

LAB: Building a Recommender

Prerequisites

- Register for Kaggle
 - Instructions here: [Registering for Kaggle](#)
- LAB: Spark and Data Prep
- LAB: Working with Signals

Getting Started

This lab assumes that you are using an AWS virtual machine provided by Lucidworks Training. If this is not the case, your filepaths and IP addresses will vary significantly from those shown.

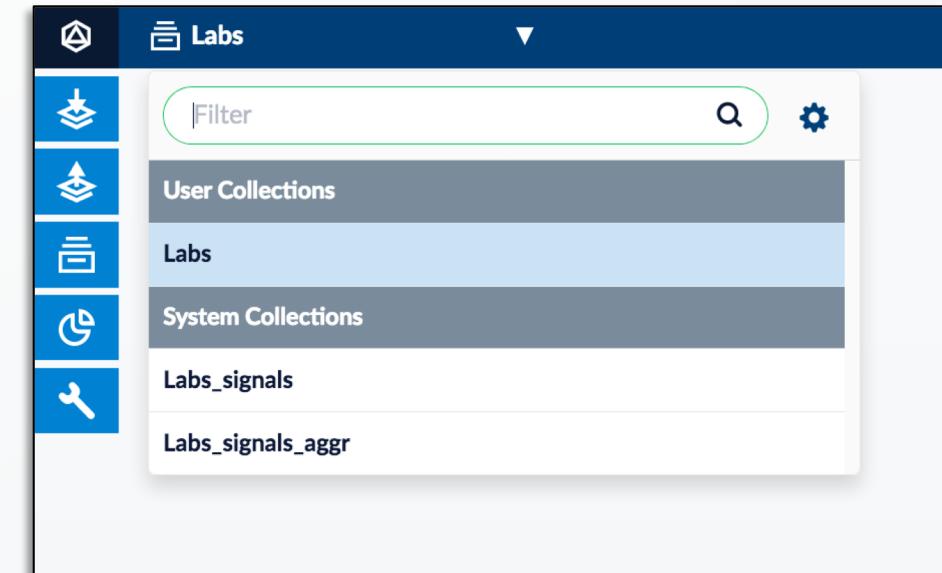
- In a bash/shell terminal, start Fusion

```
./fusion/4.0.1/bin/fusion start
```

- In a web browser, open Fusion Admin

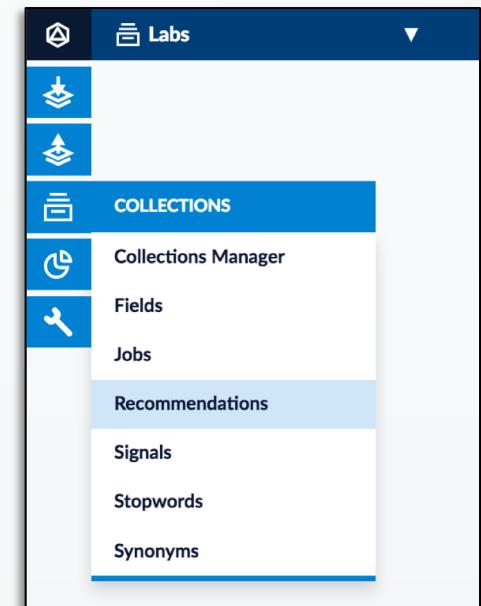
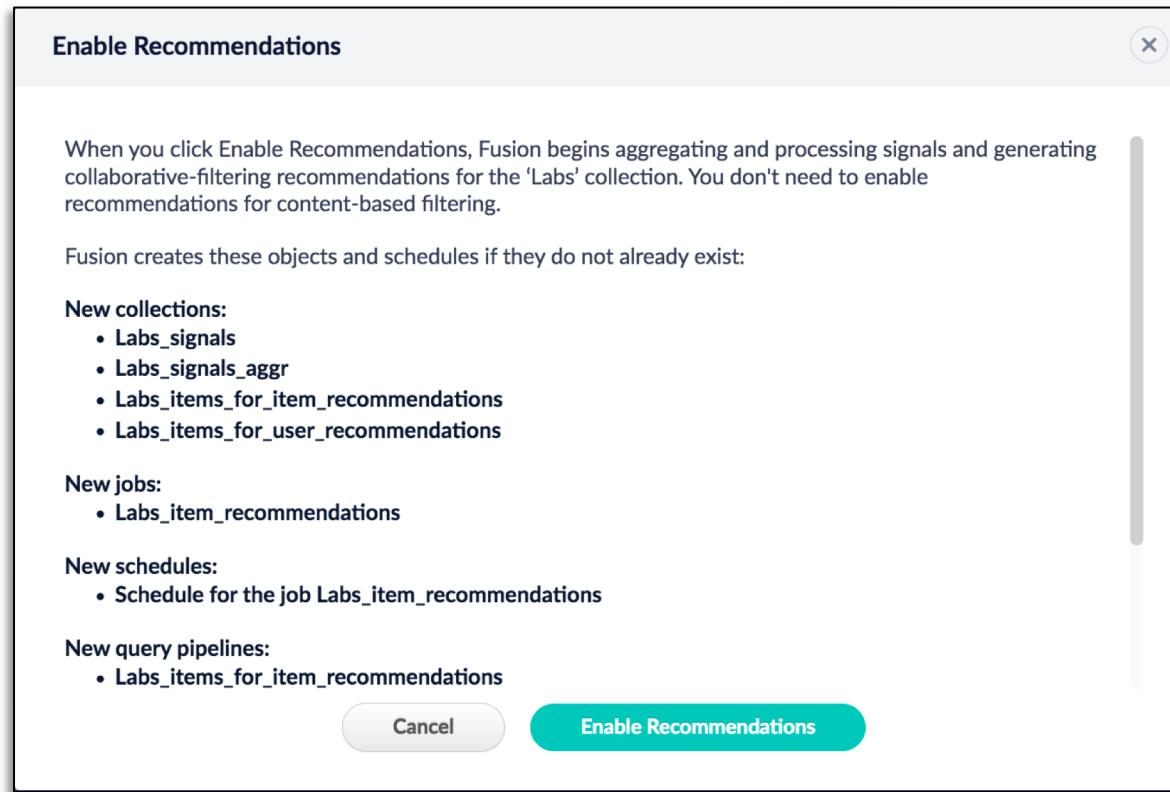
```
<your-vm-ip>:8764
```

- Enter your username and password
*The default is **admin** and **Lucidworks1***
- Click into the **Labs** Fusion App
- In the top left dropdown, change to the **Labs** collection



- In the left side menu, go to **COLLECTIONS > Recommendations**

Be sure to read and understand the dialog box that appears.



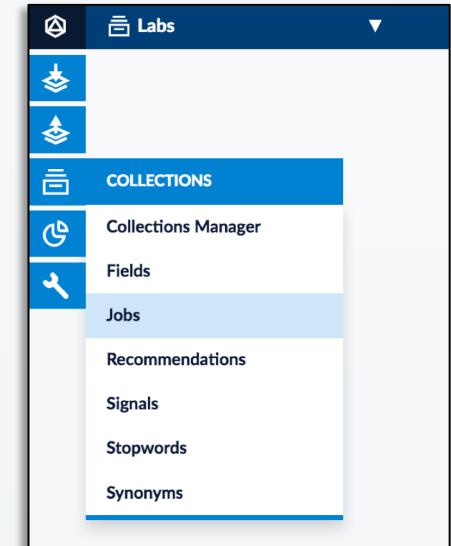
- Click **Enable Recommendations**

Configuring Recommendations

In this lab, we will set up **items-for-user** recommendations (i.e., “Recommended for you”), as well as **item-for-item** recommendations (i.e., “Customers who bought this also bought XYZ”), on the Best Buy catalog.

There are two main tasks involved in this. First, we will use existing signals to determine each past user’s level of interest in each item. Second, we will use these past user-item preferences to create a user-item matrix, which can be used both to recommend items to new users and recommend other items that tend to collocate with the one currently clicked.

- In the left side menu, go to **COLLECTIONS > Jobs**
- Select **Labs_item_recommendations**



This job is the one that creates the user-item matrix.

