, the closer the user is to having achieved their target of T units of purchase after considering the item at depth d, the more likely it is that the user will exit the SERP at that point, without considering the item at depth $d + 1$.

D4 All other things being equal, the lower the per item cost achieved is relative to the known minimum possible purchase price, the more satisfied the user is likely to be when they exit the SERP.

D5 All other things being equal, the closer the number of purchased items is to T, the more satisfied the user is likely to be when they exit the SERP.

The first three of these desiderata contribute to the manner in which the user proceeds through the SERP; and then the fourth and fifth contribute to their overall sense of satisfaction with the SERP at the time they end their perusal of it.

We trust that the reader will agree that these are all reasonable statements, and allow them to be used as the basis of a model for the manner in which users consume sorted-by SERPs. The next subsection summarizes a mechanism that allows such user models to then be interpreted as effectiveness metrics.

3.2. The C/W/L/A Framework and User Behavior

In the C/W/L/A framework [6, 16] user SERP browsing behavior is modeled via the function $C(i)$, the conditional continuation probability that the user will proceed from the item at rank i to the item at rank $i + 1$ in their sequential top-down scan through the SERP. In regular web search the conditional continuation probability might be influenced by any or all of: the value of i; the relevance values observed through until depth i (that is, r_j for $1 \leq j \leq i$); and by T, the relevance target sought. Knowledge of $C(i)$ then allows the "last" function $L(i)$ to be computed, the fraction of the user population that exits the SERP immediately after having examined the item at rank i:

$$L(i) = (1 - C(i)) \cdot L(i-1)\,,$$

with $L(0) \equiv 1$ as a base case for the recurrence. For example, if $C(i) = 1$ for some rank i, then $L(i) = 0$, because there are no users arriving at rank i that don't also continue on to rank $i+1$. A wide range of $C()$ functions have been considered in connection with retrieval effectiveness metrics [16]. One simple option is the constant function $C(i) = \phi$ for some constant ϕ, which is the defining equation for the *rank-biased precision* effectiveness metric, RBP [14].

A second function is then added: $A(i)$ represents the "aggregation" of utility perceived by a user that exits at rank i. There have again been a range of $A()$ functions proposed as being ways to capture the impression a user has assembled in regard to a SERP through until depth i, with possible influencing factors also including all of i; the relevance values observed through until depth i; and T. For example, one simple aggregation function is provided by $A(i) = \sum_{j=1}^{i} r_j$, the sum of the relevance encountered to that depth [16]. With $L(i)$ the fraction of the user population that exit the ranking at depth i, and $A(i)$ their perceived net benefit, the metric value is then the expectation on benefit:

$$\mathsf{Metric} = \sum_{i=1}^{\infty} L(i) \cdot A(i)\,. \tag{6}$$

To define a metric we thus need to propose a combination of $C()$ and $A()$, and then argue for them as being "appropriate" in some way. As an example, consider Trotman and Kitchen's "searching for one item" metric, $\mathsf{bp}@k$. For binary relevance, it is defined via the conditional continuation probability

$$C(i) = \begin{cases} 1 - r_i & \text{if } i < k \\ 0 & \text{if } i \geq k \end{cases} \tag{7}$$

which then means that $L(i) = 0$ unless one of two cases applies: when either $m(1, i) = i$; or when $i = k$ and $m(1, k) = \infty$. That continuation function is coupled with an aggregation function that computes $A(i) = c_{\min} / (\sum_{j=1}^{m(1,i)} c_j)$.

3.3. C/W/L/A For Shoppers

We suggest changes to both $C()$ and to $A()$ in order to obtain a product search effectiveness metric that we argue better resonates with the anticipated actions of shoppers.

First, consider the $A()$ function that indicates the "value" that is acquired by a user who exits the SERP after encountering the products shown at ranks 1 to i inclusive. We suggest that $A(i)$ needs to incorporate two separate components: one that indicates how satisfied the user is in regard to the price that they were obliged to pay in order to secure the items that they have purchased (relationship **D4** in Section 3.1); and a second that reflects how much progress they have made in their quest to purchase T items that match their query and intent relationship (**D5** in Section 3.1). Like Trotman and Kitchen, we handle the second of these two factors by summing the costs of the purchased items and computing a ratio between "best possible" and "actual". The second factor affecting $A(i)$ tracks how close each user is to their search goal. For example, if a user is engaged in a $T = 5$ search and at depth i would have purchased 4 items, this second factor is 4/5. In the first rule in Equation 8, we argue that the user's net state in regard to the outcome of their search is unchanged in the event that they encounter a non-relevant item. The first and second multiplicands in the second $r_i = 1$ rule in Equation 8 then correspond respectively to the two components described earlier in this paragraph:

$$A(i) = \begin{cases} A(i-1) & \text{if } r_i = 0 \\ (p_i \cdot c_{\min}/s_i) \cdot (p_i/T) & \text{if } r_i = 1\,, \end{cases} \tag{8}$$

in which $A(0) = 0$ is defined as a base case, and where p_i is the number of purchases made up to and including row i of the SERP, calculated via $p_0 = 0$ and then (recall that n_i is the availability of the item listed at position i in the SERP) the recurrence:

$$p_i = \begin{cases} p_{i-1} & \text{if } r_i = 0 \\ p_{i-1} + \min(n_i, T - p_{i-1}) & \text{if } r_i = 1\,. \end{cases} \tag{9}$$

Similarly, the quantity s_i is the total payment through to depth i that resulted in the purchase of those p_i chosen items:

$$s_i = \sum_{j=1}^{i} c_j \cdot (p_j - p_{j-1})\,.$$

For example, $A(i) = 0.5$ might arise via the purchase of T items at a total price of $2 \cdot c_{\min}$, or via the purchase of $T/2$ items at a best-possible price of $c_{\min}$ dollars each, or via some combination in between those two.

