Improving Product Search with Session Re-Rank

a Walmart data mining project

Charles Celerier (cceleri), Bill Chickering (bchick), and Jamie Irvine (jirvine)

June 8, 2013

Walmart.com maintains an online catalog of over 2M products. Consequently, enabling users to quickly find products that conform to their specific needs and tastes is especially challenging. Given the difficulty of its task, Walmart.com's product search engine does an impressive job in interpretting the user-provided query and rapidly returning relevant results. Yet, there remains highly significant information that is not fully leveraged. The details of a user's online shopping session are indicative of a user's intent and compliment—indeed, provide context for—the user-provided query. In this report we describe and analyze a ranking scheme we call *Session Re-Rank* (*SRR*) that can potentially induce a large increase in both click-through-rates and conversions on the first page of query results.

1 The Technique

SRR works by comparing previously clicked items with the top N items returned by the search engine in reponse to a query. Items to be shown that are sufficiently similar to previously clicked items are promoted. The extent (i.e. number of positions) of the promotion for a particular item is a function of its similarity to previously clicked items, its original position, and the promotions of other items.

The similarity between an item to be shown and a previously clicked item is determined within five distinct vector spaces: *click-space*, *cart-space*, *query-space*, *title-space*, *item-space*. The non-unique representation of an item within each of these spaces may be thought of as a binary vector or a set of objects. (MapReduce jobs process historic query data to construct indexes whose keys are itemids and values are lists of the appropriate objects. Great care went into ensuring that index entries can be accessed in $\mathcal{O}(1)$ and that two entries can be merged to compute their intersection or union in linear time.) The similarity $J_s(A, B)$ of two items, A and B, within a particular space s is determined using Jaccard similarity. Similarities within particular spaces are then weighted and summed to determine the composite similarity

$$S(A, B) = \sum_s C_s(J_s(A, B))^{\alpha_s}, \tag{1}$$

where C_s and α_s are tuning parameters. The score σ attributed to an item to be shown is then the summation of composite similarities between itself and all previously clicked items plus the click-through-rate (CTR) Γ_i of the item's original position i

$$\sigma = \sum_{B \in P} S(A, B) + \Gamma_i,$$

where P is the set of previously clicked items.

2 Similarity Spaces

The premise behind *click-space* is that two items are similar if they are both clicked within the same online shopping session. The dimensions, or objects, of this space are therefore past user-sessions. The *clicks-index* for the data presented in this report was contructed using approximately half of the Walmart provided data, or about 60M queries (about 120M page views).

Cart-space is based on the notion that two items are similar is they ever appear in a shopping cart together. The objects of this space are therefore shopping carts. The *clicks-index* for the data presented in this report was contructed using approximately half of the Walmart provided data.

Items are also considered similar if they appear in a query together. The objects of *query-space* are therefore queries. We make a distinction, however, between *user-queries* and *unique-queries*. The former are the well-defined entities within the raw Walmart data. The latter is an abstraction based on the notion that multiple *user-queries* can correspond to a single *unique-query*. To derive *unique-queries* from our data, we cluster *user-queries* as follows: two *user-queries* with the same search attributes (e.g. category or price filters) are considered the same *unique-query* if the strings constructed by concatenating the space-seperated, stemmed (we use the Python stemming.porter2 module), forced to lower-case terms from each of their rawqueries are equal. We point out that while we achieved better results with this policy compared to simply using *user-queries*, we have no reason to believe that this is the ideal way to cluster queries for use within *SRR*. Indeed, we believe one way to improve *Session Re-Rank* is to optimize the query clustering policy.

Title-space is straightfoward: each item is associated with a set of terms from its title. We ignore case, but at present do not stem, discard stop words, or weight terms in any way.

Finally, the structure of *item-space* is unique because it involves a level of indirection. The premise here is that if items A and B are clicked in a single user-session and items A and C are clicked in another user-session, that items B and C are similar because they have item A in common. In this way, a large number of relationships between items is created. *Item-space* resembles *click-space* in that if two items are clicked during a single session, they will have nonzero similarity. It differs from *click-space* in two key respects, however. First, items that have historically never been clicked in the same session can have nonzero similarity if they were each clicked with a common third item. Second, if items are clicked together in many sessions this will increase their Jaccard similarity in *click-space* but not in *item-space*.