*If you click the **Run** button, you will see that it is scheduled to automatically run after the **Labs_user_item_preferences_aggregation** job, which is the one that calculates past user-item preferences.*

Jobs

Filter Add + Delete Run Job History

Job Id	Status
Baseline_vs_Signals_vs_QI-clickthro...	
Last run: Tue, Jul 17, 2018 at 03:10:23 PM -0400	
department_qi_model	
Last run: Tue, Jul 10, 2018 at 08:56:59 AM -0400	
Generate_clicks	
Last run: Tue, Jul 17, 2018 at 02:27:10 PM -0400	
Labs_click_signals_aggregation	
Last run: Wed, Jul 18, 2018 at 10:01:39 AM -0400	
Labs_head_tail	
Last run: Tue, Jul 17, 2018 at 02:31:46 PM -0400	
Labs_item_recommendations	
Last run: Never run	

Labs_item_recommendations

Train a collaborative filtering matrix decomposition recommender using Least Squares (ALS) to batch compute user recommendations and item s

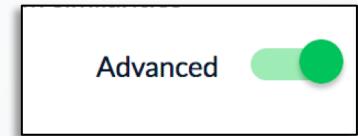
* Spark Job ID: Labs_item_recommendations

* Recommender Training Collection: Labs_signals_aggr

Items-for-users Recommendation Collection: Labs_items_for_user_recommendations

Item-to-item Similarity Collection: Labs_items_for_item_recommendations

- Toggle the **Advanced** switch at the top right
- Fill out the job parameters according to the following table:



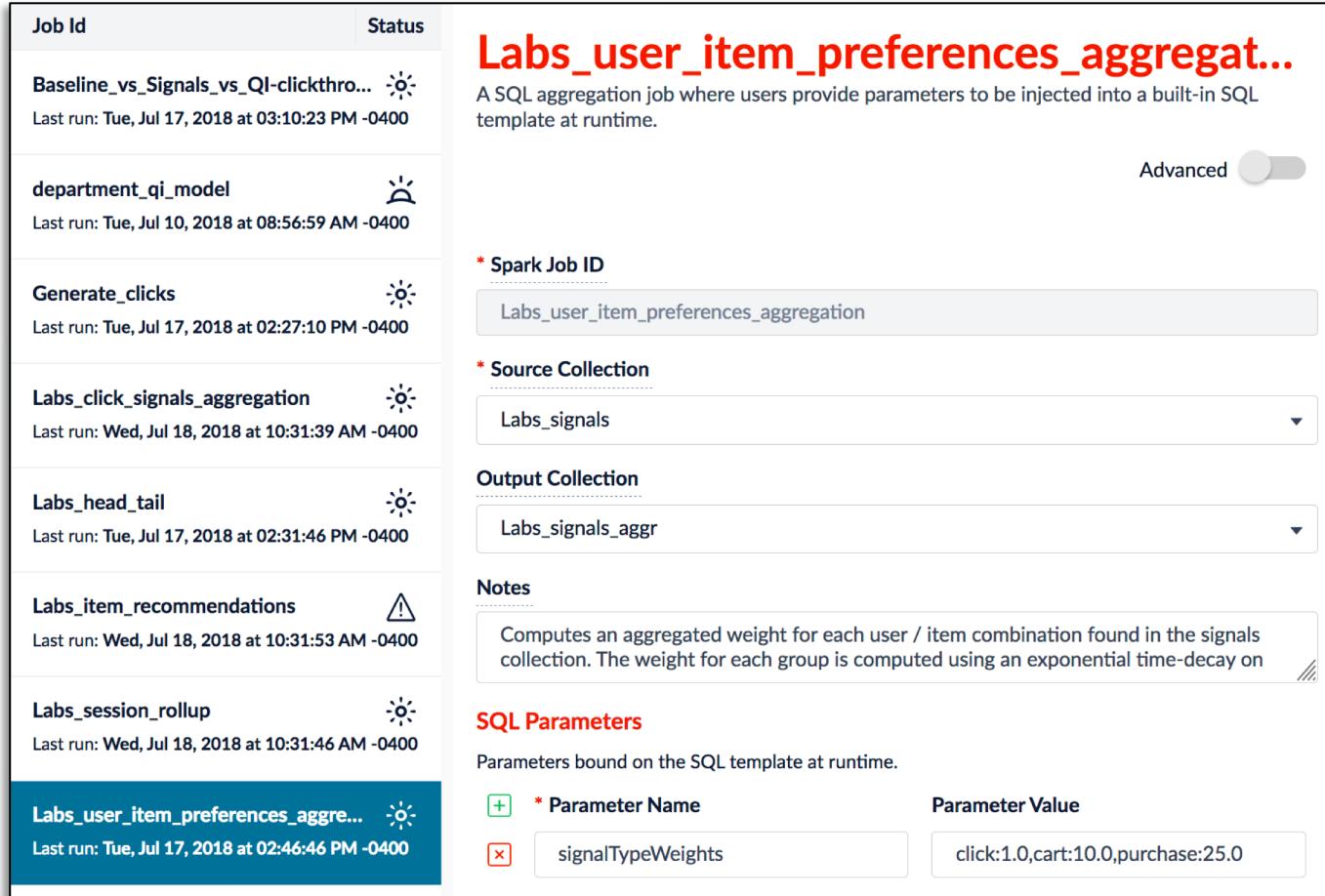
Parameter	Value	Explanation
Training Data Sampling Fraction	0.05	<i>Downsampling makes the job finish faster. You would probably not do this in production, though, as this can degrade output quality.</i>
Item Metadata Fields	department	<i>Additional fields to add to the aggregation documents. These are not used in the matrix calculations, but can be used to fine-tune the time usage of recommendations.</i>
Item Metadata Collection	Labs	<i>Collection from which to pull the metadata fields</i>
Item Metadata Join Field	id	<i>Join key between aggregate document ID (default: <code>doc_id_s</code> in the <code>Labs_signals_aggr</code> collection) and metadata ID (in this case, from <code>Labs</code>)</i>

- Save the `Labs_items_recommendations` job. Do not run it yet!

- Open the `Labs_user_item_preferences_aggregation` job

As mentioned before, this job is the one that will calculate user-item preferences based on past behavior.

The output of this will be used by `Labs_item_recommendations` to create a user-item matrix



The screenshot shows the 'Recommender Jobs' interface with a list of jobs on the left and a detailed configuration panel on the right.

Job List:

Job Id	Status
Baseline_vs_Signals_vs_QI-clickthro...	⌚
Last run: Tue, Jul 17, 2018 at 03:10:23 PM -0400	
department_qi_model	⌚
Last run: Tue, Jul 10, 2018 at 08:56:59 AM -0400	
Generate_clicks	⌚
Last run: Tue, Jul 17, 2018 at 02:27:10 PM -0400	
Labs_click_signals_aggregation	⌚
Last run: Wed, Jul 18, 2018 at 10:31:39 AM -0400	
Labs_head_tail	⌚
Last run: Tue, Jul 17, 2018 at 02:31:46 PM -0400	
Labs_item_recommendations	⚠
Last run: Wed, Jul 18, 2018 at 10:31:53 AM -0400	
Labs_session_rollup	⌚
Last run: Wed, Jul 18, 2018 at 10:31:46 AM -0400	
Labs_user_item_preferences_aggre...	⌚
Last run: Tue, Jul 17, 2018 at 02:46:46 PM -0400	

Labs_user_item_preferences_aggregat...

A SQL aggregation job where users provide parameters to be injected into a built-in SQL template at runtime.