Note that Equation 8 also includes a small variation in the way that multi-item searches are normalized. Equation 4 (taken from Trotman and Kitchen [23]) makes use of a quantity $c_{\min,T}$, the average price across the T cheapest instances of a matching product in the collection. In line with the user-oriented perspective proposed here, we prefer to use the single value $c_{\min}$, assuming that the user will expect that as many products as are required are available in the collection at that minimum price, regardless of what the system is able to deliver. For example, suppose that a user wanted to purchase $T = 3$ items, but had to go to two different vendors and ended up paying \$100, \$100, and \$120 for them. That might indeed be the lowest combination of prices possible (with $c_{\min,T} = \$106.67$ in this instance), but the user might nevertheless feel a sense of chagrin as a result of their shopping experience, and not be "1.0 satisfied". This is why Equation 8 takes (as is the intention throughout this proposal) a user-oriented point of view, and a score of 1.0 is only generated if T items all costing exactly $c_{\min}$ are available at the head of the SERP. This slight change to the normalization regime also allows $c_{\min}$ to be set externally in some way, for example, if the user had seen an advertised lower price from a competing shopping service.

Next, consider the user behavior to be modeled using the $C()$ function. We want users to stop if they attain T units of relevance. We also expect that they will exit the SERP if they encounter "sticker shock", where the price of the items on offer has gradually increased to the point where they regard them as being overpriced. Finally, we also want users to eventually stop even if they don't encounter any matching products, and even if all of the listed products have the same price. That means that we cannot allow $C(i) = 1$ indefinitely for non-matching products, even if they all have the same price. Each of these three influences (corresponding in aggregate to relationships **D1**, **D2**, and **D3** in Section 3.1) is incorporated into the proposed $C(i)$ function shown in Equation 10, below. First, if the product at depth i was a match to the query and as a consequence the user has now acquired their target of T items purchased, they do not continue. That is achieved by explicitly setting $C(i) = 0$, and is captured as the first case in Equation 10.

Second, to measure the degree of sticker shock at depth i, and quantify its effect on $C(i)$, once $c_i \geq c_{\min}$ we compute the ratio c_i/c_{i+1}, comparing the price of the current item and the price of the next one as a way of gauging whether or not the user will continue. This factor will be 1.0 if the two items have the same price, which seems reasonable – once any particular user has stepped to any given price level, they can be assumed to be willing to continue considering products at that price level. That is, after they have evaluated the item at rank i in the SERP, we suggest that each user implicitly (or perhaps even explicitly) employs the price step to the next product in the SERP, at rank $i + 1$, as a factor in their decision as to whether the i th item is the last one they will consider. With this formulation, the telescoping $C(i)$ products mean that once c_i is double $c_{\min}$ at most 50% of the user population will still be consuming the SERP. The second case in Equation 10 covers this situation when $r_i = 1$, when one or more purchases have been made at depth i, and when more purchases are still hoped for.

Finally, to prevent endless scanning if there is a long sequence of non-matching products all of the same price, when $r_i = 0$ we also include the "fixed continuation probability" factor that was introduced as part of the rank-biased precision (RBP) effectiveness metric [14]. When the user has just viewed a non-matching product at depth i, they are modeled as computing the same price ratio between adjacent rows, but then subconsciously discounting it slightly, by some factor ϕ. The value chosen for ϕ then controls the background "dissatisfaction" probability of the user drifting away from the SERP. For example, if $\phi = 0.95$, users will in expectation look at 20 same-price rows in the SERP – each perhaps listing multiple identical non-matching items for sale – before they abandon their scanning. Given this context, the third rule in Equation 10 covers the case when the minimum purchase prices has not yet been reached; and the fourth rule then also incorporates the sticker shock of prices that increase beyond that value. In the examples reported shortly we use $\phi = 0.95$ as a persistence value for illustrative purposes. Any particular product search evaluation might choose to adopt different values of ϕ, based on specific knowledge of the behavior of their user population; and can also use other relationships between the various visible parameters.

Assembling these ideas we arrive at this four-way definition:

$$C(i) = \begin{cases} 0 & \text{if } r_i = 1 \text{ and } p_i \geq T\,. \\ c_i/c_{i+1} & \text{if } r_i = 1 \text{ and } p_i < T \\ \phi & \text{if } r_i = 0 \text{ and } c_i \leq c_{\min} \\ \phi \cdot c_i/c_{i+1} & \text{if } r_i = 0 \text{ and } c_i > c_{\min} \end{cases} \tag{10}$$

The SERP length k does not appear in Equation 10, and as a consequence the user model is able to handle the "infinite scroll" interfaces that are common in e-commerce sites. Nor is it necessary for T items to be acquired before stopping is permissible, and instead we allow users to drift away from (or be scared away from, by high prices!) the SERP before they have completed their intended set of purchases. On the other hand, Equation 10 by design ensures that users never purchase more than T products in total – they leave the SERP once they have their desired T items in their shopping basket.

If the SERP is fixed-length and contains k items, there are several options that can be considered for $C(k)$. One alternative is to say "well, that's it, there aren't any more products on offer", and define $C(k) = 0$. In this case the metric value and expected number of items purchased are calculated using those k available product rows. A second option is to calculate $C(k)$ via Equation 10, and then compute a *residual* [14], a score range based on the best and worst that might occur at ranks $k+1$ and beyond; that is, considering all possibilities for all of $(r_{k+1}, c_{k+1}, n_{k+1})$, then the same at depths $k+2$, $k+3$, and so on. An example showing this is provided shortly. The third alternative is to consider *session-level metrics*, with a "next page" conditional continuation probability computed via a page-level continuation function $C_p()$ that also accounts for the reluctance most users show in regard to crossing page boundaries [11, 26, 24].

3.4. User Model and Score Computation

In combination, Equations 8 to 10 then defined an evaluation measure for "ordered by" product SERPs that we refer to as PBG, standing for *price biased gain*, in which user progress through the SERP is affected by both price and quality, and in which their satisfaction upon exit is affected by the extent to which they have realized their search goal, and the expenditure associated with the items that they have purchased. Price biased gain is thus an expectation over a population of stochastic users, in the same was as RBP is an expectation over a cohort of users who have different behaviors as individuals, but predictable aggregate behavior when considered as a population.

