

# LAB: Building a Query Intent Classifier

## Prerequisites

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

## Recommended

- LAB: Working with Signals

# Inspecting Labs\_signals

*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.*

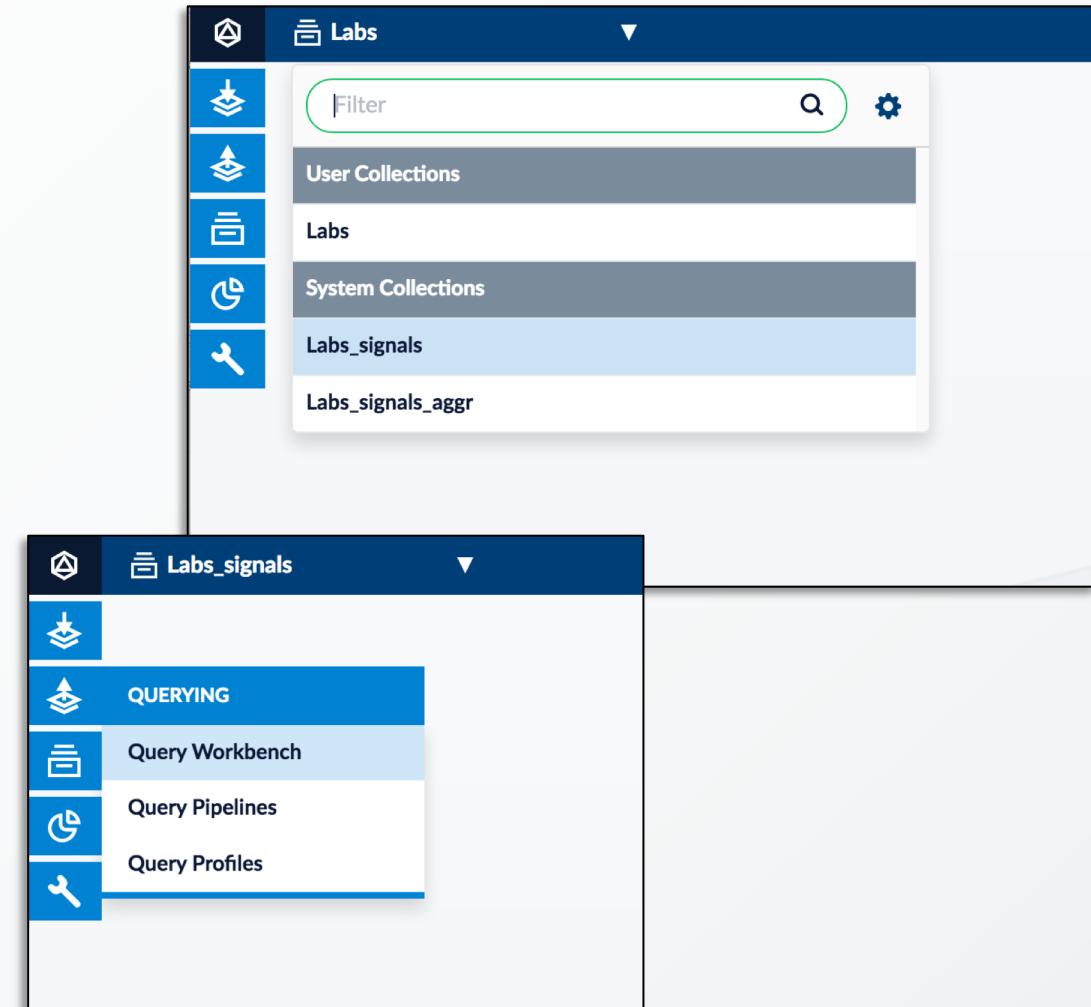
- In a bash/shell terminal, start Fusion

```
./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\_signals** collection
- In the left side menu, go to **QUERYING > Query Workbench**

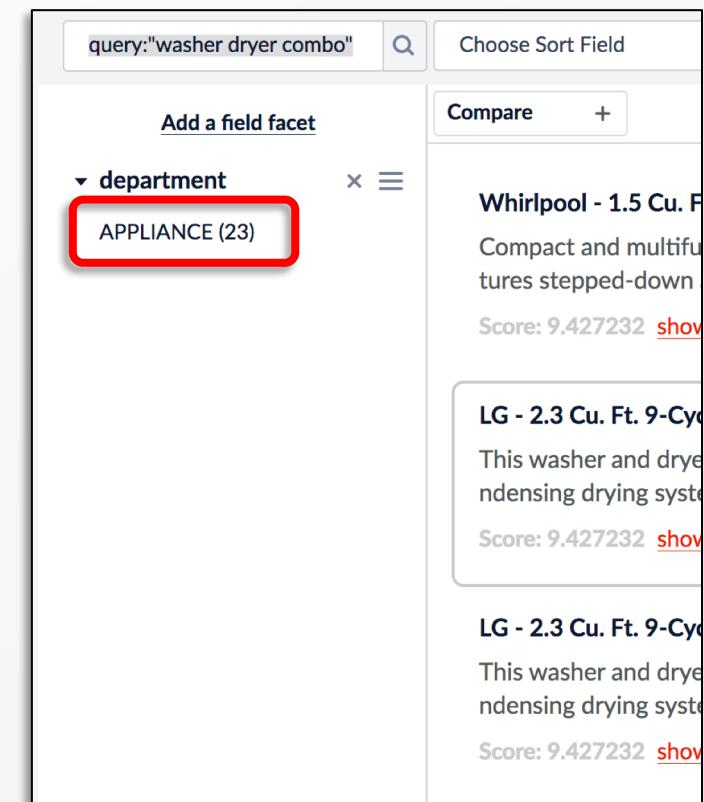


*In this lab, we will build a classifier that predicts which store department a user is interested in based on their input query. Doing so involves finding correlations between certain query terms and the store department the user ultimately clicks into.*

- Execute the query **query:"washer dryer combo"**

*Unsurprisingly, everyone who searched for “**washer dryer combo**” ultimately clicked on an item from the **APPLIANCE** department.*

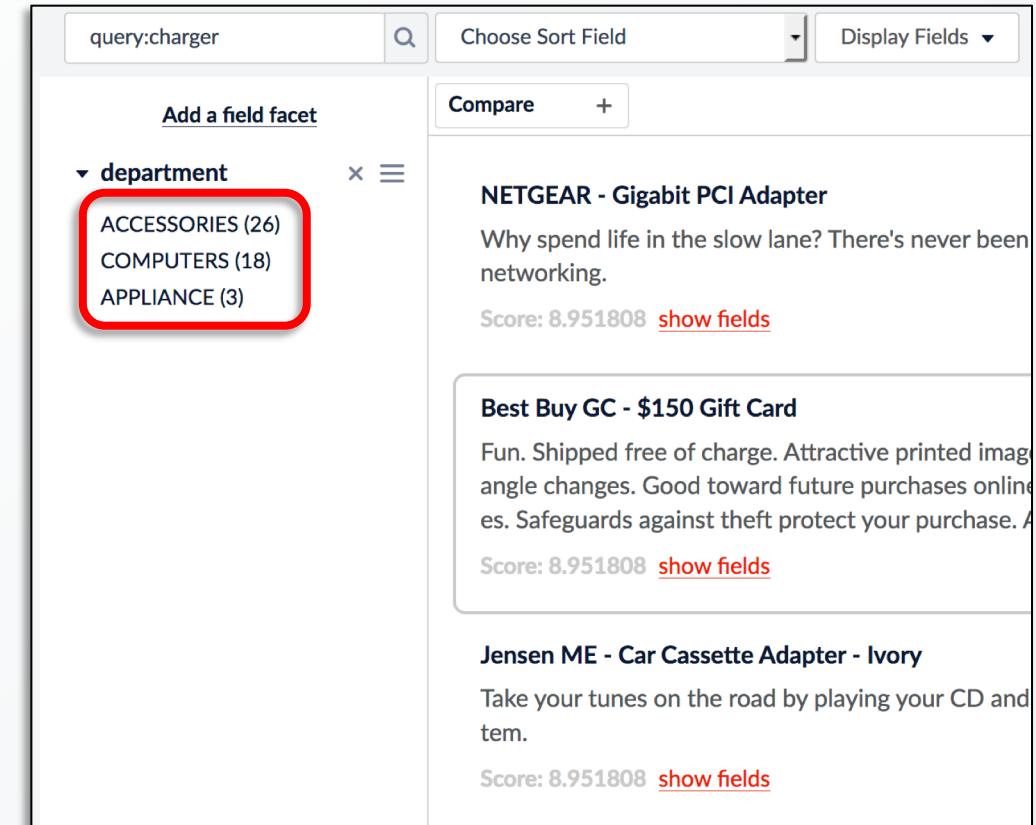
*Not every query is so clear-cut, however...*



- Execute the query **query:charger**

*These results are far more ambiguous. That said, there is still a clear trend towards **ACCESSORIES**.*

*These patterns can be captured and learned by a Classifier, and used to influence relevancy at query time*



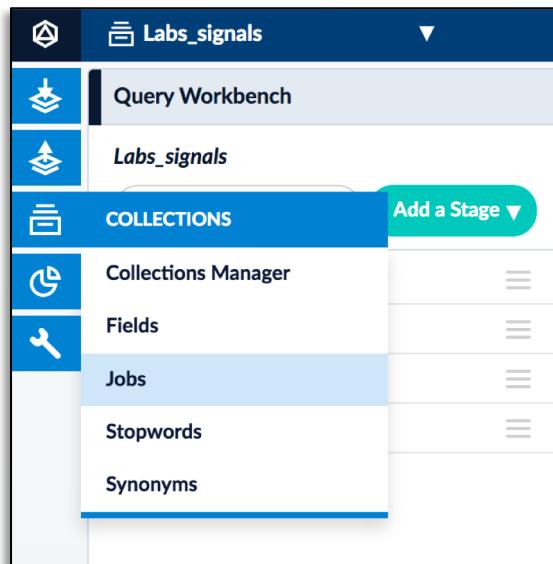
The screenshot shows a search interface with the query "query:charger" entered. On the left, a facet panel titled "department" displays three categories: "ACCESSORIES (26)" (highlighted with a red box), "COMPUTERS (18)", and "APPLIANCE (3)". To the right, three search results are listed:

- NETGEAR - Gigabit PCI Adapter**  
Why spend life in the slow lane? There's never been networking.  
Score: 8.951808 [show fields](#)
- Best Buy GC - \$150 Gift Card**  
Fun. Shipped free of charge. Attractive printed image. Angle changes. Good toward future purchases online. Safeguards against theft protect your purchase. A  
Score: 8.951808 [show fields](#)
- Jensen ME - Car Cassette Adapter - Ivory**  
Take your tunes on the road by playing your CD and  