Advanced

*** Spark Job ID**: `Labs_user_item_preferences_aggregation`

*** Source Collection**: `Labs_signals`

Output Collection: `Labs_signals_aggr`

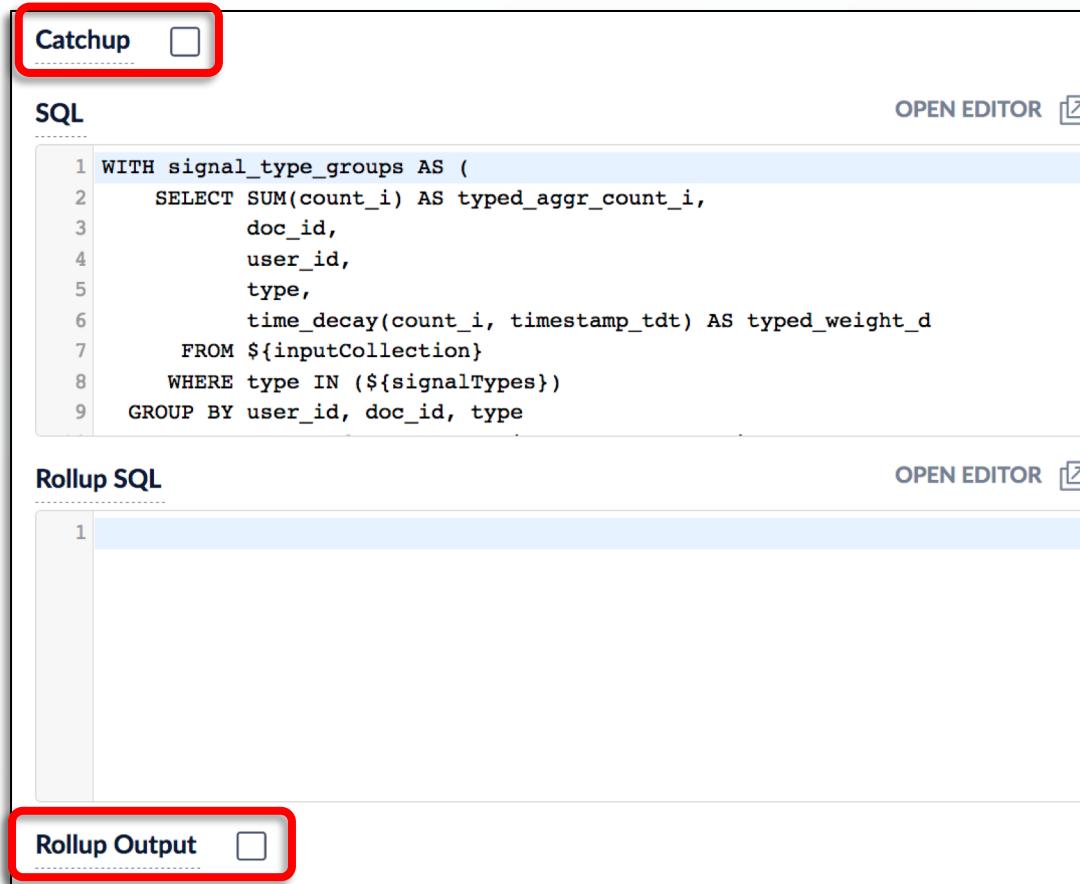
Notes: Computes an aggregated weight for each user / item combination found in the signals collection. The weight for each group is computed using an exponential time-decay on

SQL Parameters

Parameters bound on the SQL template at runtime.

Parameter Name	Parameter Value
+ * Parameter Name	
signalTypeWeights	click:1.0,cart:10.0,purchase:25.0

- Toggle the **Advanced** switch in the top right



The screenshot shows a job configuration interface with two main sections: SQL and Rollup SQL. The SQL section contains the following code:

```
1 WITH signal_type_groups AS (
2     SELECT SUM(count_i) AS typed_aggr_count_i,
3         doc_id,
4         user_id,
5         type,
6         time_decay(count_i, timestamp_tdt) AS typed_weight_d
7     FROM ${inputCollection}
8     WHERE type IN (${signalTypes})
9     GROUP BY user_id, doc_id, type
```

The Rollup SQL section has one line of code:

```
1
```

At the top of the interface, there is a "Catchup" checkbox and a "Rollup Output" checkbox, both of which are highlighted with red boxes.

- Uncheck the boxes **Catchup** and **Rollup Output**

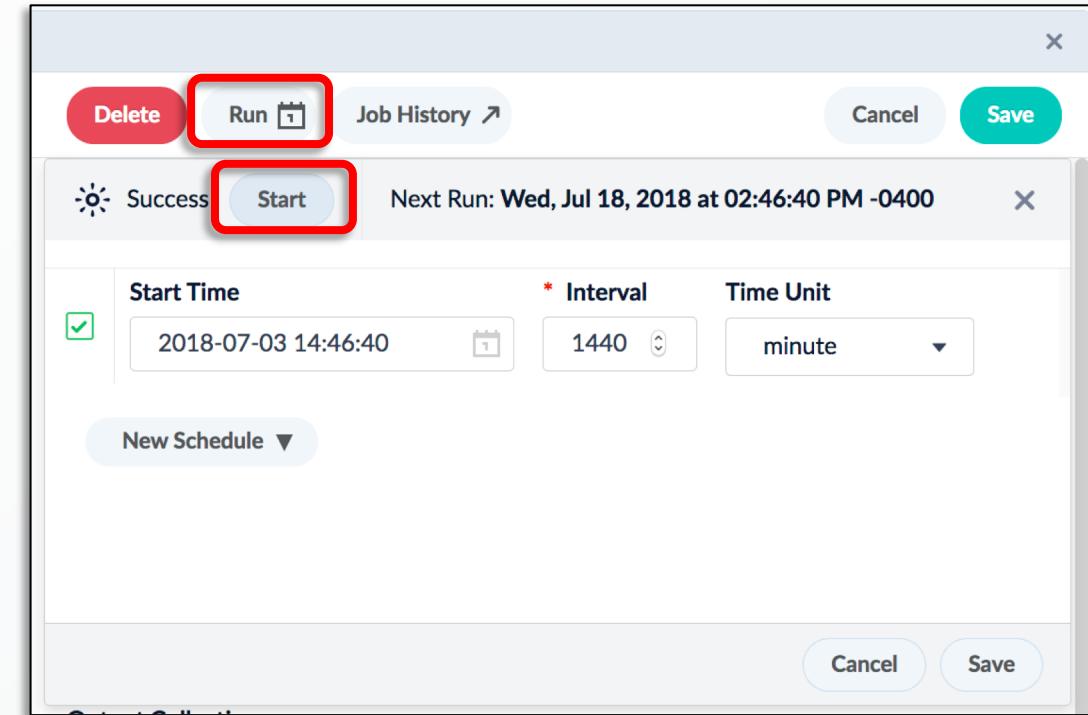
These parameters are only functional for subsequent runs of the job.

- Save the modified job

Running the Recommender Jobs

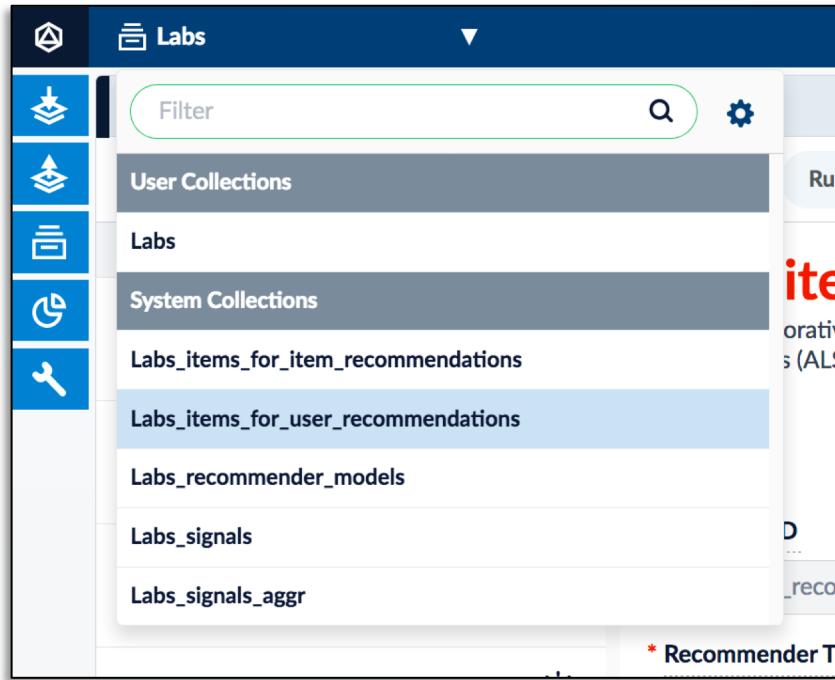
- Click Run > Start

The job should take ~30 seconds to finish. Once it does, the `Labs_item_recommendations` job will kick off. That job will take a bit longer; ~2 minutes.