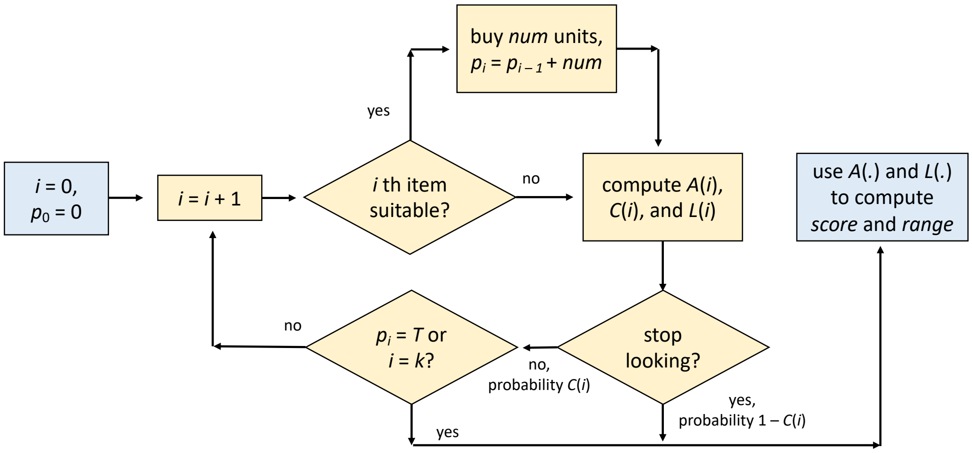

Figure 1: Flow diagram of user model when viewing "sort by price" e-commerce SERPs. Algorithm 1 provides further details of how this user model corresponds to a calculable metric.

Figure 1 shows in schematic form the user model that is assumed in this proposal and for which Equations 8 and 10 provide the details. The same structure is also captured in the pseudocode shown in Algorithm 1. A population of users commences browsing the SERP at item $i = 1$, and examines products in the order they are presented. If the product matches the query that was issued, items are acquired from that product row, with p_i and s_i being larger than p_{i-1} and s_{i-1}. Or, if that product row is not a match, p_i and s_i are equal to p_{i-1} and s_{i-1}. Either way, the values of $A(i)$ and $C(i)$ are next calculated, with the latter determining $L(i)$, and then a new fraction (recorded in variable *frac* in Algorithm 1) of the user population that are regarded as still being active within the SERP is computed.

The metric value is then calculated from $L(i)$ and $A(i)$ at step 13, as determined by Equation 6. If *frac* reaches zero by the end of the SERP at depth k, the metric computation ends at a single value. This occurs when T items have been accumulated. The case when *frac* > 0 is discussed in Section 3.6; step 17 in Algorithm 1, and the variable *range*, will also discussed at that time.

3.5. Detailed Example

With Equations 8 to 10 defining a C/W/L/A metric PBG that has been argued for by considering user product SERP browsing behavior and user "value assessment", it remains now to consider what occurs for typical product rankings. Table 3 provides a first example, in which a set of $T = 6$ items are desired. They can be purchased at ranks two (3 items), four (another 2 items), and five (1 more item selected, of the 3 available for purchase), for a total purchase price of $\$12 \times 3 + \$14 \times 2 + \$15 \times 1 = \$79$. The value of $A(5)$ is thus $(6 \times 10/79) \cdot (6/6) = 0.7595$, as shown in the table – a user who exits the SERP at rank 5 is

Algorithm 1 Computing an effectiveness score for SERP $S = [(r_i, c_i, n_i) \mid 1 \leq i \leq k]$, where T items are required, the minimum cost is $c_{\min}$, and the background persistence parameter is ϕ. This pseudocode should be considered in conjunction with Figure 1.

```
function $scorer(S, k, c_{\min}, T, \phi)$
    $frac \leftarrow 1$                                  ▷ fraction of all users that are still viewing SERP $S$
    $A(0) \leftarrow 0$ and $p_0 \leftarrow 0$ and $s_0 \leftarrow 0$
    for $i \leftarrow 1$ to $k$ do                       ▷ for each supplied rank position in $S$
        if $r_i = 1$ then
            $num \leftarrow \min(n_i, T - p_{i-1})$      ▷ useful item, make some "purchases"
            $p_i \leftarrow p_{i-1} + num$ and $s_i \leftarrow s_{i-1} + num \cdot c_i$
        else
            $p_i \leftarrow p_{i-1}$ and $s_i \leftarrow s_{i-1}$    ▷ non-useful item, no purchases made
        compute $A(i)$ and $C(i)$ using Equations 8 and 10
        $L(i) \leftarrow (1 - C(i)) \cdot frac$
        $frac \leftarrow frac \cdot C(i)$                ▷ update the fraction of all users still active
    $score \leftarrow \sum_{i=1}^{k} L(i) \cdot A(i)$    ▷ compute the metric inner product
    if $frac = 0$ then
        $range \leftarrow 0$                             ▷ all users have been accounted for, no score uncertainty
    else
        $range \leftarrow$ compute score range from $L(i)$ and $A(i)$ by searching over $c_{k+1}$
    return $score$ and $range$
```

sated in terms of quantity, and "76% satisfied" in terms of price paid. Because there are six items located by rank five, $C(5) = 0$, and all of the users in the population are modeled as having exited the SERP by that point. Over all depths (that is, all exit points from depth 1 to depth 5) the expected benefit gained is 0.6008, and hence this is the score assigned to this combination of SERP, T, and ϕ. That is, balancing quantity acquired and price paid, the expected "satisfaction of users" across all SERP exit points is 60%. In this search instance all users end their scanning prior to the end of the SERP, meaning that the metric score (the final value in the last row) is fully determined.