Score: 8.951808 [show fields](#)

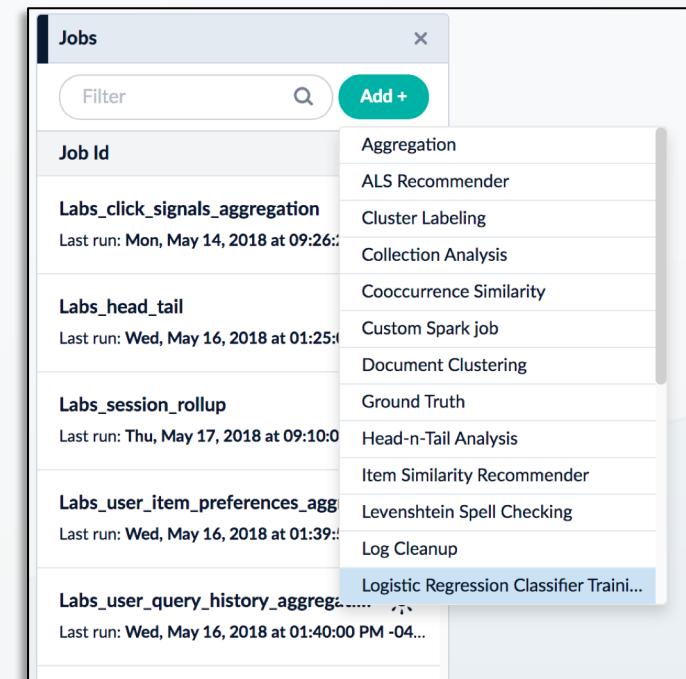
# Training a Classifier Model

*Like most Machine Learning tasks, Classification models are built using Spark jobs*

- In the left side menu of Fusion Admin, navigate to **COLLECTIONS > JOBS**



- In Jobs pane, click Add
- Select **Logistic Regression Classifier Training Job**



A screenshot of the "Jobs" pane in Fusion Admin. The pane has a header with "Jobs" and a close button. Below the header are "Filter" and "Add +" buttons. A table lists various jobs with their details:

Job Id	Type
Labs_click_signals_aggregation	Aggregation
Labs_head_tail	ALS Recommender
Labs_session_rollup	Cluster Labeling
Labs_user_item_preferences_agg	Collection Analysis
Labs_user_query_history_aggregat...	Cooccurrence Similarity
	Custom Spark job
	Document Clustering
	Ground Truth
	Head-n-Tail Analysis
	Item Similarity Recommender
	Levenshtein Spell Checking
	Log Cleanup
	Logistic Regression Classifier Train...

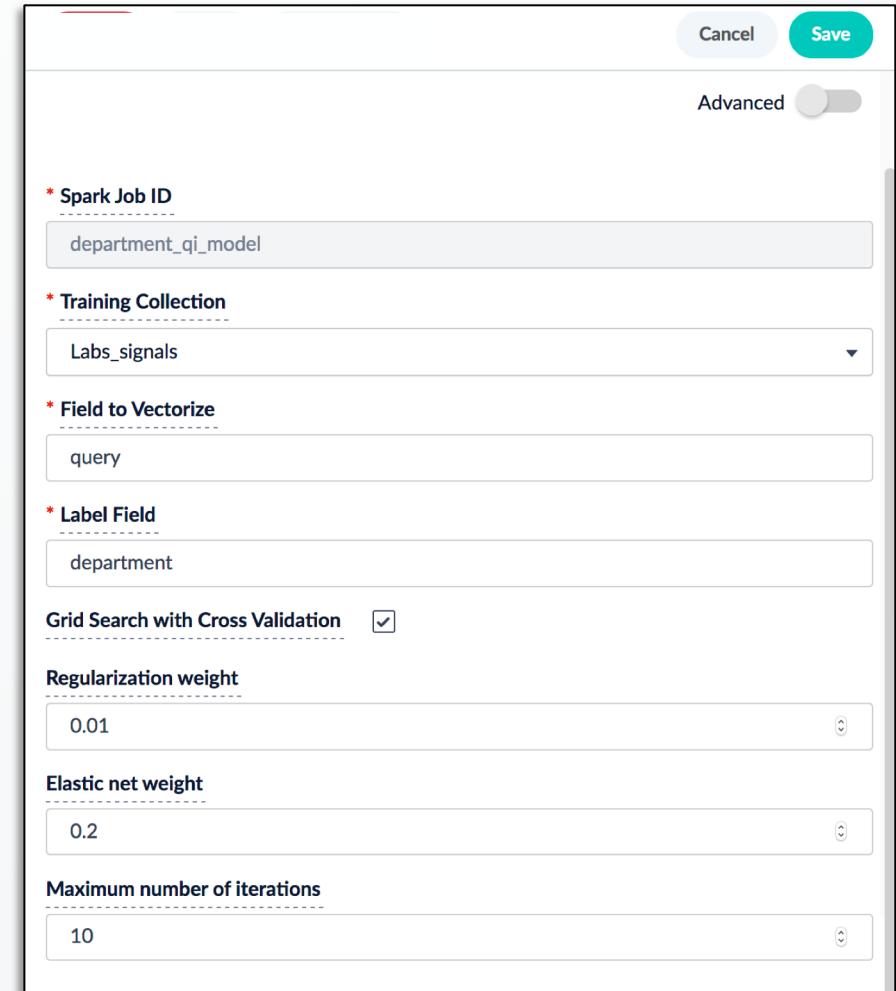
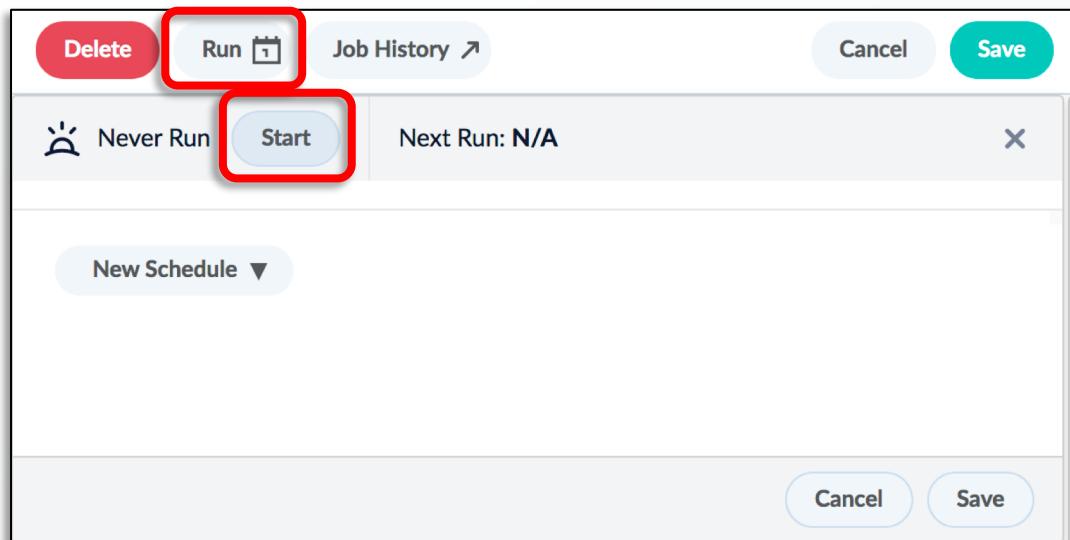
- Fill out the job parameters according to the following table:

Parameter	Value	Explanation
<b>Spark Job ID</b>	department_qi_model	<i>Unique name for this job. This will also be the model name, so try to make it intuitive</i>
<b>Training Collection</b>	Labs_signals	<i>Collection from which to draw training data.</i>
<b>Field to Vectorize</b>	query	<i>The input field. The model will predict a label based on the contents of this field</i>
<b>Label Field</b>	department	<i>The output field. The model will write its prediction label here.</i>
<b>Elastic Net Weight</b>	0.2	<i>The Elastic Net (<a href="#">link</a>) allows smooth interpolation of the <math>L_1</math> and <math>L_2</math> regularization methods. There is no single “correct” value for this, but a small number between 0 and 1 is a good base.</i>
<b>Grid Search with Cross Validation</b>	enabled	<i>Cross Validation is always enabled. This parameter also enables Grid Search, which will experimentally determine the “best” values for <b>Elastic Net Weight</b> and <b>Regularization Weight</b></i>