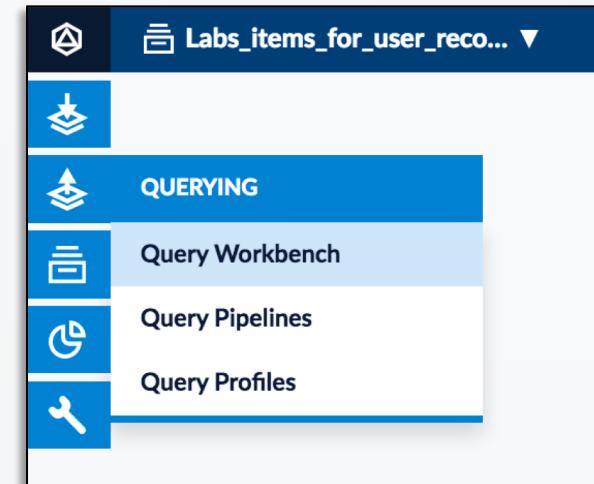


Verifying Successful Model Build

- In the top left dropdown of Fusion Admin, change to the **Labs_items_for_user_recommendations** collection



- In the left side menu, go to **QUERYING > Query Workbench**



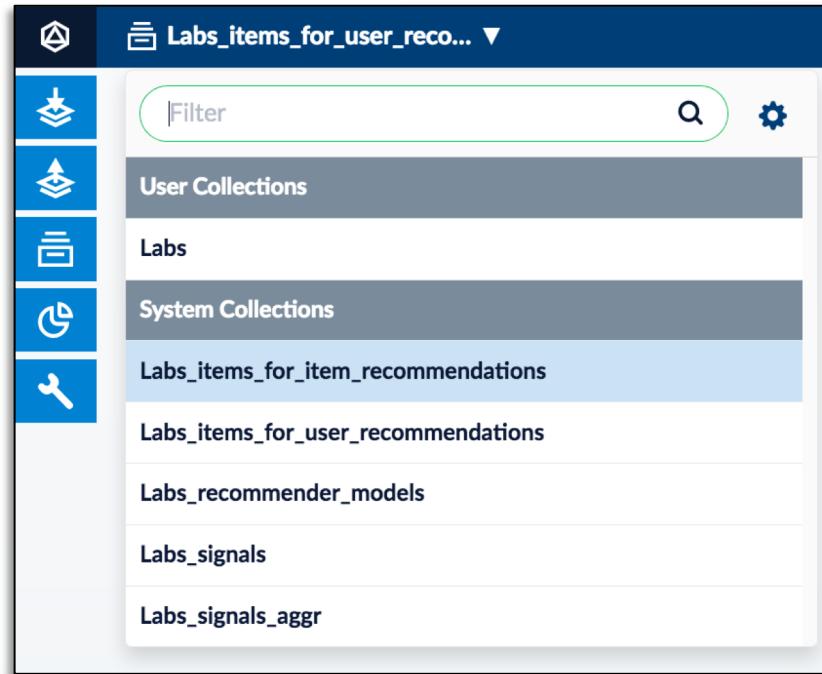
- Click **show fields** under any document

c2cb85bd-7679-4563-87c2-66aed9a64de6	
1251309	
Score:	1 hide fields
department	COMPUTERS
id	c2cb85bd-7679-4563-87c2-66aed9a64de6
itemId	1251309
jobId	164aefc7dddT5c052642
modelId	Labs_recommender
numInteractionsForItem	0
numInteractionsForUser	0
score	1
type	items-for-user
userId	d3b26eb7245cbc325ad8aaaf24d1c8214329411b
weight	0.09701470803776147
version	1606359972646486000

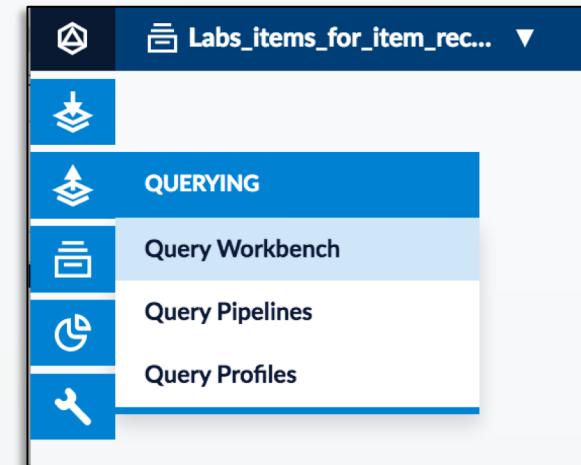
*Key among the fields here are **itemID** and **userID**, representing a single, unique user-item pairing. The **weight** field is a prediction of how well this item matches this user, based on the behavior of other users with similar activity.*

*Also note the **department** field. This was pulled in by the Metadata Join we requested in the **Labs_item_recommendations** job. This field can be used at query time; for example, to only recommend items from certain departments.*

- In the top left dropdown of Fusion Admin, change to the **Labs_items_for_item_recommendations** collection



- In the left side menu, go to **QUERYING > Query Workbench**



- Click **show fields** under any document

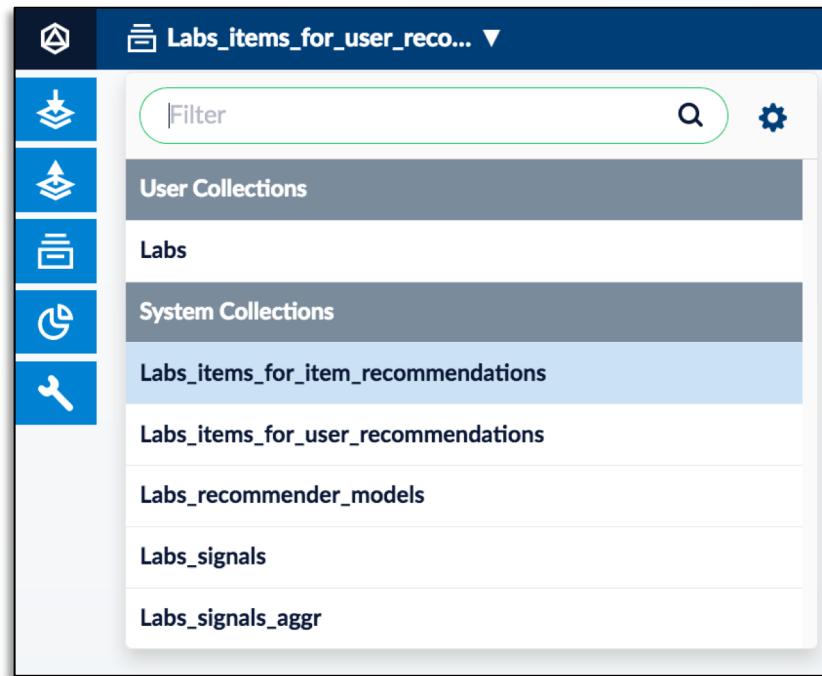
*Notable fields here are **itemId** and **otherItemId**, representing a single unique pairing of items. The **sim** field is a measurement of how “similar” these two items are, based on collocation in user activities.*

*Again, also note the **department** field from the Metadata Join.*

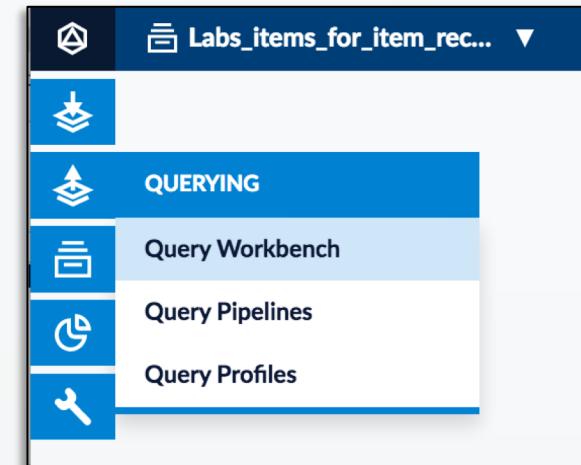
1b946685-72ae-4cab-a130-04003f5e1680	
2383054	↑ Boost Ø Block
Score: 1 hide fields	
department	COMPUTERS
id	1b946685-72ae-4cab-a130-04003f5e1680
itemId	2383054
jobId	164aefc7dddT5c052642
modelId	Labs_recommender
otherItemId	2969326
score	1
sim	0.7551533779761058
type	items-for-item
version	1606359980714229800

