Integration: Early Recommendation Quality

Overview

It is important that those on the project team responsible for testing/validating placements and recommendations throughout the Integration and Listen Mode stages, are aware of the context in which strategies operate during this early period. It is equally important that the right expectations are also set with the wider internal audience who may be exposed to recommendations once in Live Mode and may have questions about what they see.

Intended Audience

This guide assumes that its readers:

- Are experienced web or application developers
- Are familiar with JavaScript
- Have a background in the ecommerce/retail industry
- Understand how to populate functions with dynamic page-specific values

What are the benefits of this feature?

By understanding the concepts and how the rules and strategies work, retailers will be better able to understand the results of early recommendations.

What are the drawbacks of this feature?

This set of documents are only guides to help you validate your instrumentation. The code provided are samples to help showcase the proper syntax and parameters.

How It Works

At a high level, the JavaScript "stub", which lives in the HTML of a retailer's website, populates a number of product- and customer-related fields in a JavaScript object, transmits the fields to the RichRelevance server using HTTP or HTTPS, and receives a dynamically-generated JavaScript file that modifies the eCommerce site to display relevant product recommendations.

Common questions/comments are:
• Should I be seeing these products in recommendations?
• These recommended products are random/irrelevant.
• Why does this recommendation include products from categories other than the one I am currently in?
• Why am I seeing generic ‘top seller’ recommendations when I thought we were implementing a personalization engine?
• How long will it take before customers start to see relevant recommendations?

This document aims to address some of these common concerns regarding the quality/relevance of recommendations during the early stages of an implementation project, in particular during Listen Mode, but also during the early weeks of Live Mode.

**Strategy Configuration**

**Best Practice**

The default strategy configuration is based on a best-practice implementation for which strategies are appropriate for each page type. This best-practice has been derived from extensive evaluation of strategy performance across many RichRelevance client sites. This configuration is the starting point for all implementations, however in time this will be customized based on the reality of customer behavior on your site and your merchandising objectives.

**Enabled Strategies**

Each page type will have multiple strategies enabled by default, drawing from several strategy classes: cross-sell; search; similar items; top sellers, as well as user-specific. Although the ‘enabled’ list for each page type may include some product or shopper-specific strategies, there will also be strategies of other classes and best-practice is to always include some generic strategies in the case that the other strategy classes do not qualify (see below). This is of particular concern when a page contains multiple placements, as sufficient strategies must be enabled for a given page type to ensure that enough qualify to fill all the placements on the page.

**Real-Time Decisioning**

**Strategy Qualification**

Different strategies have different pre-requisites (some require particular instrumentation on a page, others require particular view or purchase history to be available) and so will not always qualify.
King of the Hill

At run-time, the personalization engine (i.e. the decision engine that determines which strategy becomes ‘King of the Hill’ in any scenario) will determine which of the enabled strategies for that page type qualify for display to that visitor, at that point in time and given the data available. Merchandising rules and filters are then applied and from the surviving candidates in the list, the engine will determine the ‘King of the Hill’ based on historic click-through rate of that strategy.

Data Availability

Strategy Performance Data

During Listen Mode, strategy performance data is not available because rec placements are not yet being displayed to customers. Once Recs are live (aka Display Mode) and the performance data starts to become available, the engine will start to favor those strategies that perform better (i.e. generate more click-through). Also, given the limited data available at this stage, there will often be a high variability in the strategies being used.

Data Feed

Product names, descriptions and other attributes that may appear in placements (as well as pricing, by default) are determined by the most recent product catalog feed. If incorrect values are seen in recommendations, this is likely to be due to incorrect data in the feed or simply that the updated feed has not yet been uploaded/processed.

Modeling

Modeling Approach

During Listen Mode and the first two weeks of Display Mode, RichRelevance uses a modeling approach called ‘weighted randomization’, which means that all enabled (and qualified) strategies are given an equal chance to perform (so a number of the placements shoppers will see will be ‘test’ rather than ‘best’ strategies and may appear to be less relevant than the ‘best’ strategies). After two weeks in Display Mode (typically a long enough period for the engine to learn the strategies relative strengths and weaknesses), your Deployment Manager with will update the site configuration to use the ‘adaptive window’ modeling approach. This uses the least amount of testing possible to maintain a level of confidence that the engine is selecting the highest performing strategies (adaptive, because the higher trafficked page types will require a lower number of tests to maintain statistical relevance). Under ‘adaptive window’ the shopper will now be seeing the highest performing strategy the highest percentage of time.


Updated: Tue, 07 May 2019 14:57:37 GMT
Powered by
Automated Strategy Testing

In order to avoid the best performing strategy on day one simply finding its way to the top and just staying there permanently, the engine constantly tests the different enabled strategies to see how they perform – effectively giving other strategies the chance to become performers themselves. This happens, and will continue to happen, automatically in the background all the time. While the engine is still learning and strategies are still building a performance history, there may be a higher proportion of ‘test’ strategies being displayed.

Testing

When testing ‘recommendation quality’, be aware that a number of strategies are based on view history and will therefore include product types that the tester has previously viewed. For a valid test, the tester should build a realistic view history of someone who is shopping on the site. Clicking around a number of random categories in order to view recommendations in different areas of the site is an atypical view history, so the recommendations seen will not necessarily be representative of what a real shopper would see. By building a more realistic view history and refreshing the item page, you will likely see the ‘relevance’ of recommendations increase.

Some strategies are seeded by views/purchases of multiple items, potentially across categories, which can result in a heterogeneous group of products in a placement. Although these strategies and resulting products may not appear relevant at first, they can perform well and should not be discounted without first analyzing performance.

Ongoing Optimization

Within a week or two of transitioning to Display Mode, your Deployment Manager will conduct an initial review of strategy performance. At that time, non-performing strategies can be de-activated and, if necessary, back-filled with other strategies to see how well they perform. Following appropriate analysis of performance, there are also options to ‘prefer’ specific strategies on a placement-by-placement basis if required, as well as utilize the merchandising controls to apply rules and filters in support of your objectives.