# Running the Classifier Job

The configured job should look like this:

- Click **Save**
- Run the job by clicking **Run > Start**

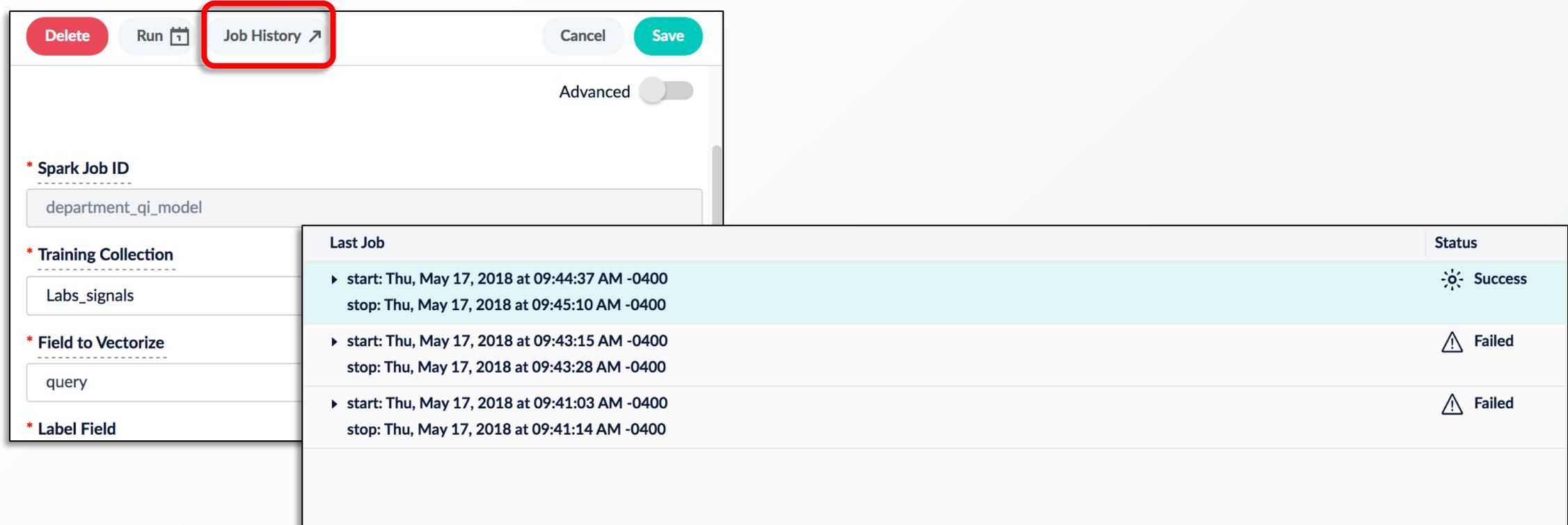


A detailed configuration form for a classifier job. It includes fields for 'Spark Job ID' (department\_qj\_model), 'Training Collection' (Labs\_signals), 'Field to Vectorize' (query), 'Label Field' (department), and a checked 'Grid Search with Cross Validation' checkbox. It also specifies 'Regularization weight' (0.01), 'Elastic net weight' (0.2), and 'Maximum number of iterations' (10). At the top right, there are 'Cancel' and 'Save' buttons, with an 'Advanced' toggle switch turned off.

* Spark Job ID	department_qj_model
* Training Collection	Labs_signals
* Field to Vectorize	query
* Label Field	department
Grid Search with Cross Validation	<input checked="" type="checkbox"/>
Regularization weight	0.01
Elastic net weight	0.2
Maximum number of iterations	10

# Running the Classifier Job

- Once the job finishes (~2 minutes), click **Job History**
- Expand the topmost run report



The screenshot shows the 'Job History' tab highlighted with a red box. On the left, there are configuration fields for 'Spark Job ID' (set to 'department\_qi\_model'), 'Training Collection' (set to 'Labs\_signals'), 'Field to Vectorize' (set to 'query'), and 'Label Field'. A 'Save' button is visible. On the right, a table titled 'Last Job' lists three runs:

	Status
▶ start: Thu, May 17, 2018 at 09:44:37 AM -0400 stop: Thu, May 17, 2018 at 09:45:10 AM -0400	 Success
▶ start: Thu, May 17, 2018 at 09:43:15 AM -0400 stop: Thu, May 17, 2018 at 09:43:28 AM -0400	 Failed
▶ start: Thu, May 17, 2018 at 09:41:03 AM -0400 stop: Thu, May 17, 2018 at 09:41:14 AM -0400	 Failed

Observe the **confusion matrix**. This is a quick report of how accurate the model is, based on cross-validation testing. In this case, the model shows 82% accuracy for **COMPUTERS** labels, 79% accuracy for **ACCESSORIES**, and 77% accuracy for **APPLIANCE**. Not bad, considering we made no effort to clean or balance the training data.