Testing Item-Item Recommendations

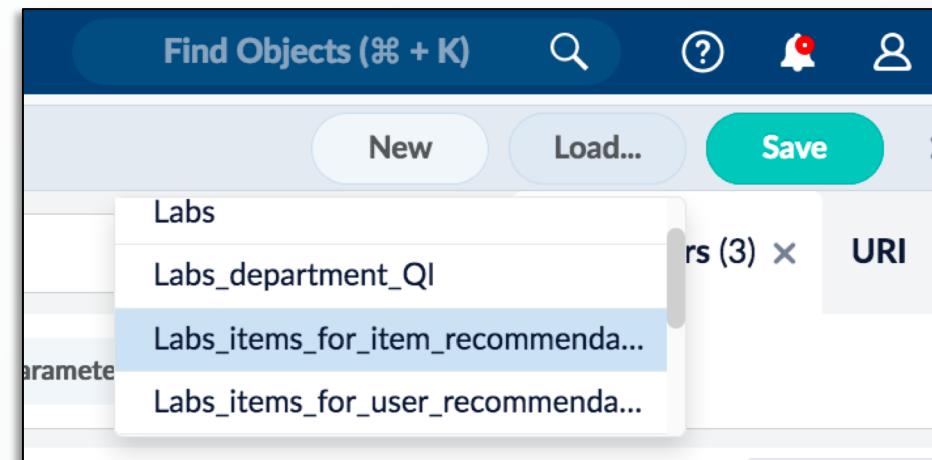
- In the top left dropdown of Fusion Admin, change to the **Labs** collection



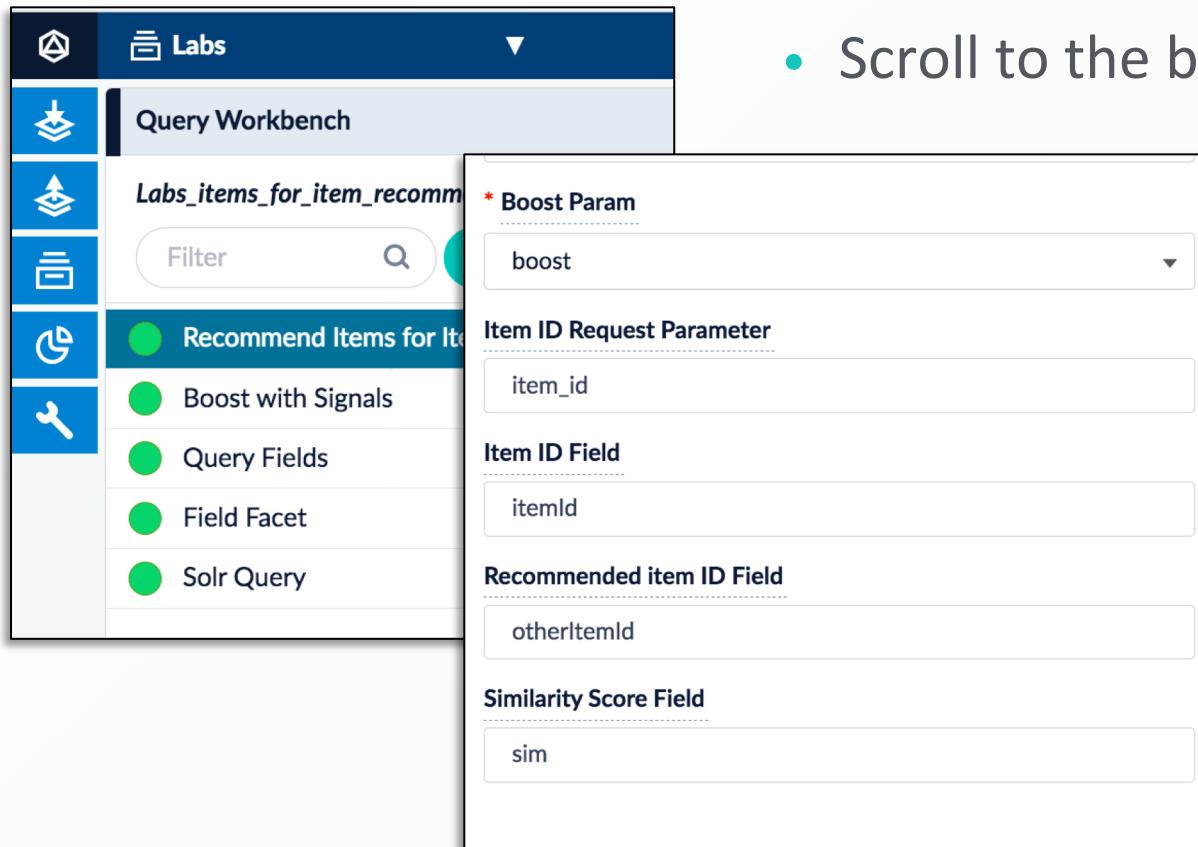
- In the left side menu, go to **QUERYING > Query Workbench**



- At the top right, click **Load**
- Select the pipeline **Labs_items_for_item_recommendations**



- In the Stages list, select **Recommend Items for Item**



The screenshot shows the Lucidworks Query Workbench interface. On the left, there's a sidebar with icons for Home, Labs, and a search bar. Below these are several stages listed: 'Recommend Items for Item' (selected), 'Boost with Signals', 'Query Fields', 'Field Facet', and 'Solr Query'. The main area is the 'Stage Editor pane' for 'Recommend Items for Item'. It contains the following fields:

- * Boost Param: boost
- Item ID Request Parameter: item_id
- Item ID Field: itemId
- Recommended item ID Field: otherItemId
- Similarity Score Field: sim

- Scroll to the bottom of the Stage Editor pane

*This stage implements a relevance boost for similar items, as long as a single item is “selected” via an **item_id** parameter attached to the query. This parameter will be populated, for example, by going to the details page for a given item.*

The last three parameters relate to the structure of the items-for-item documents we examined a moment ago.

- At the top of the page, click **Parameters > Edit Parameters**



- Add a parameter named **item_id** with value **2460117**

Parameter Name	Parameter Value
echoParams	all
debug	true
item_id	1945531

*As mentioned before, this parameter prompts the **Recommend Items for Item** stage to boost items similar to the one with doc id 2460117*

Testing Item-Item Recommendations

- Execute the query **id:2460117**

As this document is a SSD, items boosted by the recommender should be related to hard drives.

Parameter Name	Parameter Value
echoParams	all
debug	true
item_id	2460117

- Execute the query ***:***

These results are pretty good. Yours, however may differ, and may not be so good. This is due to the downsampling we did when building the model. For better results, use a larger sample!

Corsair - Force Series 3 60GB Internal Serial ATA III Solid State Drive for Laptops
This solid state drive features a 60GB capacity for storing important documents, special photos and more. The Serial ATA III interface allows for a simple connection to your computer.
Score: 15 [show fields](#) + Added to page

Corsair - Professional Gold Series 1200-Watt Power Supply
This power supply features reliable delivery of power for your work and gaming needs. The modular cabling system makes installation and customization simple, while short-circuit protection helps prevent damage to your computer and components.
Score: 14.8681 [show fields](#) + Added to page

Corsair - Graphite Series 600T Mid-Tower Gaming Case and AX850 Power Supply Bundle
This mid-tower gaming case features dual 200mm fans and a 120mm rear fan to keep your installed components cool. The Professional Series Gold AX850 power supply delivers plenty of power for your system.
Score: 14.8656 [show fields](#) + Added to page

Corsair - Gaming Series 700-Watt ATX CPU Power Supply
Power your gaming computer with this power supply that features 700 watts of output power for enhanced performance. The 0.99 active power factor correction delivers reliable power.
Score: 12.4673 [show fields](#) + Added to page