3 The Data

Walmart.com has generously supplied us with a large dataset consisting of about 250M pageviews comprising about 120M query results which occurred over about 30 days. The data includes the user-provided rawqueries together with search attributes, visitorIds and sessionIds, shown items, clicked items, which items were placed in a shopping cart, and which items were ultimately purchased. In addition, they have provided detailed item information including title, description, category, and other details. The query data was randomized with respect to search time and then segregated into three disjoint sets. The first set, which consists of about half of the data, was re-structured into indexes that form four of the similarity spaces (*click-space*, *cart-space*, *query-space*, and *item-space*) we use to identify relationships between items in realtime (the remaining similarity space, *title-space*, was compiled separately using the provided item data). The second set, which consists of less than 5% of the data, was used for testing and optimization allowing us to refine our technique and tune its parameters. And the third set, which includes about 10% of the data, was used in the experiments described and analyzed in this report.

4 The Technique vs The Experiment

An important distinction should be made between the *SRR* technique and the experiment described in this report. Both the technique and the experiment leverage the provided data—however, the experiment is a simulation and a limited one at that. A key limitation is that the provided query data is confined to what the user was actually shown. That is, the search engine may have identified several pages worth of results in response to a *user-query*, but our dataset consists only of those pages actually seen by the user. Meanwhile, the concept behind the *SRR* technique calls for a search engine to deliver to the algorithm the top N items in response to a *user-query independent of the number of items ultimately shown to the user*. As a consequence, it is difficult, if not impossible, to simulate our technique using shown query results that are truncated because a user only viewed one or two pages. Even more generally, the use of historic data to demonstrate the consequences of a online ranking algorithm is intrinsically limited by the fact that one cannot be certain how users would have behaved if presented with different results. Nonetheless, we have done our best to conduct the most fair and informative experiment and analysis.

5 The Experiment

The goal for the experiment is to simulate *SRR* using the provided historical query data, which is limited to what users were actually shown. Since an online implementation of our technique would receive the top N items from the search engine for re-ranking prior to showing any results to a user, the final ranking would be independent of the total number of items actually shown (e.g. the number of page views requested by a user). Our test set χ therefore consists solely of queries where either all items in the query resultset or at least $N = 100$ were shown to the user. For example, if the search engine found only 13 items in response to a query, all of these items were shown to the user on a single page. In this case, we have the complete query resultset and can therefore determine how *SRR* would have reordered the shown items. At the same time, if more than $N = 100$ were shown to the user, we can determine the reordering regardless of whether the query resultset is truncated since *SRR* only considers and re-ranks the first $N = 100$ items.

To construct χ we therefore must discard all queries with a number of shown results less than $N = 100$ that are also divisible by 16. The reason for this is that Walmart.com provides two options for the number of items shown per page: 16 or 32. Thus, by performing the experiment on this subset of the data we precluded queries where the top N items are not available to our algorithm. The choice of $N = 100$, meanwhile, is somewhat arbitrary and was made by balancing our desire for a large test set with our desire to use a value comparable to what would be appropriate for an online implementation. It is therefore quite possible that a larger value of N (e.g. 1000) would achieve better results in the actual online scenario.

While we must constrain χ in this way given the nature of the available data, we stress that this subset is certainly biased with respect to queries in general. For starters, queries with short resultsets are more likely to have all of their resultset seen by a user, and therefore, are more likely to be included in χ. It is not clear, however, if this particular bias tends to under- or overestimate the effectiveness of *SRR* since, as we will show, it is more effective on longer query results. Similarly, highly qualified queries—e.g. through the use of category or price filters—tend to have shorter resultset, and hence, are more likely to be included in χ.

Just as interesting are the ways in which the queries and resultsets of χ are not biased. In Fig. **??** we show click-through-rate (CTR) as a function of position of the original data (i.e. not re-ranked) for both χ and query results in general. As can be seen from the data, the quantity and distribution of clicks within χ are essentially representative of those in general.

6 Metrics

A true test of the effectiveness of *SRR* would require online A/B testing. In the meantime, we can simulate the effect of *SSR* by running it on historical data and examining the new positions of clicked and purchased items. Assuming the user would have clicked or purchased the same items in this new ordering, we can compare the distribution of clicks in the original ranking to that of yielded by *SRR*.

To compare two rankings of shown items, we focus on three key metrics. Our primary metric is the first-page CTR $\mathcal{C}$, defined as the likelihood an item presented on the first page receives a click. As noted above, 75% of all queries are only one page long. This shows the importance of bringing desirable items to the first page and it motivates our focus on $\mathcal{C}$. We calculate this metric by counting the number of items in first-page positions that were clicked and dividing by the number of total items in first-page positions. More formally

$$\mathcal{C} = \frac{\sum_{q \in Q} \sum_{i=1}^{L_q} \mathbb{1}\left\{click@i \wedge i \leq 16\right\}}{\sum_{q \in Q} \sum_{i=1}^{L_q} \mathbb{1}\left\{i \leq 16\right\}}.$$