Additionally, we can see that **COMPUTERS** are most often mistaken for **ACCESSORIES**; **ACCESSORIES** are most often mistaken for **COMPUTERS**; and **APPLIANCE** are most often mistaken for **ACCESSORIES**.

Starttime: 2018-05-17T13:44:37.286Z

Endtime: 2018-05-17T13:45:10.073Z

Status: success

Extra:

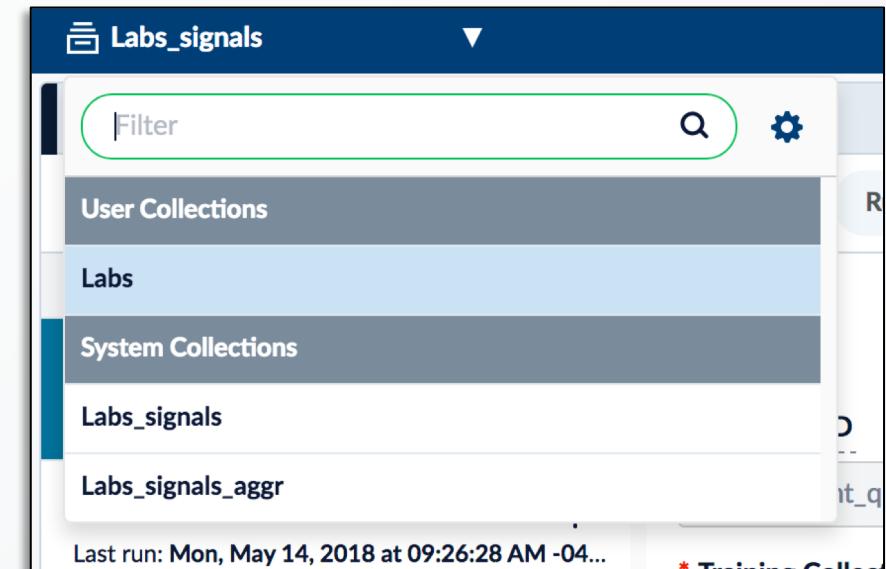
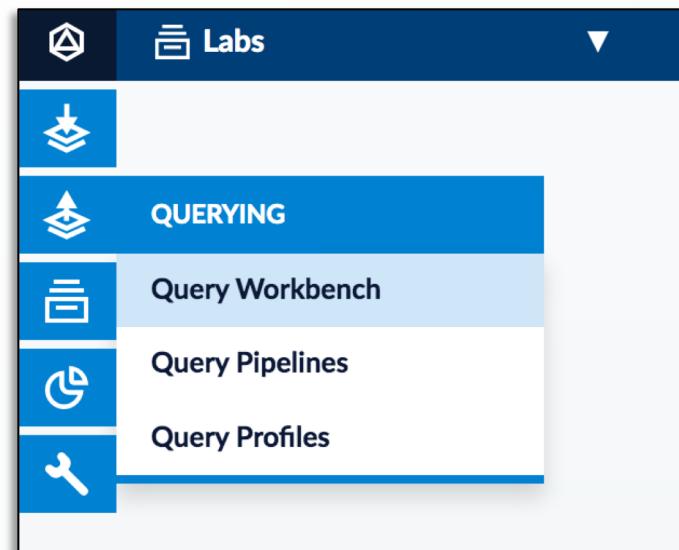
```
Sparkjobstatus.result.data: { "confusionMatrix": ["COMPUTERS": {"APPLIANCE": 1533.0, "ACCESSORIES": 5300.0, "COMPUTERS": 31266.0}, "ACCESSORIES": ["APPLIANCE": 827.0, "ACCESSORIES": 10127.0, "COMPUTERS": 1727.0], "APPLIANCE": ["APPLIANCE": 4955.0, "ACCESSORIES": 1111.0, "COMPUTERS": 366.0]}, "accuracy": ["COMPUTERS": 0.820651460668259, "ACCESSORIES": 0.7985963252109455, "APPLIANCE": 0.7703669154228856] }
```

Sparkjobstatus.jobid: 1636e5866a1Tf796a648

Sparkjobstatus.result.state: finished

# Implementing Classification at Query Time

- In the topmost dropdown in Fusion Admin, switch to the **Labs** collection
- In the left side menu, go to **QUERYING > QUERY WORKBENCH**

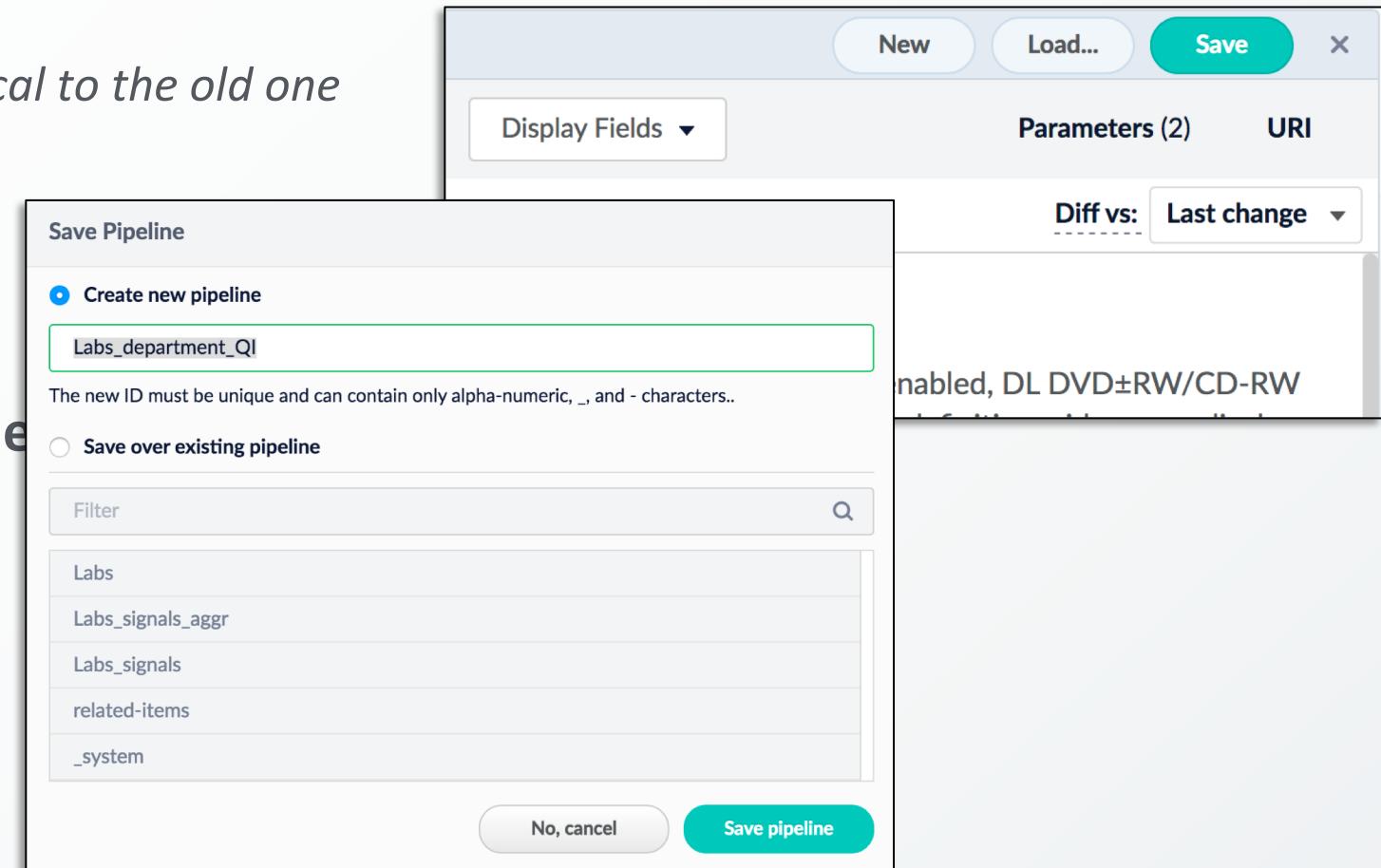


The image shows the 'Labs\_signals' collection page in the Fusion Admin interface. At the top, there is a search bar labeled 'Filter' with a magnifying glass icon and a gear icon for settings. Below the search bar, the page is divided into sections: 'User Collections' (containing 'Labs'), 'System Collections' (containing 'Labs\_signals' and 'Labs\_signals\_aggr'), and a footer note: 'Last run: Mon, May 14, 2018 at 09:26:28 AM -04...'. The 'Labs' collection is currently selected.