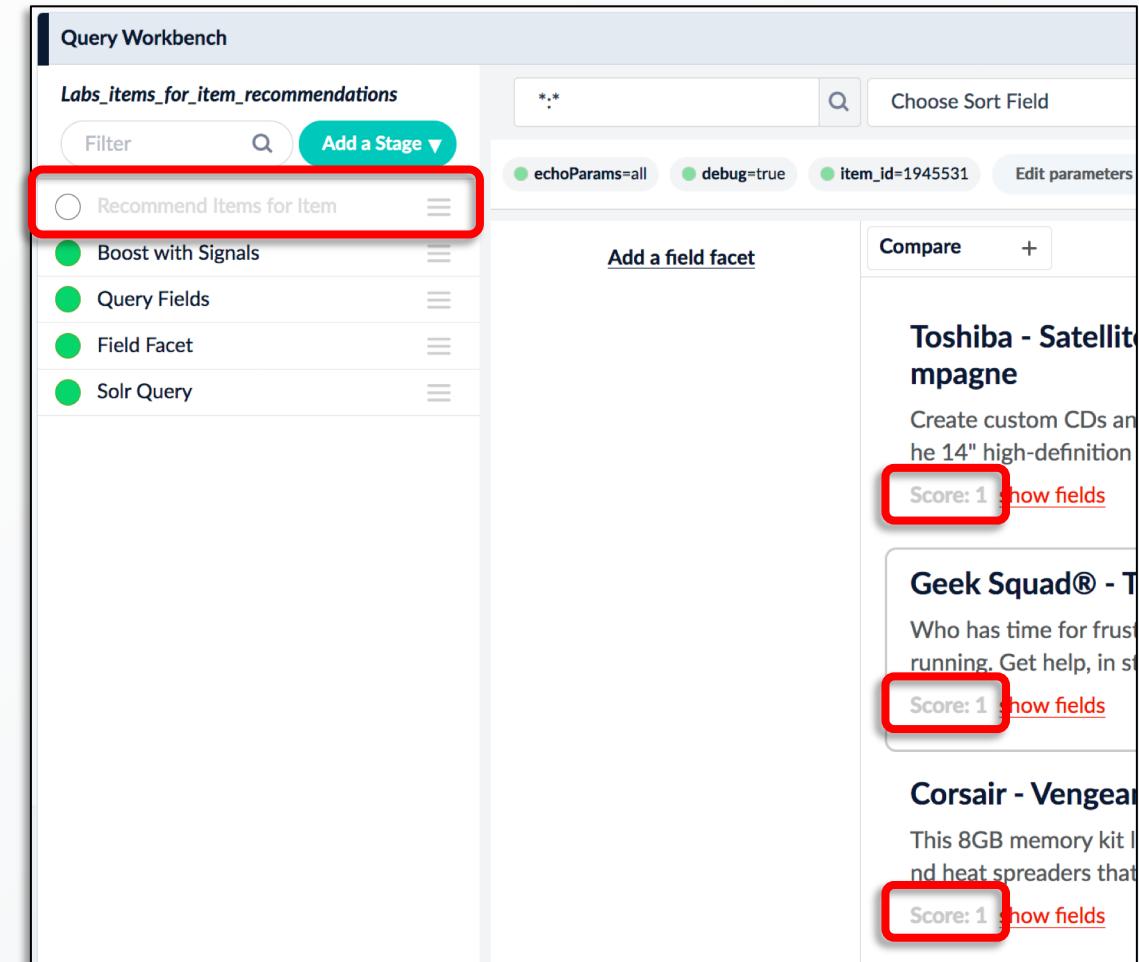
1 2 3 4 5 6 7 8 ...4202 Next
1-10 of 42,012 docs (1 ms, max-score 15)

Testing Item-Item Recommendations

- In the Pipeline pane, disable the **Recommend Items for Item** stage by clicking the green circle

Now we are back to a pseudo-random document ordering, with all results having an identical score of 1 despite our `item_id` parameter

- Re-enable **Recommend Items for Item** stage by clicking the circle again



The screenshot shows the Lucidworks Query Workbench interface. In the top left, there's a pipeline pane titled "Labs_items_for_item_recommendations" with several stages listed:

- Filter (disabled)
- Recommend Items for Item** (disabled, highlighted with a red box)
- Boost with Signals
- Query Fields
- Field Facet
- Solr Query

At the top right, there are search parameters: `*:*`, `Choose Sort Field`, and `echoParams=all`, `debug=true`, `item_id=1945531`. Below the pipeline pane, there's a "Compare" section and a list of search results:

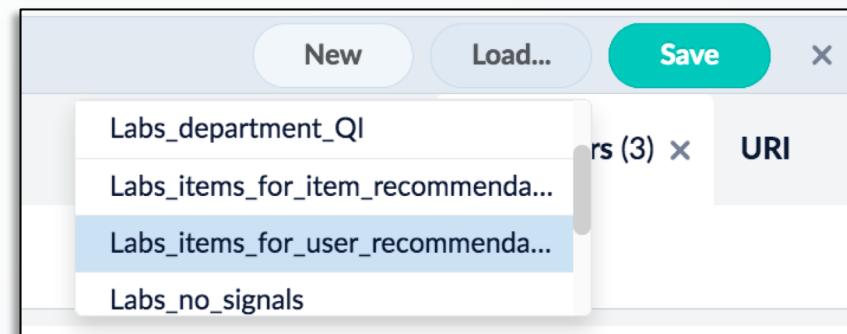
- Toshiba - Satellite mpagne: Score: 1 [show fields]
- Geek Squad® - T: Who has time for frust running. Get help, in st: Score: 1 [show fields]
- Corsair - Vengea: This 8GB memory kit l and heat spreaders that: Score: 1 [show fields]

Each result entry includes a "Score: 1" and a "[show fields]" link, which is also highlighted with a red box.

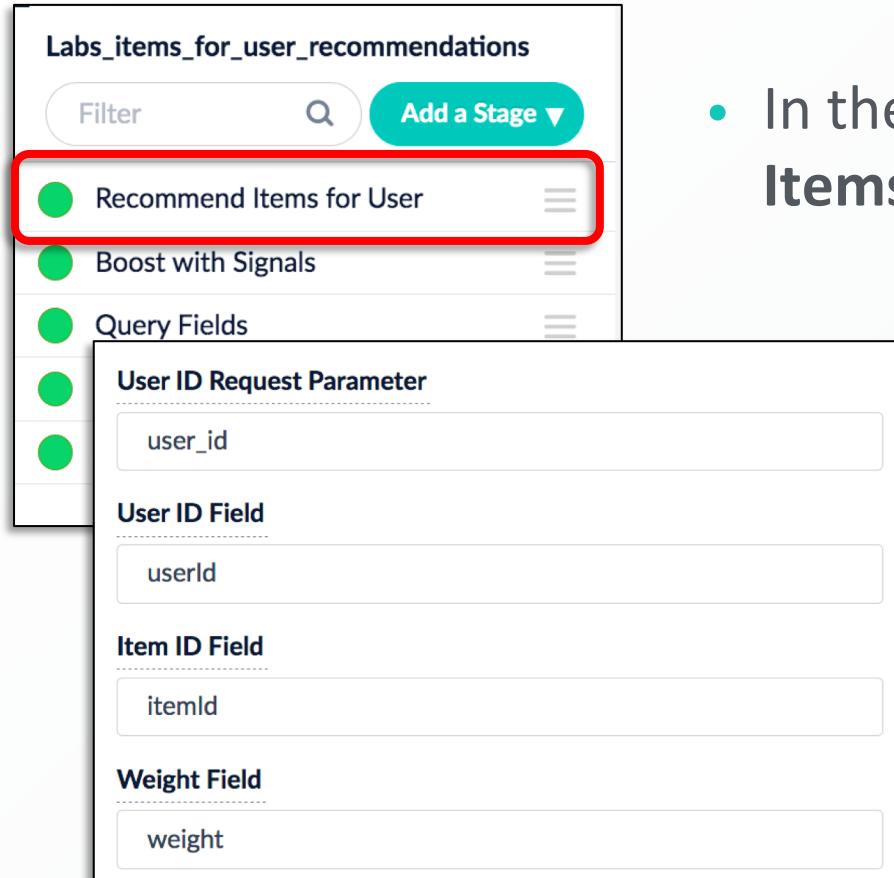
Testing User-Item Recommendations

There are two ways to implement a user-item recommendation. One is Personalized Recommendations, where we offer users a list of interesting items before they ask for anything. Another is Personalized Search, where we apply boost factors to items based on the user-item similarity of the person looking. We will start by testing Personalized Recommendations.

- Go to the **Query Workbench** of the **Labs** collection
- At the top right, click **Load**
- Select the pipeline **Labs_items_for_user_recommendations**



Similarly as before, this pipeline has a special stage that implements recommendations.