Since $L(5) = 66\%$ and $p_5 = 6$, two-thirds of the user population are modeled as exiting the SERP having accumulated $T = 6$ items of the suitable products. But the other third of the user population is modeled as exiting the SERP with fewer than six items. In expectation, a $T = 6$ search in the SERP shown in Table 3 results in $\sum_{i=1}^{k} L(k) \cdot p_k = 4.69$ items being acquired. Broadly speaking, the higher the metric score, the closer the expected number of items acquired will be to T, the user's initial search target; and at the same time, the closer the average purchase price will be to $c_{\min}$. The implications of the average user only securing

k	r_k	c_k	n_k	buy	p_k	$C(k)$	$L(k)$	$A(k)$	$L(k) \cdot A(k)$
1	0	~~$11~~	2	–	0	0.8708	0.1292	0.0000	0.0000
2	1	$12	3	3	3	1.0000	0.0000	0.4167	0.0000
3	0	~~$12~~	5	–	3	0.8143	0.1617	0.4167	0.0674
4	1	$14	2	2	5	0.9333	0.0473	0.6510	0.0308
5	1	$15	3	1	6	0.0000	0.6618	0.7595	0.5027
6	0	~~$18~~	4	–	6	0.0000	0.0000	0.7595	0.0000
Totals				6			1.0000		0.6008

Table 3: An example showing the new PBG proposal for sorted product search evaluation. It is assumed that $c_{\min} = \$10$ is the best available price, that $T = 6$ items are sought, and that the background persistence parameter ϕ has the value 0.95.

k	r_k	c_k	n_k	buy	p_k	$C(k)$	$L(k)$	$A(k)$	$L(k) \cdot A(k)$
1	0	~~$11~~	2	–	0	0.8708	0.1292	0.0000	0.0000
2	1	$12	3	3	3	1.0000	0.0000	0.2500	0.0000
3	0	~~$12~~	5	–	3	0.8143	0.1617	0.2500	0.0404
4	1	$14	2	2	5	0.9333	0.0473	0.3906	0.0185
5	1	$15	3	3	8	0.8333	0.1103	0.5872	0.0648
6	0	~~$18~~	4	–	8	0.0000	0.5515	0.5872	0.3238
Totals				8			1.0000		0.4475

Table 4: A second example of the proposed PBG mechanism. It is again assumed that $c_{\min} = \$10$ is the best available price, and that $\phi = 0.95$. But now $T = 10$ items are sought, and the known SERP prefix down to depth $k = 6$ can only supply $p_k = 8$ of them.

4.69 items when they had intended to acquire $T = 6$ are discussed in Section 4.2.

3.6. Score Ranges and Residuals

Table 4 shows the same SERP, but now evaluated against $T = 10$. There are fewer than $T = 10$ matching products available in the visible SERP, and hence no naturally-arising value $C(i) = 0$. So now the metric calculation is forced to end by setting $C(6) = 0$ in the last row of the table, mirroring the behavior of users who might wish to scan more product rows, but find they cannot. The final score of 0.4475 indicates that in expectation a $T = 10$ user is less satisfied than the $T = 6$ users depicted in Table 3. The $T = 10$ score that arises from $C(k) = 0$ truncation at $k = 6$ also corresponds to normal processing (that is, following the progress established by Equation 10) of an infinite SERP in which there are no more matching products to be found at any ranks, and hence for which $p_i = 6$ for all $i \geq 7$.

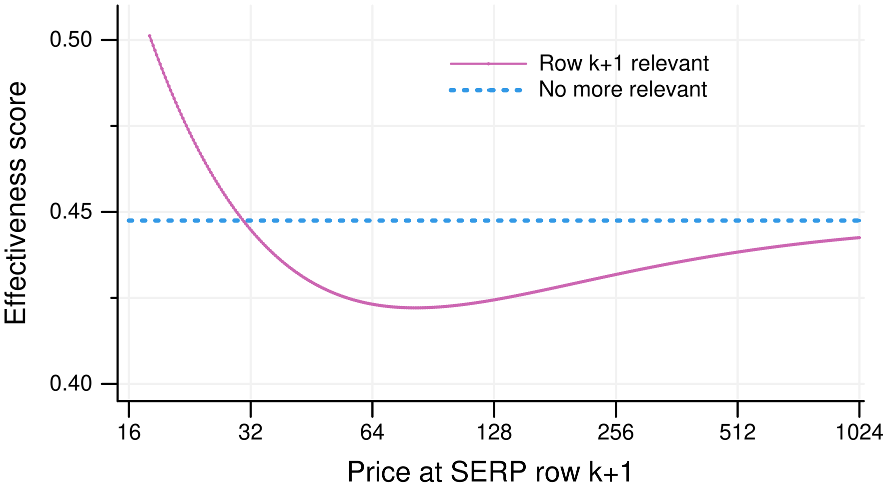

Figure 2: Final score as a function of the hypothetical c_{k+1} value, the price at depth $k+1$ at which the remaining $T - p_k = 2$ units of demand are acquired in connection with the scenario shown in Table 4. The dashed horizontal line shows the score that would result if no more matching products appear below depth k in the ranking.

It might thus be tempting to assume that 0.4475 is a lower bound on the score that can result if what is being measured is indeed a $k = 6$ fixed-length prefix of an infinite SERP. And that if any further relevance did occur in the non-visible SERP tail at $k \geq 7$, then the score would become greater than 0.4475 – in other words, that 0.4475 marks the bottom end of the score *range* that would emerge if the prefix was extended. But that assumption is incorrect, an observation that is illustrated in Figure 2.