- Create a new query pipeline by clicking **New** in the top right corner

*The initial pipeline will look identical to the old one*

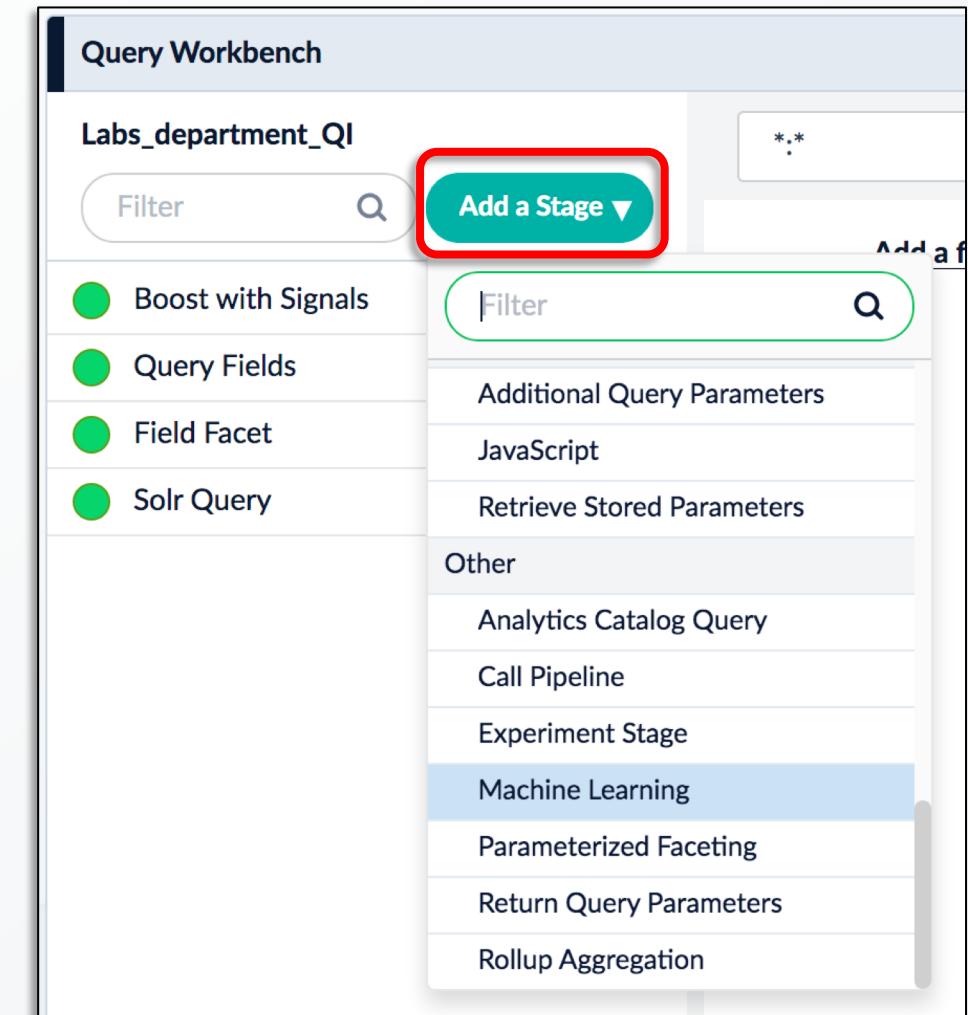
- Click **Save**
- Name the pipeline **Labs\_department\_Q1**
- Click **Save Pipeline**



# Adding a Machine Learning Stage

- In the Pipeline pane, click Add Stage
- Select Machine Learning

*A new pipeline stage editor pane will appear*



- Fill out the stage parameters according to the following table:

Parameter	Value	Explanation
<b>Label</b>	Department_QI_classifier	<i>Unique name for this pipeline stage.</i>
<b>Machine Learning Model ID</b>	department_qi_model	<i>Classification model to use. Recall that the model has the same name as the job used to create it. You can check the name and existence of the model by navigating to <b>SYSTEM &gt; Blobs &gt; ML Model</b></i>
<b>Prediction Field Name</b>	predicted_department	<i>Field used to store the prediction label output by the classifier. By itself, this labeling does nothing—we will implement another stage later that does something with this field</i>

# Adding a Machine Learning Stage

**Department\_QI\_classifier**

Machine Learning  
Use a machine learning model to generate a prediction about a query.

**Label**  
Department\_QI\_classifier

**Condition** OPEN EDITOR 

1	
---	--

**\* Machine Learning Model ID**  
department\_qi\_model

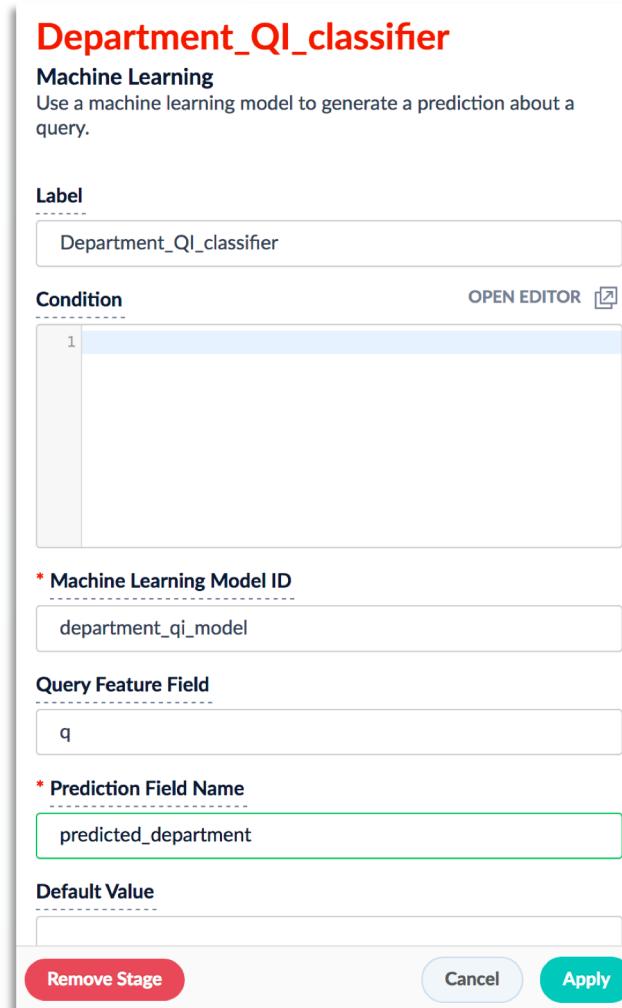
**Query Feature Field**  
q

**\* Prediction Field Name**  
predicted\_department

**Default Value**

**Actions**

Remove Stage Cancel Apply



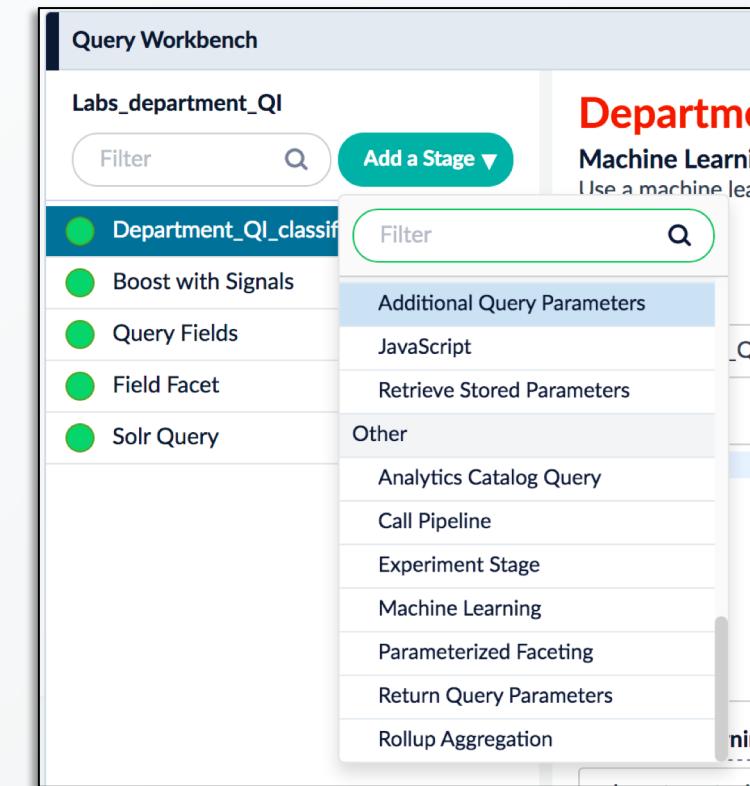
- At the bottom of the stage editor, click **Apply**
- At the top right of the workbench, click **Save**

*If prompted, specify **Save over existing pipeline***

The classifier stage applies a label in the `predicted_department` field of each query request, but doesn't actually do anything with that label. We will create another stage that uses the label to filter or boost documents based on the label.