The screenshot shows the Pipeline pane with the title "Labs_items_for_user_recommendations". At the top, there is a "Filter" button and a "Add a Stage" button. Below these, a list of stages is shown:

- Recommend Items for User (highlighted with a red box)
- Boost with Signals
- Query Fields

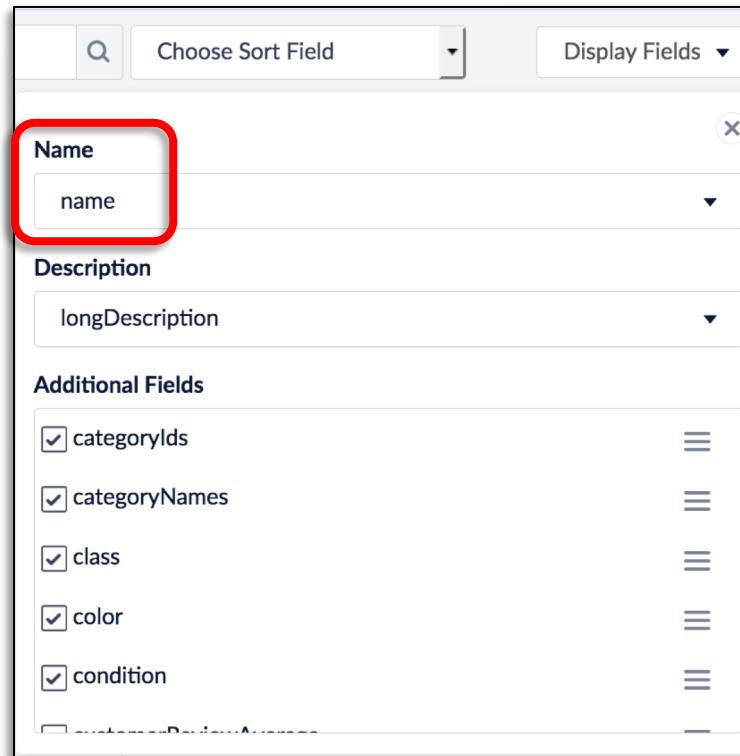
Below the stages, there is a "User ID Request Parameter" section with a field for "user_id" containing the value "user_id". There are also sections for "User ID Field" (with "userId") and "Item ID Field" (with "itemId"). At the bottom, there is a "Weight Field" section with a field for "weight".

- In the Pipeline pane, select the **Recommend Items for User** stage

Near the bottom of the stage editor you will see some familiar parameters. In this case, the key query parameter is **user_id**, which will generally be populated by a login action.

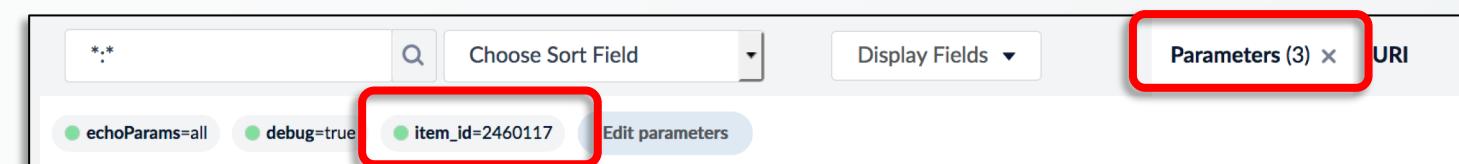
The user-item recommendations collections will be searched for items that have a **userId** field equal to **user_id**, and will provide a relevance boost on the associated **itemId**. The **weight** determines the magnitude of the boost

- In the **Display Fields** dropdown, set the **Name** field to **name**



The screenshot shows a search interface with a 'Choose Sort Field' dropdown and a 'Display Fields' dropdown. The 'Display Fields' dropdown is open, showing a list of fields under 'Name'. The 'name' field is highlighted with a red box. Other fields listed are 'Description' (with 'longDescription') and 'Additional Fields' (with checked boxes for 'categoryId', 'categoryName', 'class', 'color', and 'condition').

- At the top right, click **Parameters > Edit Parameters**



The screenshot shows the search interface with the 'Parameters' section highlighted by a red box. It displays three parameters: 'echoParams=all', 'debug=true', and 'item_id=2460117'. The 'item_id' parameter is also highlighted with a red box.

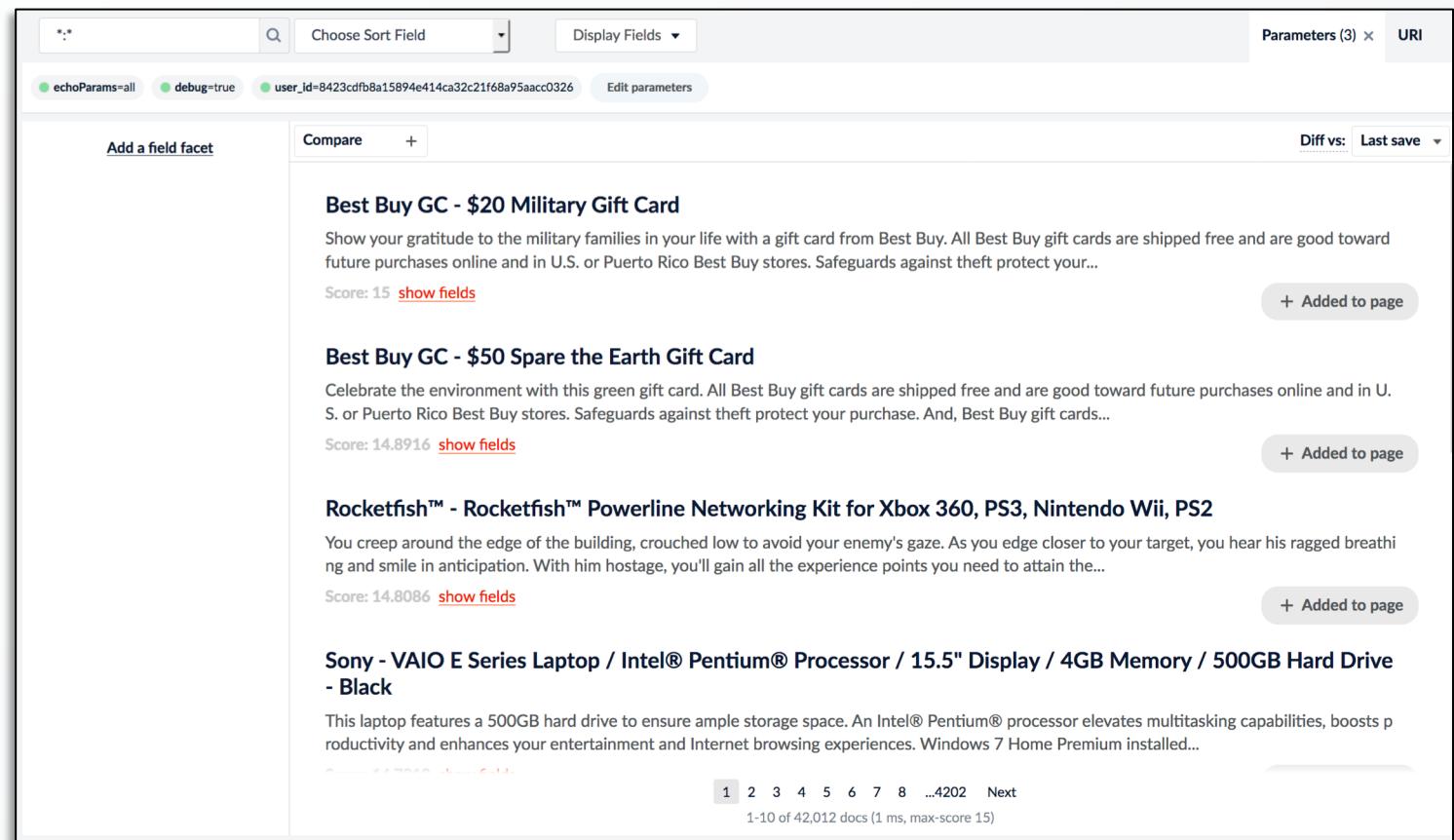
- Delete the `item_id` parameter
- Add a parameter named `user_id` with value
8423cdfb8a15894e414ca32c21f68a95aacc0326

Parameter Name	Parameter Value
echoParams	all
debug	true
user_id	8423cdfb8a15894e414ca32c21f68a95aacc0326

*The default *:***** query, combined with the user-item recommender, gives us Personalized Recommendations. These are the items that this user would see when they first log on, before executing any queries.*

Again, your results may differ from these. The severe downsampling we did in the model building job yields inconsistency.