Our second metric is the purchasing rate of items on the first page $\mathcal{P}$. This is similar to $\mathcal{C}$, except here we consider purchases per first-page item instead of clicks. It is calculated as the number of items in first-page positions that were purchased divided by the total number of items in first-page positions. The importance of position for purchases is even stronger than that for clicks. While 70% of all clicks were presented on the first page, that number is 85% for purchases. Formally, we have

$$\mathcal{P} = \frac{\sum_{q \in Q} \sum_{i=1}^{L_q} \mathbb{1}\left\{purchase@i \wedge i \leq 16\right\}}{\sum_{q \in Q} \sum_{i=1}^{L_q} \mathbb{1}\left\{i \leq 16\right\}}.$$

The first two metrics focus on whether or not a desirable item was presented on the first page. To obtain a more granular picture of where desirable items are positioned, we also compute a third metric which we call *click-position score* $\mathcal{S}$. Somewhat similar to normalized discounted cumulative gain (NDCG), which is a common metric of search engine results, this score weighs the value of a clicked item by its position, giving higher weights to items closer to the top. For $\mathcal{S}$, the weight given to a click in position i is the CTR at position i Γ_i. In this way, we equate how often users click on a certain position to with how valuable it is to put a desirable item there. Formally, we define *click-position score* as

$$\mathcal{S} = \frac{1}{|Q|} \sum_{q \in Q} \sum_{i=1}^{L_q} \mathbb{1}\left\{click@i\right\} \Gamma_i$$

.

These metrics give a sense of how successful a ranking scheme is. Each one looks at a slightly different aspect of the ordering. Indeed, optimizing for one metric does not necessarily optimize for the others. We focus our optimizations and primary analysis on $\mathcal{C}$ as we believe it is the simplest and has the clearest impact to overall CTRs.

7 Results

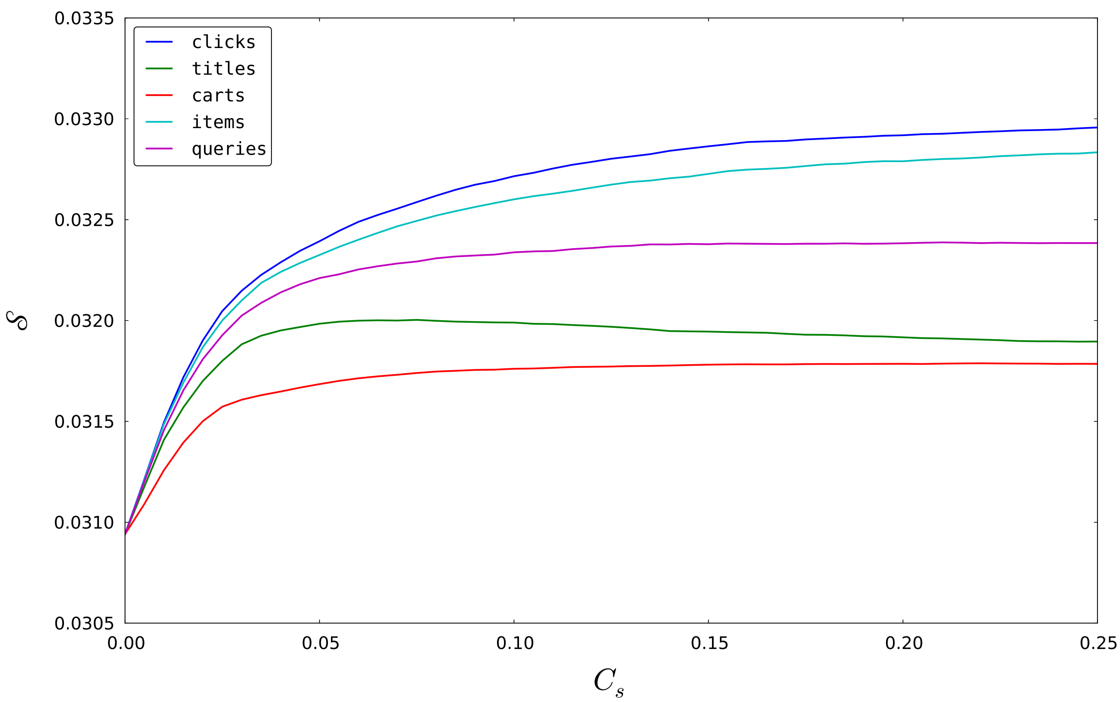

Figure 1: average clicks position score

8 An Example

To illustrate the efficacy of our technique, we present a real query example. The only fictitious part of the example will be our shopper's name, David.

David is interested in the "Primo Ceramic Crock Water Cooler with Stand" and clicks on this item during his session. Sometime later he navigates to the "Grocery →Beverages →Water" category and searches for "water". He is presented with 300+ results and clicks 90th item, a 3 liter jug of water.