- At the top right of the workbench, click **Save**
- In the Pipeline pane, click **Add Stage**
- Select **Additional Query Parameters**

A new pipeline stage editor pane will appear

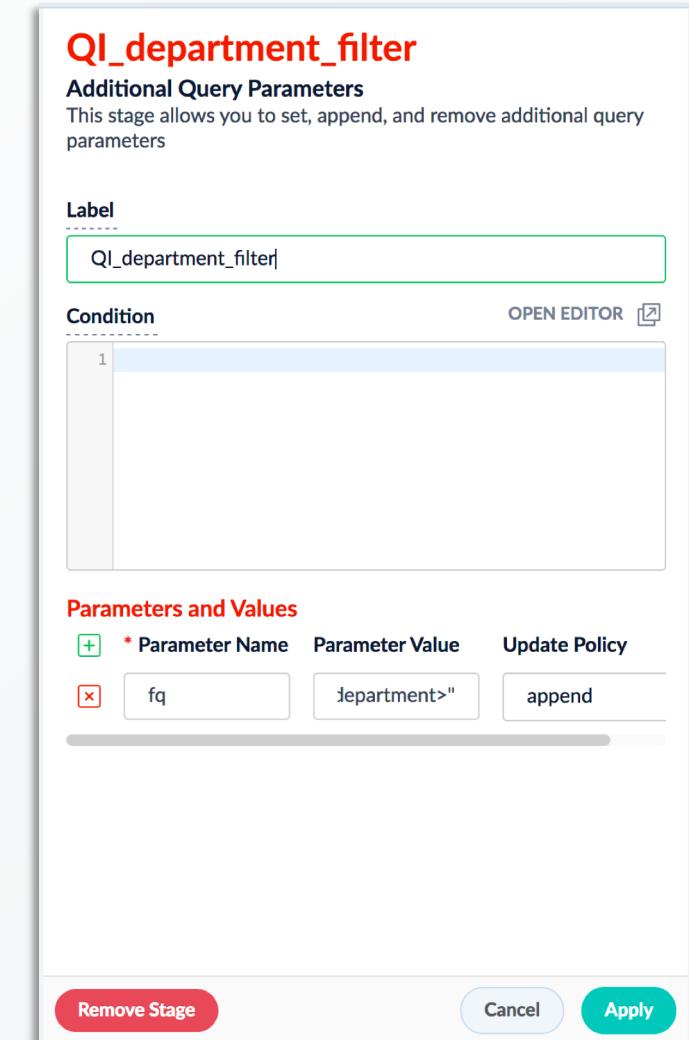
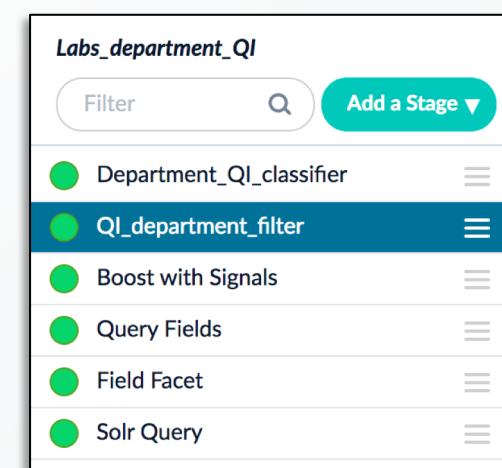


- Name the new stage **QI\_department\_filter**
- Click the green + to add a new parameter
- Fill out that parameter according to the following table:

Parameter	Value	Explanation
Parameter Name	fq	<i>fq (Filter Query) filters search results, rather than affecting relevance</i>
Parameter Value	department:<predicted_department>	<i>This query matches items where the department field in the document matches the predicted_department field in the query</i>

*The finished stage should look like this:*

- At the bottom of the stage editor, click **Apply**
- In the pipeline pane, click and drag **QI\_department\_filter** so that it occurs after **Department\_QI\_classifier**
- At the top right of the workbench, click **Save**

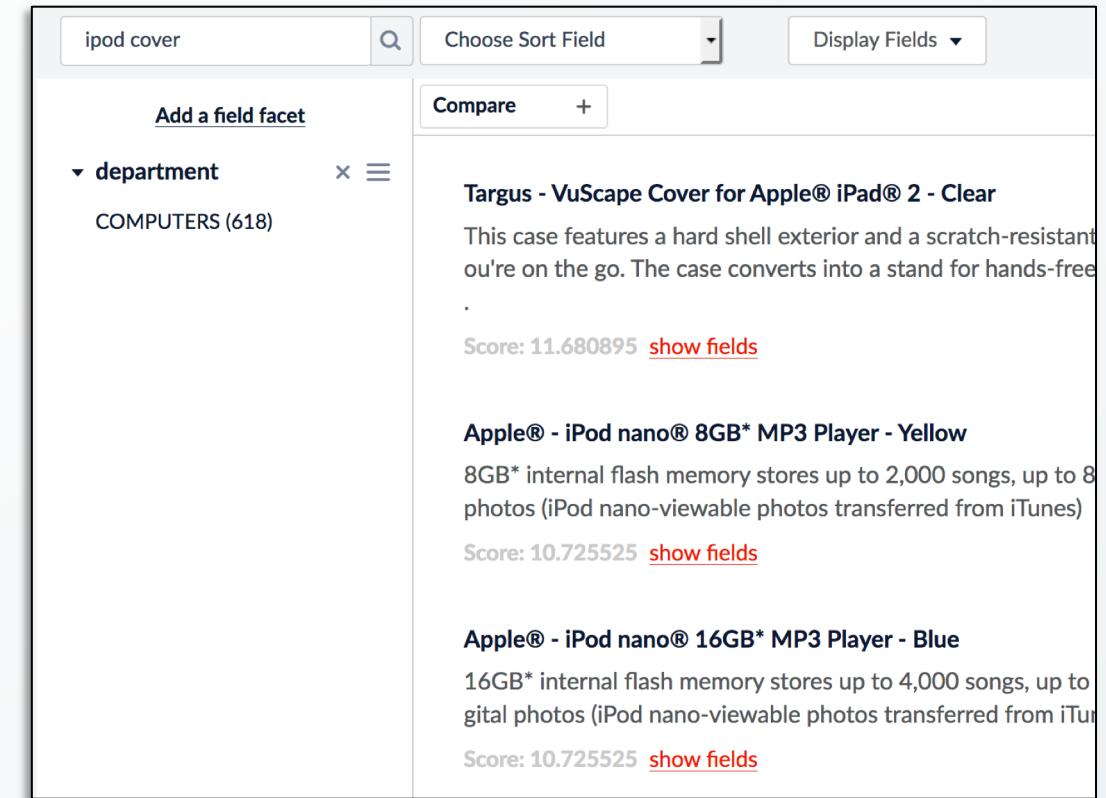


# Testing the Classifier Pipeline

- In the **Labs** Query Workbench, execute the query **iPod cover**

- Add a facet field for **department**

*The classifier associates the query **iPod cover** with the **COMPUTERS** department, so the department filter stage eliminates all documents from every other department*

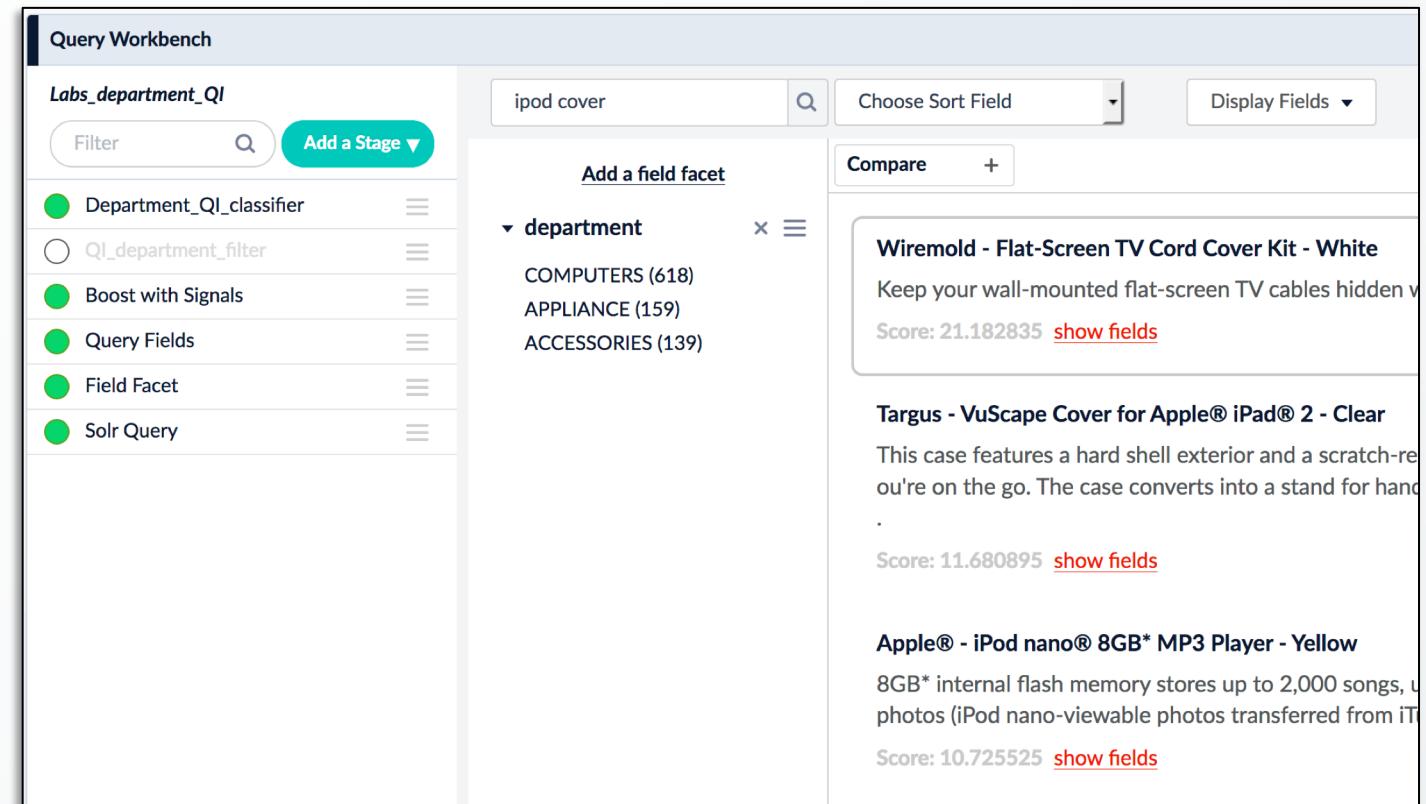


The screenshot shows the Lucidworks Query Workbench interface. The search bar at the top contains the query "ipod cover". Below the search bar, there are buttons for "Choose Sort Field" and "Display Fields". A facet panel on the left shows a selected facet for "department" with the value "COMPUTERS (618)". To the right of the facet panel, the search results are displayed. The first result is a product listing for "Targus - VuScape Cover for Apple® iPad® 2 - Clear", which is associated with the "COMPUTERS" department. The second result is a product listing for "Apple® - iPod nano® 8GB\* MP3 Player - Yellow", and the third result is another listing for "Apple® - iPod nano® 16GB\* MP3 Player - Blue". Each result includes a score (e.g., 11.680895, 10.725525) and a "show fields" link.

