



Test Driven Relevancy

How to Work with Content Experts to Optimize and Maintain Search Relevancy

Doug Turnbull

OpenSource Connections



Its us!

Hi I'm Doug!



@softwareoug

<http://www.linkedin.com/in/softwareoug>

<http://bit.ly/softwareoug>

Hi I'm Rena!



renam@silverchair.com

How do sales/content curators collaborate with devs?

“When doctors search for ‘myocardial infarction’ these documents about ‘heart attacks’ should come up”



“Myocardial in-what-tion? Dangit Rena, I’m a Solr consultant – not a doctor”

“I don’t even...”



“let me work my Solr magic and get back to you next week...”

How do content curators collaborate with devs?

- Doug knows Solr



`q={!boost b=log(numCatPictures)}bunnies`

`<tokenizer class="solr.WhitespaceTokenizerFactory"/>`

- Bob knows his business

Sales
Supplier Pressure
Conversions
Niche customers



Myocardial infarction

na

This is a universal pattern – it takes different strokes!

How do content curators collaborate with devs?



“Rena, I fixed that myocardial in-whatever-tion relevancy issue”

“Ok but you broke this other thing I thought was fixed!”



<reiterates that he’s a search expert not a doctor>

<reiterates she’s a paying client>



“ok let me see what I can do.... I’ll get back to you in a week”

Our Problems

- People Problem: **Our collaboration stinks**
 - “Throw problems over the fence... wait a week”
 - Siloed by expertise (search vs content experts)
 - Potential for politics, anger, frustration (all paths to the dark side)
- Technical Problem: Search testing is hard
 - Small set of relevancy rules impacts all user searches
 - Therefore: Much easier to have regressions than in software
 - Very easy to have regressions

Our Problems

- In short, **Broken Workflow**:
 - Iterations between devs and non-technical experts take a long time
 - Devs need immediate access to non-technical expertise to make rapid progress
 - Gather broken searches
 - Rate existing searches
 - Find searches that have slid backwards in quality
 - Non-technical experts clearly need devs
 - Translate business rules to relevancy rules
 - Bending sophisticated token matching engine to our approximate user intent

Our Problems

- Our lack of collaboration means our testing stinks
 - Need expert feedback to test

“this is good, this is bad, this is ok....”



- Search devs often don't know good search! They need help.



“I need an army of Renas locked in a room telling me what is good and bad”

Solutions?

- In s/w development -- automated testing is often away to collaborate
 - Devs Can sit together with content experts and ask:
 - What should happen in this case?
 - Then record that in the form of a test

```
@Given("tab $asciiTab")
public void tab(String asciiTab) {
    tab = new Tab(asciiTab, new TabParser());
}
```

```
@When("the guitar plays")
public void guitarPlays() {
    guitar = new Guitar();
    guitar.play(tab);
}
```

```
@Then("the following notes will be played $notes")
public void theFollowingNotesWillBePlayed(String notes) {
    ensureThat(expectedNotes(notes), arePlayedBy(guitar));
}
```

Test Driven Development with Search

- Collaborative testing is absolutely essential for search
 - Good search is best defined by experts in the content:
 - Marketing, sales, users, etc



I'm a search expert! Not a Doctor!
How can I possibly measure search

- Unfortunately there's nothing to help content experts communicate with search devs around search (frankly this is rather shocking to me)

Help me help you! I have few ways to
record, measure, and evaluate search



Test Driven Development with Search

- Collaborative testing is absolutely essential for search
 - Every change to search relevancy *will* cause other queries to change
 - MUST know how much our relevancy has changed



“I fixed your search... Does it matter that toe fungus query changed by 30%?”

“Yeah lets see if we can work together to balance the two relevancy concerns”



“I’m glad we have tests to know what’s changed!”



Test Driven Development with Search

- Apply these ideas to search quality:
 - Given query Y
 - What documents are awesome/good/ok/bad/terrible?
- Record these ratings somewhere
- Observe the changes of all queries simultaneously as we modify relevancy params



Now I've got the ultimate relevancy workbench. I can see if my ideas are working or failing right away!

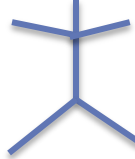
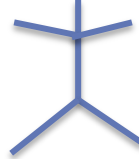
Now I can see instantly if Doug's changes are making progress or if we're moving backwards!



Quepid!

- We built a tool (a product!) around these ideas
- Now our favorite relevancy workbench

Try Quepid!
<http://quepid.com>



“Hey we’re kicking butt and taking names on these search queries!”

Test Driven Development with Quepid

Search Quality is about Collaboration*

**and collaboration is about testing*

Demo time

The screenshot shows the Quepid search interface. At the top, there is a navigation bar with "Select Case: Grocery Store" and a "Developer" link. Below this is a search bar with "Add a Query" and a "Go!" button. The main content area displays a list of search cases:

- 52 **goat milk +**
- 0 **bottled water +**
- 30 **milk -**

Each case has a small "x" icon to its right. Below the list, there are tabs for "List View" and "Image View". The "Image View" is selected, showing a detailed view of a search result for "Vitamin D Goat Milk". The result includes a small image of the product, a rating of 3, and the following details:

- maxPrice: 3.79
- margin: 0.46
- score: 2.6786919

At the bottom of the result, there is a keyboard layout showing the numbers 1-10, a "RESET" button, and a search bar containing "lk".