In Figure 2 the horizontal axis depicts possible prices c_{k+1} at depth $k+1 = 7$ in the setting established by Table 4, starting at $c_{k+1} = c_k = \$18$. Each possible value for c_{k+1} determines (via Equation 10) a value for $C(k)$ which, in turn, determines (via Equation 8) the contribution to the metric score that results from $A(k) \cdot L(k)$. That contribution replaces the corresponding one in the $k = 6$ row in Table 4. Moreover, if it is supposed that $r_{k+1} = 1$ and $n_{k+1} \geq T - p_k$, so that all unmet demand can be met at depth $k+1$ and hence that $C(k+1) = 0$, then the hypothesized value for c_{k+1} can also be used to determine s_{k+1}, and then $A(k+1)$, and hence the contribution $A(k+1) \cdot L(k+1)$. Those two score contributions – a revised penultimate one at rank k and an added final one at $k+1$ – complete the evaluation of the metric, and those final metric scores are what is plotted on the vertical axis in Figure 2.

The vertical extent of the curve in Figure 2 thus represents the spread of metric scores that can arise as c_{k+1} varies. As can be seen, the maximum metric score occurs when $c_{k+1} = c_k = \$18$, and the purchase can be completed at the "current price". That occurs at the left end of the graph. Then, as the price c_{k+1} increases, the final metric score decreases; moreover, it decreases *below* the value that would result if there were no more matching products at all (the dashed line). Then, once c_{k+1} starts becoming so large that $C(k) \to 0$, the metric climbs again, back towards the "no more matching products" value shown in the last row of Table 4 and also plotted as the dashed line in Figure 2.

For any given SERP this relationship is captured by:

$$\mathsf{PBG@}(k+1) = \mathsf{PBG@}(k-1) + \mathit{frac} \cdot ((1 - C(k)) \cdot A(k) + C(k) \cdot A(k+1)) \qquad (11)$$

in which c_{k+1} is the only independent variable; in which all of c_k, p_k, s_k, and $p_{k+1} = T$ are constant; in which $C(k+1) = 0$ by definition; in which $\mathsf{PBG@}(k-1) = \sum_{i=1}^{k-1} L(i) \cdot A(i)$ is constant and is the sum of the first $k-1$ terms of the metric computation; in which $\mathit{frac} = \prod_{i=1}^{k-1} C(i)$ is constant and is the fraction of users that enter the kth row of the SERP; and in which s_{k+1} and $C(k)$ depend on c_{k+1}. Equation 11 is thus minimized when

$$(1 - C(k)) \cdot A(k) + C(k) \cdot A(k+1) \qquad (12)$$

is minimized. In the case $r_k = 1$ Equation 12 expands to

$$\left(1 - \frac{c_k}{c_{k+1}}\right) \cdot \frac{p_k^2 \cdot c_{\min}}{s_k \cdot T} + \frac{c_k}{c_{k+1}} \cdot \frac{T \cdot c_{\min}}{s_k + (T - p_k) \cdot c_{k+1}}\,; \qquad (13)$$

and when $r_k = 0$ it expands to either

$$\left(1 - \frac{\phi \cdot c_k}{c_{k+1}}\right) \cdot \frac{p_k^2 \cdot c_{\min}}{s_k \cdot T} + \frac{\phi \cdot c_k}{c_{k+1}} \cdot \frac{T \cdot c_{\min}}{s_k + (T - p_k) \cdot c_{k+1}}\,, \qquad (14)$$

or to

$$(1 - \phi) \cdot \frac{p_k^2 \cdot c_{\min}}{s_k \cdot T} + \phi \cdot \frac{T \cdot c_{\min}}{s_k + (T - p_k) \cdot c_{k+1}}\,. \qquad (15)$$

All of Equations 13 to 15 can be differentiated with respect to c_{k+1} and equated to zero to determine their minimizing c_{k+1} values, for substitution back into Equation 11; or (as we have done for the results provided in this paper) an exhaustive scan can be performed over c_{k+1} values in steps of (say) \$0.01, starting at $c_{k+1} = c_k$. For the example presented in Table 4 and plotted in Figure 2 the minimizing price is \$81.75, and the attainable score range is from 0.4221 at $c_{k+1} =$ \$81.75 to 0.5012 at $c_{k+1} =$ \$18.00, with (in expectation) between 6.25 and 7.06 items purchased respectively. Note that both of those final values – as is the expected purchase quantity associated with every other point on the red line in Figure 2 – are greater than the expectation of 6.02 items purchased if it is assumed that there are no further relevant items in the ranking at any price (the blue dashed line).

3.7. Further Examples

The next several figures illustrate the behavior of the PBG metric in a range of simple situations, and compare that behavior to the scores that would be assigned to the same SERPs via the bp4T approach of Trotman and Kitchen [23].

SERP A	SERP B	SERP C
$110	~~$110~~	~~$110~~
~~$120~~	$120	~~$120~~
~~$130~~	~~$130~~	~~$130~~
~~$140~~	~~$140~~	$140
$150	$150	$150

SERP	bp4T@5 score	PBG@5 score	PBG@5 purch.
A	0.3077	0.6524	1.63
B	0.3077	0.5666	1.50
C	0.3077	0.4497	1.30

Figure 3: Example showing top-weightedness effects on SERPs of length $k = 5$, assuming that $c_{\text{min}} = \$100$ is the minimum price for the desired product, that $T = 2$ items are being sought, that each row offers a single product instance ($n_i = 1$ throughout), and that for PBG the global parameter $\phi = 0.95$. The calculation of bp4T also assumes that there are two or more items available at $c_{\text{min},T} = \$100$, none of which were surfaced in these SERPs. The final column in the score table shows the expected number of items acquired by the users modeled by PBG.

SERP D	SERP E	SERP F
~~$10~~	~~$70~~	~~$105~~
~~$10~~	~~$70~~	~~$105~~
~~$10~~	~~$70~~	~~$105~~
~~$10~~	~~$70~~	~~$105~~
$110	$110	$110

SERP	bp4T@5 score	PBG@5 score	PBG@5 purch.
D	0.6667	0.7405	0.81
E	0.2564	0.7405	0.81
F	0.1887	0.7068	0.78

Figure 4: Example showing the effect of cheap incorrect products entering the SERP, assuming that $T = 1$ items are sought, with other parameter settings as for Figure 3.