Score	Product	Description
11.680895	Targus - VuScape Cover for Apple® iPad® 2 - Clear	This case features a hard shell exterior and a scratch-resistant interior. It's perfect for on-the-go. The case converts into a stand for hands-free viewing.
10.725525	Apple® - iPod nano® 8GB* MP3 Player - Yellow	8GB* internal flash memory stores up to 2,000 songs, up to 800 photos (iPod nano-viewable photos transferred from iTunes)
10.725525	Apple® - iPod nano® 16GB* MP3 Player - Blue	16GB* internal flash memory stores up to 4,000 songs, up to 1,000 digital photos (iPod nano-viewable photos transferred from iTunes)

- In the Pipeline pane, disable the classifier by clicking the green circle next to **QI\_department\_filter**

*With the filter disabled, the query is still classified as before, but nothing is done with the label. Note that the department facet now includes documents from APPLIANCE and ACCESSORIES as well as COMPUTERS*



The screenshot shows the Lucidworks Query Workbench interface. In the top left, the pipeline pane displays stages: 'Department\_QI\_classifier' (green circle), 'QI\_department\_filter' (white circle), 'Boost with Signals', 'Query Fields', 'Field Facet', and 'Solr Query'. The 'QI\_department\_filter' stage is currently disabled. The main search bar contains the query 'ipod cover'. A facet panel titled 'department' shows categories: COMPUTERS (618), APPLIANCE (159), and ACCESSORIES (139). Below the search bar, three product results are listed:

- Wiremold - Flat-Screen TV Cord Cover Kit - White**  
Keep your wall-mounted flat-screen TV cables hidden  
Score: 21.182835 [show fields](#)
- Targus - VuScape Cover for Apple® iPad® 2 - Clear**  
This case features a hard shell exterior and a scratch-resistant interior. It's perfect for on-the-go. The case converts into a stand for hands-free viewing.  