*You can observe the difference in results when the **Recommend Items for User stage** is disabled by clicking the green circle, as before*

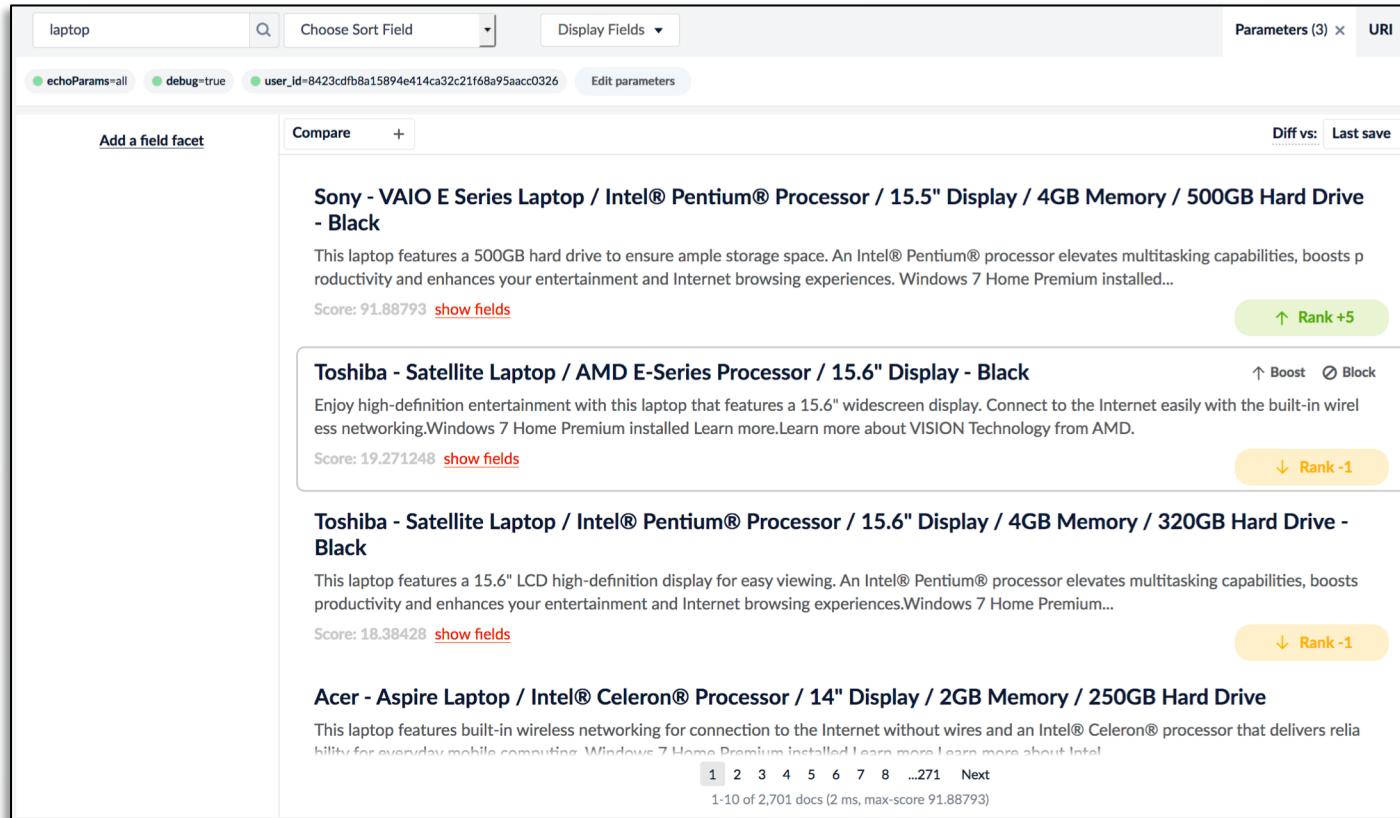


The screenshot shows a search interface with the following details:

- Query Bar:** *:*****
- Search Buttons:** Choose Sort Field, Display Fields
- Parameters:** echoParams=all, debug=true, user_id=8423cdfb8a15894e414ca32c21f68a95aacc0326, Edit parameters
- Compare:** Diff vs: Last save
- Results:**
 - Best Buy GC - \$20 Military Gift Card**
Score: 15 [show fields](#)
+ Added to page
 - Best Buy GC - \$50 Spare the Earth Gift Card**
Score: 14.8916 [show fields](#)
+ Added to page
 - Rocketfish™ - Rocketfish™ Powerline Networking Kit for Xbox 360, PS3, Nintendo Wii, PS2**
Score: 14.8086 [show fields](#)
+ Added to page
 - Sony - VAIO E Series Laptop / Intel® Pentium® Processor / 15.5" Display / 4GB Memory / 500GB Hard Drive - Black**
Score: 14.8086 [show fields](#)
+ Added to page
- Pagination:** 1 2 3 4 5 6 7 8 ...4202 Next
- Page Info:** 1-10 of 42,012 docs (1 ms, max-score 15)

Testing User-Item Recommendations

- Execute the query **laptop**



The screenshot shows a search interface with the query "laptop" entered in the search bar. The results are displayed as a list of four laptop models:

- Sony - VAIO E Series Laptop / Intel® Pentium® Processor / 15.5" Display / 4GB Memory / 500GB Hard Drive - Black**
This laptop features a 500GB hard drive to ensure ample storage space. An Intel® Pentium® processor elevates multitasking capabilities, boosts productivity and enhances your entertainment and Internet browsing experiences. Windows 7 Home Premium installed...
Score: 91.88793 [show fields](#) ↑ Rank +5
- Toshiba - Satellite Laptop / AMD E-Series Processor / 15.6" Display - Black**
Enjoy high-definition entertainment with this laptop that features a 15.6" widescreen display. Connect to the Internet easily with the built-in wireless networking. Windows 7 Home Premium installed [Learn more](#). Learn more about VISION Technology from AMD.
Score: 19.271248 [show fields](#) ↓ Rank -1
- Toshiba - Satellite Laptop / Intel® Pentium® Processor / 15.6" Display / 4GB Memory / 320GB Hard Drive - Black**
This laptop features a 15.6" LCD high-definition display for easy viewing. An Intel® Pentium® processor elevates multitasking capabilities, boosts productivity and enhances your entertainment and Internet browsing experiences. Windows 7 Home Premium...
Score: 18.38428 [show fields](#) ↓ Rank -1
- Acer - Aspire Laptop / Intel® Celeron® Processor / 14" Display / 2GB Memory / 250GB Hard Drive**
This laptop features built-in wireless networking for connection to the Internet without wires and an Intel® Celeron® processor that delivers reliability for everyday mobile computing. Windows 7 Home Premium installed [Learn more](#) [Learn more about Intel](#)
Score: 17.597222 [show fields](#)

At the bottom of the results page, there is a navigation bar with page numbers 1, 2, 3, 4, 5, 6, 7, 8, ..., 271, Next, and a footer note: 1-10 of 2,701 docs (2 ms, max-score 91.88793).

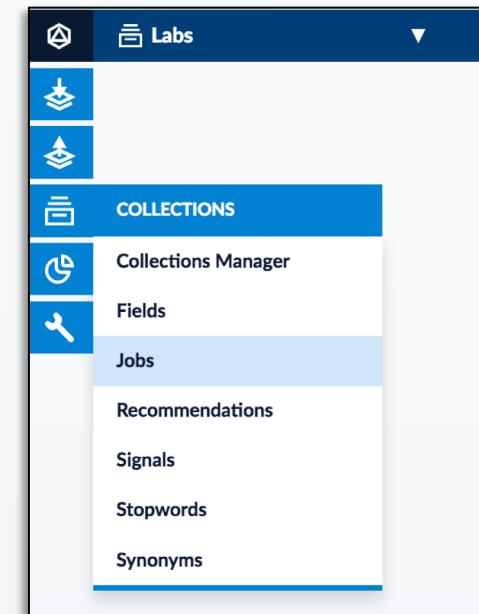
The top result for this user was originally 5 ranks lower, without the recommender active. This is an example of Personalized Search.

Again, your results may differ from these. For better results, use a larger sample when building the recommender model.

Building a Better Recommender (optional)

If you can spare the time, try the Testing sections of this lab with a non-downsampled model. This section will walk you through the steps for doing so.

- In the left side menu of Fusion Admin, go to **COLLECTIONS > JOBS**
- Select **Labs_items_recommendations**
- Toggle the **Advanced** switch at the top right



- Change the **Training Data Sampling Fraction** parameter to **1**



- Save and Run the updated job

With full sampling, the job should take ~10 minutes to complete. Once finished, try the Testing steps again, and see the improved results!

End of Lab