Figure 3 considers three SERPs with $T = 2$. Because bp4T scores rankings based on the sum of all product prices down to the position of the T th purchase, all of SERPs A, B, and C are assigned the same score. On the other hand, even though the second purchased item is at rank five in each example SERP, PBG prefers SERPs A and B to SERP C, because the first of the two matches is found earlier, and costs less. The bp4T user is modeled as always making two purchases from these rankings, whereas the PBG user is modeled as acquiring between 1.30 and 1.63 items – some of the users are discouraged by the non-relevant items, and leave the SERP before making two purchases, or (in the case of SERPs B and C) even before finding one item to buy. Note that in PBG the degree of top-weightedness can be adjusted via the parameter ϕ; here we have modeled relatively patient users by setting $\phi = 0.95$, so as to be consistent with the earlier examples. Choosing a smaller value such as $\phi = 0.5$ would create harsher penalties for SERPs B and C and result in PBG being more strongly top-weighted, if that is what is desired.

Figure 4 illustrates the situation that was noted in connection with Table 2, in which

SERP G	SERP H	SERP I
\$110	\$110	\$110
~~\$120~~	~~\$120~~	~~\$120~~
\$130	\$130	~~\$130~~
~~\$140~~	~~\$140~~	~~\$140~~
\$150	~~\$150~~	~~\$150~~

SERP	bp4T@5 score	PBG@5 score	PBG@5 purch.
G	0.4615	0.6474	2.47
H	0.0000	0.4742–0.6404	1.80–2.43
I	0.0000	0.2885–0.5591	1.00–2.19

Figure 5: Example showing scores assigned when there is a product shortfall, and fewer than $T = 3$ items can be supplied, with the other parameters as described for Figure 3, and assuming that there are three or more items available at $c_{\min} = c_{\min,T} = \$100$, none of which surfaced in these SERPs.

implausibly cheap – and hence non-matching – items occupy early positions in the SERP. Because bp4T sums the prices of all of the products in the SERP rather than just the relevant ones, its scores are affected by items the users does not want. In particular, bp4T rates SERP D much more highly than SERP E, whereas PBG scores them equally. In this trio of SERPs PBG down-rates SERP F slightly, because in this one the incorrect items have prices greater than $c_{\min}$, which is modeled as being somewhat discouraging to the user (see Equation 10). The users modeled by bp4T will again always acquire one item, whereas the users modeled by PBG will in expectation only acquire around 0.8 items each, because none of these rankings is considered by a PBG user to be "ideal". This difference in purchase behavior, and what it then suggests in terms of longer-term user actions, is discussed further in Section 4.2.

A final trio of example SERPs is shown in Figure 5. Amongst these three, only SERP G can deliver the $T = 3$ items that are sought, whereas SERPs K and L cannot. As a result, bp4T assigns scores of zero to the latter two SERPs and signals them as being failures. On the other hand, PBG assigns partial credit for partial product supply; note also the computed ranges for both PBG score and PBG acquisitions, in accordance with the discussion in Section 3.6.

In combination, the nine SERPs shown in Figures 3 to 5 also illustrate the effects of the five monotonic desiderata that were proposed in Section 3.1 and then incorporated into PBG via the various factors making up Equations 8 and 10.

4. Discussion and Conclusions

We have presented a batch effectiveness measure for price-ordered and other "sorted-by"-style SERPs, expressed in the C/W/L/A framework by describing a continuation function $C(i)$ and an aggregation function $A(i)$. Our argument in favor of the proposed formulation of these two functions is largely based on rhetoric, and in essence is an appeal to what is probably best referred to as "informed common sense". We have argued for the plausibility of the relationships that the two functions encapsulate, based on what we believe to be likely user behavior, but without being able to prove that our claims are correct, nor demonstrate their validity experimentally. We comment further on that aspect of our work shortly.

Our presentation commenced with a critique of Trotman and Kitchen's bp4T metric, and the observation that it gives rise to some SERP relationships that we regard as implausible. Developing a "sorted by" metric that avoids those concerns was a key aim of this work; but we also regard the manner in which we have presented the PBG metric as being a further contribution, in that it serves as a tutorial on the C/W/L/A framework, showing how the framework allows effectiveness metrics for new applications to be developed based on principled foundations. Given that context, the second subsection below considers limitations that may affect the validity of those relationships, and considers ways in which they might be varied. The third subsection then considers possible future directions in which this work might be taken.

4.1. Experimentation

Research in information retrieval is often presented as a mix of theory and experimentation, and most research papers contain both – new ideas are motivated by insightful research questions and are then validated via experimentation. In the area addressed by this paper – that of effectiveness measurement and the connection through to user satisfaction – experimentation must thus connect metric scores and user-perceived SERP usefulness, since the latter is what the score is intended to approximate. This kind of experimentation is challenging to carry out, and is a research study in its own right. For example, Zhang et al. [27], Sakai and Zeng [19], and Moffat et al. [16] have all carried out studies that seek to measure correlations between metric scores and either user self-reported satisfaction, or some measurable surrogate for it such as click, scrolling, or reformulation behavior.

To experimentally confirm that any proposed "sorted-by" effectiveness metric of the type presented here is a suitable choice requires that similar studies be undertaken. In particular, it is worth reiterating that the scores assigned by different metrics are incomparable and of themselves do not allow comparison – statements such as "metric X assigns a higher average score than does metric Y" are meaningless, even when statistically significant.

Another possible approach is to apply a suite of metrics to a suitable dataset of queries, relevance judgments, and system runs (SERPs), so as to establish the relative system ordering induced by each of the alternative metrics. Indeed, this is exactly the nature of the experimentation reported by Trotman and Kitchen, over a collection of 150 topics and a set of fourteen systems. In such an experiment, if a proposed new metric consistently induces the same system ordering as another metric, we might conclude that one (or the other) of them is redundant; or might conclude that the pool of systems is too small to draw out any difference. But if the new metric induces a different ordering, we can only conclude that it is measuring something different, and must not make the mistake of believing that "different" equates to "better".