Score: 11.680895 [show fields](#)
- Apple® - iPod nano® 8GB\* MP3 Player - Yellow**  
8GB\* internal flash memory stores up to 2,000 songs, up to 1,000 photos (iPod nano-viewable photos transferred from iPhoto)

- Re-enable the classifier by clicking the circle next to **QI\_department\_filter**

*Filtering is not the only way we can utilize the classifier label. In fact, it is probably one of the least effective things we could do, because it means that in cases where the classifier is “wrong” the department that the user is actually interested in will not appear at all.*

*Let us instead use the classifier label to implement a relevance boost. In this case, the predicted department will still rise to the top of the results, but the other documents are still there to be found in case the prediction was wrong.*

- Open the **QI\_department\_filter** editor by clicking on the stage in the Pipeline pane

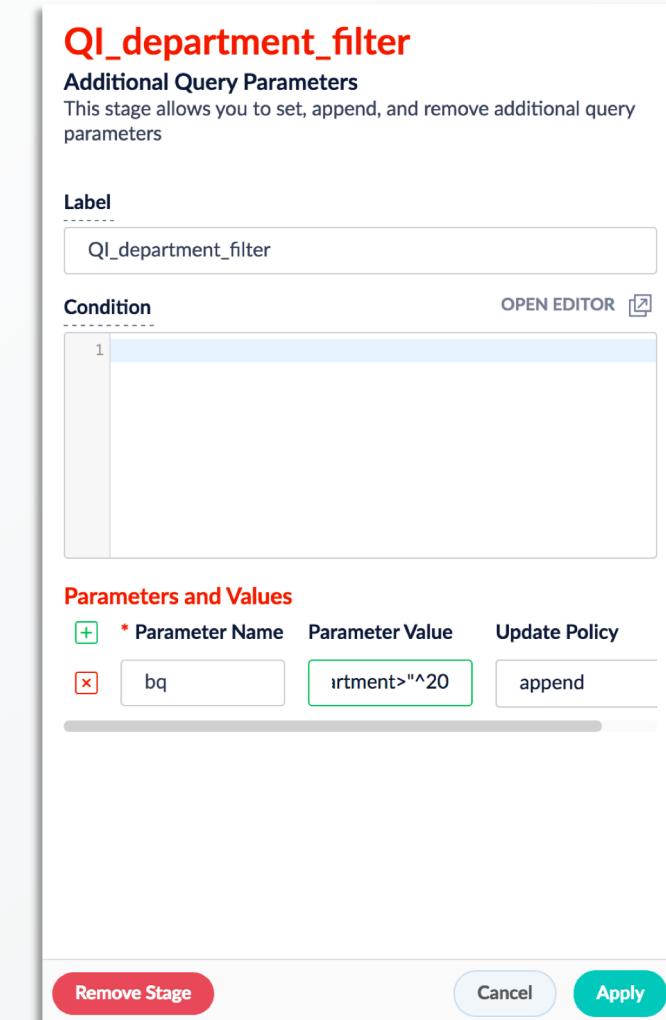
- Change the **fq** parameter to **bq**

*This changes the Filter Query into a Boost query, swapping a filtering action for a relevance-boosting action.*

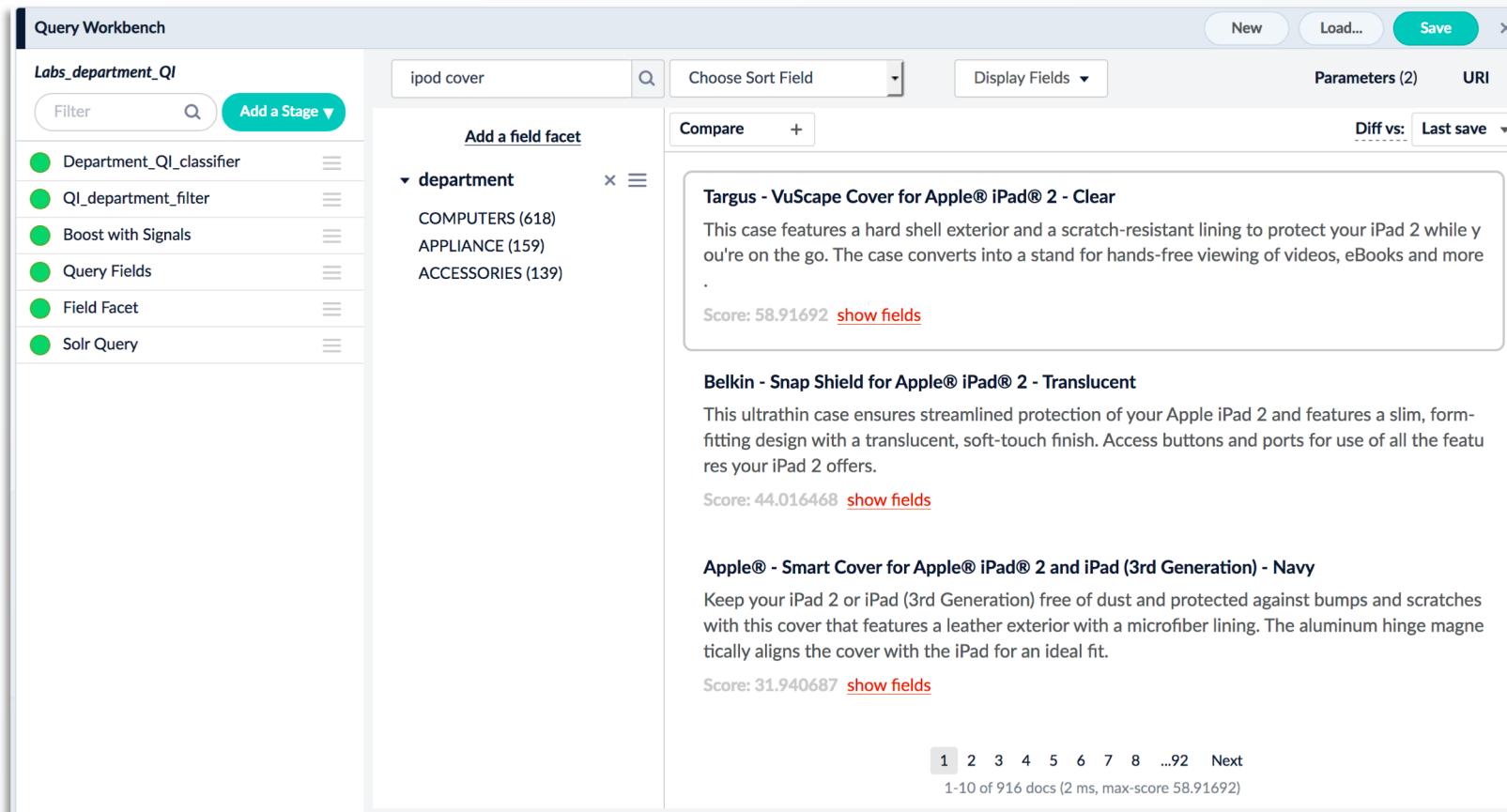
- Change the **Parameter Value** to  
**department:<predicted\_department>"^20**

*The default behavior for **bq** is to add 1 to the relevance score. We'd like a larger boost, adding 20.*

- Click **Apply**



*Now we get the same COMPUTERS documents in the top results as we did with a Filter Query, but without sacrificing the other documents in case of a classifier mistake.*



The screenshot shows the Lucidworks Query Workbench interface. The search bar at the top contains the query "ipod cover". The search results are displayed in a card-based format. The first result is a "department" facet for "COMPUTERS (618)". Below the facet, three document cards are shown:

- Targus - VuScape Cover for Apple® iPad® 2 - Clear**  
This case features a hard shell exterior and a scratch-resistant lining to protect your iPad 2 while you're on the go. The case converts into a stand for hands-free viewing of videos, eBooks and more.  
Score: 58.91692 [show fields](#)
- Belkin - Snap Shield for Apple® iPad® 2 - Translucent**  
This ultrathin case ensures streamlined protection of your Apple iPad 2 and features a slim, form-fitting design with a translucent, soft-touch finish. Access buttons and ports for use of all the features your iPad 2 offers.  
Score: 44.016468 [show fields](#)
- Apple® - Smart Cover for Apple® iPad® 2 and iPad (3rd Generation) - Navy**  
Keep your iPad 2 or iPad (3rd Generation) free of dust and protected against bumps and scratches with this cover that features a leather exterior with a microfiber lining. The aluminum hinge magnetically aligns the cover with the iPad for an ideal fit.  
Score: 31.940687 [show fields](#)

At the bottom of the interface, there is a navigation bar with page numbers (1, 2, 3, 4, 5, 6, 7, 8, ...92, Next) and a footer note: "1-10 of 916 docs (2 ms, max-score 58.91692)".