The top six original results are compared to the *SRR* results in Table 1 where the first and third column represent the original ranking presented to David. Note that we rerank the 90th item from the original results to be the 3rd item in the *SRR* results. Tables 2 and 3 show the similarity scores from each index for each item in the two orderings compared to David's lone previously clicked item. In each of these tables, the first column correponds to Table 2 and the second column is the similarity metric from Equation (1) using our optimized tuning parameters.

The similarity scores from *title-space* are easily calculated by hand. We will show the calculation for the *item-space* similarity for the 90th item in the original ranking. By referring to the corresponding index for *item-space*, we can recall that the 90th item was clicked in the same query with 39 different items and the previously clicked item was clicked in the same query with 455 different items. The titles of the 13 items found in common are:

Great Value: Distilled Water, 1 Gal

Original Ordering		*SRR* Ordering	
1	Great Value Purified Water, 24ct	1	Great Value Purified Water, 24ct
2	Nestle Waters Bottled Spring Water, 24ct	2	Nestle Waters Bottled Spring Water, 24ct
3	Voss Water, 16.9 oz (Pack of 24)	90	Arrowhead Mountain Spring Water, 3 l
4	Clear American Cherry Sparkling Water, 1 l, 12pk	63	Great Value: Distilled Water, 1 Gal
5	Clear American Water, 1 l, 12ct	38	Arrowhead Mountain Spring Water, 2.5gal
6	Clear American Peach Sparkling Water, 1 l, 12ct	8	Clear American Mandarin Orange Sparkling Water, 1 l, 12pk

Table 1: Original Ordering vs. *SRR* Ordering for “water” query

	S	**CTR**	**Clicks**	**Items**	**Carts**	**Queries**	**Titles**
1	0.943 75	0.075 40	0.002 28	0.017 96	0.000 00	0.010 47	0.090 91
2	1.202 11	0.039 00	0.003 89	0.025 15	0.000 00	0.025 48	0.083 33
3	0.357 06	0.025 40	0.000 00	0.022 79	0.000 00	0.005 55	0.076 92
4	0.752 68	0.019 50	0.001 68	0.023 39	0.000 00	0.001 63	0.066 67
5	0.736 73	0.015 30	0.001 73	0.016 08	0.000 00	0.003 90	0.066 67
6	0.174 83	0.012 90	0.000 00	0.009 54	0.000 00	0.000 79	0.066 67

Table 2: Index similarity scores of the top six original results to
“Primo Ceramic Crock Water Cooler with Stand”

	S	**CTR**	**Clicks**	**Items**	**Carts**	**Queries**	**Titles**
1	0.943 75	0.075 40	0.002 28	0.017 96	0.000 00	0.010 47	0.090 91
2	1.202 11	0.039 00	0.003 89	0.025 15	0.000 00	0.025 48	0.083 33
90	0.953 10	0.000 23	0.003 57	0.027 03	0.000 00	0.000 92	0.083 33
63	0.949 54	0.000 49	0.002 61	0.033 83	0.000 00	0.003 86	0.083 33
38	0.937 69	0.000 91	0.003 44	0.033 14	0.000 00	0.000 00	0.090 91
8	0.896 71	0.010 00	0.003 55	0.024 86	0.000 00	0.000 00	0.066 67

Table 3: Index similarity scores of the top six *SRR* results to
“Primo Ceramic Crock Water Cooler with Stand”

Nestle Waters Bottled Spring Water, 24ct

Primo Mineral Water, 5 gal

Deer Park Sumo Bottle Natural Spring Water, 3l

Arrowhead Mountain Spring Water, 3l

PUR Advanced Faucet Water Filter Vertical - Chrome

Ozarka Natural Spring Water

Formula 409 All Purpose Lemon Scented Cleaner, 32 fl oz

Great Value Spring Water, 1 gal

Nestle Pure Life Purified Water, .5l, 35pk

Arrowhead Mountain Spring Water, 2.5gal

Primo Ceramic Crock Water Cooler with Stand

Augason Farms Emergency Water Storage Kit

We then calculate the similarity of the 90th item and the previously clicked item in *item-space* to be $13/(455+39-13) = 0.0270$.

We believe this example illustrates how our technique can form subtle relationships between items based on historical user session data. In this case, we have a shopper who had an interest in a water cooler and subsequently made a query for "water". If the shopper had been in a brick-and-mortar store, he likely would have been directed to the section of the store selling water jugs to use with the cooler he had picked up. In this case, *SRR* recognized David's previously click, related the water cooler to large water jugs, and showed David the water jug he was looking for all along.