The challenge of establishing and interpreting a suitable experimentation regime is why we have avoided tabulations of system scores and correlation coefficients in this paper. Instead we have established plausibility by employing the C/W/L/A framework, and then arguing for certain user behaviors in the context of the user model that is embedded in that framework. If the desiderata **D1** to **D5** and Equations 8 to 10 resonate with anticipated user behavior, then their combination into a metric must represent a useful development.

Carefully structured user studies that employ "sorted-by" rankings of different qualities, and seek to directly correlate user satisfaction and metric scores, represent a substantial fresh challenge that we defer for future work.

4.2. Limitations

While our proposals for the $C()$ and $A()$ functions are motivated by a desire to capture, respectively, user browsing behavior and user aggregate assessment, it needs to be acknowledged they express broad influences rather than exact numeric relationships. For example, the use of c_i/c_{i+1} in the definition of $C()$ captures our belief that users will become increasingly discouraged as prices rise. Given that belief, division of c_i by c_{i+1} is merely one obvious choice that leads to the desired relationship, and hence happens to be the one written into Equation 10. But there are myriad other possibilities, and, in most general terms, what we are arguing for is that $C()$ is correlated with some further function $f(c_i, c_{i+1})$ that is non-decreasing in its first argument and non-increasing in its second. With detailed user-based experimentation and voluminous data it might be possible to infer that function $f(\cdot, \cdot)$ and all of the consequential coefficients. In other words, Equations 8 and 10 are proposed not because we believe that they represent the "exact best formulation", but because they represent directions in which "good" or even "great" formulations might lie. Similarly, the parameter ϕ might also take on different values in different application areas, and our use of $\phi = 0.95$ here is illustrative and most certainly not intended to be prescriptive.

Another area of possible concern is the assumption that $c_{\min}$ is somehow known to the community of users, since the $C()$ and $A()$ functions make use of that value when calculating the continuation probability and the aggregated benefit. The `bp` and `bp4T` metrics make the same assumption [23]. While it might be that each user has a fair idea of what the best possible price for any particular product might be, that still doesn't necessary match the value of $c_{\min}$ in the collection, and might be higher or lower. To address that discrepancy we could consider formulating Equations 8 and 10 according to each user's perception of the best possible price available and hence further personalize the set of responses to any given SERP, but would then need to introduce a distribution over individual $c_{\min}$ values, and carry that through the computation; and personalizing by user would also mean that we should consider introducing a distribution over T too, since each user is likely to be seeking a different purchase quantity, even if they issue the same query to find it. Rather than add to the complexity of the definitions, we prefer to assume that at the time they issue their query the user is in possession of an accurate estimate for $c_{\min}$. We note that Diaz and Ferraro [8] explore comparison techniques for normal SERPs that make use of a distribution over T, the desired quantity of relevance being sought.

A third aspect in which our proposal might be refined is in regard to unsatisfied demand. The bp4T mechanism of Trotman and Kitchen reports a score of zero if there are fewer than T purchasable items in the visible SERP, whereas (as a conscious preference, seeking to model the user's experience) our mechanism assigns a partial score for partial satisfaction. Moreover, even when there *are* T purchasable items in the visible SERP, our mechanism may, as an expectation over the universe of users modeled by the probabilistic C/W/L/A framework, result in fewer than T items on average being purchased. For example, the $T = 6$ example shown in Table 3 results in a prediction that the average user being modeled by the PBG user model will purchase 4.69 items. What does the user do then to make up the shortfall? As with all search tasks, we suggest that after they exit each SERP users make a higher-level decision: "am I now sufficiently satisfied, or will I reformulate my query and examine another SERP, or will I open a new browser tab and try my query at a different provider?" Should they decide to reformulate or to change provider, this user can be regarded as now searching for $T' = 1.31$ items. But note that the new SERP might be a more fertile source of products than the original one, and as a result of viewing the second SERP the user might even remove some of the items sitting in their current shopping basket, preferring instead cheaper products that may have been surfaced by the second SERP.

4.3. Future Directions

Like Trotman and Kitchen [23], we have focused on binary relevance, with $r_i \in \{0, 1\}$ for each element in the SERP. However, in the context of web search a wide range of *continuous* relevance mechanisms have been developed, including ERR [7], which is a continuous version of RR; NDCG [10]; and RBP [14]. Other binary relevance metrics have also had continuous gain modes added [9, 18].

Another possible way of thinking about sorted-by product SERPs could thus be that each gain value indicates the user's degree of interest in that product, with price factored in. For example, an unwanted product might always have a gain of zero, regardless of how cheap it is; with a matching product having a gain of one only if its price $c_i = c_{\min}$, to directly reflect its utility to the user. With this approach any search effectiveness metric has a sorted-by equivalent possible. Moreover, metrics such as INST [15], which have T as a parameter denoting the user's desired volume of relevance, might then be suitable as alternatives to bp4T and PBG.

A benefit of this approach is that products if have both "degree of match" indicators and "degree of best price component", then the two can be combined. For example, if a blue 256 GiB iPhone has relevance $r_i = 1$ in response to the query "blue iPhone 256 memory", then maybe a blue 128 GiB iPhone has a background relevance of $r_i = 0.8$ to the user, and a grey 256 GiB iPhone has background relevance of $r_i = 0.9$, with those relevance values converted to metric gain values by multiplying them by $c_{\min}/c_i$. Developing such approaches by devising the corresponding $C()$ and $A()$ functions might be a further useful line of investigation, complementing the point of view that we have taken here.

Different interface options also require different evaluation regimes. The discussion in Trotman and Kitchen [23] and the current work abstracts SERPs as one-dimensional lists, each containing k product options, and assumes that each user consumes each SERP sequentially from top to either bottom, or until exit. But many interfaces display two-dimensional grids, especially when images are also involved, with users following different scanning paths through the grid. Each user can thus be linearized, but with potentially multiple different linearization patterns. Extending the C/W/L/A framework into another probabilistic dimension, using a "what gets looked at next" distribution and a two-dimensional $C(\cdot,\cdot)$ function, is another interesting possibility.

It might also be interesting to look at multi-objective "sort by" operations. For example, could a "sort by price *and* distance" grid be formed with price ordered via one dimension, and distance in the second? And if so, how should it be evaluated?

A final area in which more development may be possible is the issue of sessions. The previous subsection noted that PBG assumes that shoppers might obtain only part of their need from any given SERP, purchasing fewer than T items either because T are not available, or because they became discouraged and exited the SERP prior to encountering T useful items. Submitting a reformulated query is one way of then seeking to expose further useful items; as is the more curt step of switching to a different provider. But switching provider might bring in different costs, such as further delivery fees, and so on. This is another aspect in which shopping services might differ from conventional search, with a comprehensive evaluation framework needing to allow for sessions, and all of the nuances they entail, as well as individual SERPs. Probabilistic session-level metrics have been considered for normal web search tasks [24], and might be able to be combined with the PBG mechanism described here, and the C/W/L/A framework in general, to develop ways of describing models for session-level evaluation of sorted-by product search sequences, should they be required.

Acknowledgements

This work was in part supported by the Australian Research Council (project DP190101113). The anonymous referees provided a range of helpful comments.

References

[1] L. Azzopardi. Modelling interaction with economic models of search. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 3–12, 2014.

[2] L. Azzopardi, P. Thomas, and N. Craswell. Measuring the utility of search engine result pages: An information foraging based measure. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 605–614, 2018.

[3] L. Azzopardi, J. Mackenzie, and A. Moffat. ERR is not C/W/L: Exploring the relationship between expected reciprocal rank and other metrics. In *Proc. Int. Conf. on Theory of Information Retrieval (ICTIR)*, pages 231–237, 2021.

[4] C. Buckley and E. M. Voorhees. Retrieval system evaluation. In E. M. Voorhees and D. K. Harman, editors, *TREC: Experiment and Evaluation in Information Retrieval*, chapter 3, pages 53–78. MIT Press, 2005.

[5] L. Busin and S. Mizzaro. Axiometrics: An axiomatic approach to information retrieval effectiveness metrics. In *Proc. Int. Conf. on Theory of Information Retrieval (ICTIR)*, pages 1–8, 2013.

[6] B. Carterette. System effectiveness, user models, and user utility: A conceptual framework for investigation. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 903–912, 2011.

[7] O. Chapelle, D. Metzler, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, pages 621–630, 2009.

[8] F. Diaz and A. Ferraro. Offline retrieval evaluation without evaluation metrics. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 599–609, 2022.

[9] G. Dupret and B. Piwowarski. A user behavior model for average precision and its generalization to graded judgments. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 531–538, 2010.

[10] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. on Information Systems*, 20(4):422–446, 2002.

[11] M. Liu, Y. Liu, J. Mao, C. Luo, and S. Ma. Towards designing better session search evaluation metrics. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 1121–1124, 2018.

[12] A. Moffat. Seven numeric properties of effectiveness metrics. In *Proc. Asia Information Retrieval Societies Conf. (AIRS)*, pages 1–12, 2013.

[13] A. Moffat. Batch evaluation metrics in information retrieval: Measures, scales, and meaning. *IEEE Access*, 10:105564–105577, 2022.

[14] A. Moffat and J. Zobel. Rank-biased precision for measurement of retrieval effectiveness. *ACM Trans. on Information Systems*, 27(1):2.1–2.27, 2008.

[15] A. Moffat, P. Bailey, F. Scholer, and P. Thomas. Incorporating user expectations and behavior into the measurement of search effectiveness. *ACM Trans. on Information Systems*, 35(3):24.1–24.38, 2017.

[16] A. Moffat, J. Mackenzie, P. Thomas, and L. Azzopardi. A flexible framework for offline effectiveness metrics. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 578–587, 2022.

[17] F. M. Nardini, R. Trani, and R. Venturini. Fast approximate filtering of search results sorted by attribute. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 815–824, 2019.

[18] S. E. Robertson, E. Kanoulas, and E. Yilmaz. Extending average precision to graded relevance judgments. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 603–610, 2010.

[19] T. Sakai and Z. Zeng. Retrieval evaluation measures that agree with users' SERP preferences: Traditional, preference-based, and diversity measures. *ACM Trans. on Information Systems*, 39(2):14:1–14:35, 2021.

[20] M. D. Smucker and C. L. A. Clarke. Time-based calibration of effectiveness measures. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 95–104, 2012.

[21] N. V. Spirin, M. Kuznetsov, J. Kiseleva, Y. V. Spirin, and P. A. Izhutov. Relevance-aware filtering of tuples sorted by an attribute value via direct optimization of search quality metrics. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 979–982, 2015.

[22] N. Su, J. He, Y. Liu, M. Zhang, and S. Ma. User intent, behaviour, and perceived satisfaction in product search. In *Proc. Conf. on Web Search and Data Mining (WSDM)*, pages 547–555, 2018.

[23] A. Trotman and V. Kitchen. Quality metrics for search engine deterministic sort orders. *Information Processing & Management*, 59(6):103102, 2022.

[24] A. F. Wicaksono and A. Moffat. Modeling search and session effectiveness. *Information Processing & Management*, 58(4):102601, 2021.

[25] F. Zhang, Y. Liu, X. Li, M. Zhang, Y. Xu, and S. Ma. Evaluating web search with a bejeweled player model. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 425–434, 2017.

[26] F. Zhang, J. Mao, Y. Liu, W. Ma, M. Zhang, and S. Ma. Cascade or recency: Constructing better evaluation metrics for session search. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 389–398, 2020.

[27] F. Zhang, J. Mao, Y. Liu, X. Xie, W. Ma, M. Zhang, and S. Ma. Models versus satisfaction: Towards a better understanding of evaluation metrics. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, pages 379–388, 2020.