

QueryTracker: An Agent for Tracking Persistent Information Needs

Gabriel L. Somlo

and

Adele E. Howe

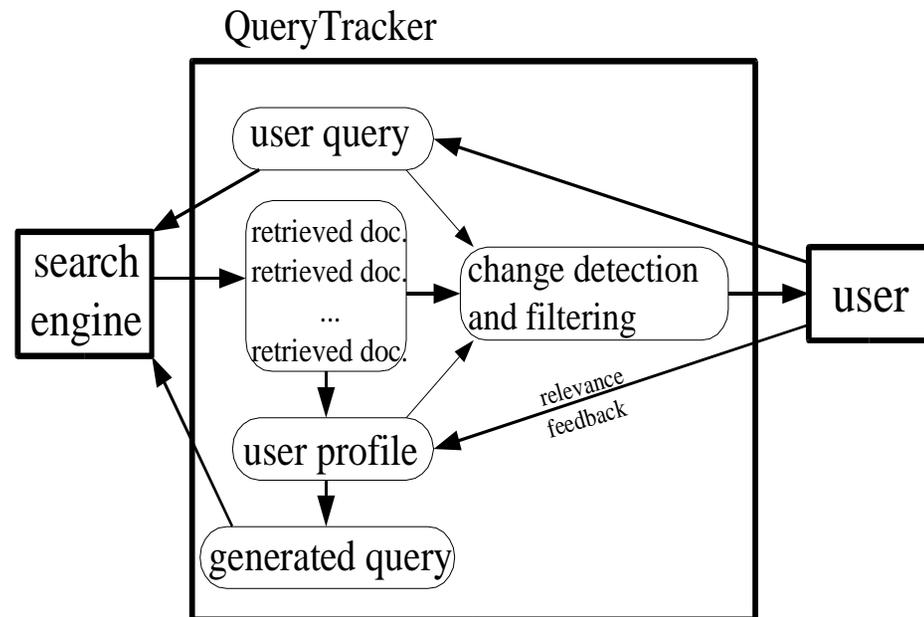
Computer Science Dept.
Colorado State University

Motivation

- Augmenting functionality of search engines
- Search engines do not help with ongoing, longterm information needs
- Search engine result quality dependent on quality of query

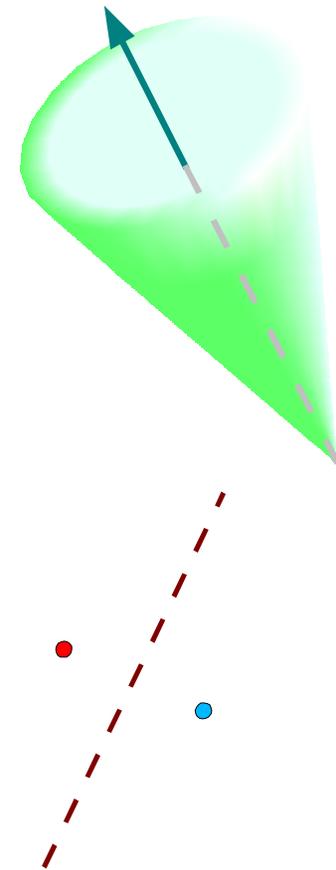
QueryTracker

- Submit query to search engine repeatedly over time
- Disseminate docs that are new and relevant, or have relevant changes
- Learn profile from user feedback
- Generate queries to supplement pool of candidate documents



User Profile and Filtering Options

- TFIDF: vector + dissemination threshold
 - Learned from positive feedback
 - Disseminate if threshold exceeded, otherwise use decaying probability
- Bayes:
 - Two classes — disseminate if closer to positive class
 - Requires both positive and negative feedback



Query Generation Method

- Top 4 by TFIDF weight from profile

$$w(t, \text{doc}) = \text{TF}_{\text{doc}, t} \cdot \log\left(\frac{D}{\text{DF}_t}\right)$$

- Previous study favored probabilistic approach
 - requires negative feedback, so we did not use it
 - pick terms based on Bayes conditional probabilities

$$w(t) = \frac{p(t|+)}{p(t|-)}$$

Change Detection

- Must identify relevant changes in content
- Profile (TFIDF or Bayes) is bag-of-words and therefore layout-insensitive
- Using TF vector difference between versions

Change Detection (continued)

1. Compute the TF vector difference:

$$V_{\text{diff}} = V_{\text{new}} - V_{\text{orig}}$$

2. Separate insertions from deletions:

$$V_{\text{diff}} = V_{\text{ins}} - V_{\text{del}}$$

where

$V_{\text{ins}} \supset$ positive terms of V_{diff}

$V_{\text{del}} \supset$ negative terms of V_{diff}

We currently use the *insert* vector only !

Dissemination Criteria

- Compute:
 - Search engine rank: Map rank to $[0,1]$
 - Similarity to original user query
 - Similarity to TFIDF profile (only if relevant feedback exists!)
 - Bayes classifier (only if both relevant and non-relevant feedback exists!)
- If any pass their threshold, then disseminate.

Experiment

- 10 distinct queries from 4 users
- Submitted daily by QueryTracker to Google
- Results rated *relevant* | *non-relevant*, or ignored
- Ignored results considered unread, hence still *new* during following iteration
- TFIDF profile built after first relevant feedback
- Bayes profile built after at least one relevant *and* one non-relevant feedback

Feedback Form Screenshot

QueryTracker: user=laura; view=sw0

Note 1: Select *Yes* for **Relevant?** if you thought the document was relevant and want it added to the query profile. This will be used to help focus future searches on your topic of interest.

Note 2: Select *No* for **Relevant?** if you didn't think the document was relevant. This will let QueryTracker know that you **did** read the document and weren't interested.

Note 3: Unless you select *Yes* or *No* for **Relevant?**, QueryTracker assumes you didn't read the document, and will keep disseminating it to you for as long as it is found by the search engine.

Note 4: When you are done making your selections, press the **Feedback** button at the bottom of the page to submit them to QueryTracker. You can do this more than once. If you make multiple conflicting selections on the same document, all but the last selection for that document will be ignored.

Note 5: If you encounter the same document content under a different URL, please give it the same rating you gave to the original document. This will help future studies on redundancy detection.

Terms: squeaky wheel optimization;		Generated on Jan 18, at 01:08
Relevant?	Status	Link
<input type="radio"/> Y <input type="radio"/> N	New	http://citeseer.nj.nec.com/context/425569/107496
<input type="radio"/> Y <input type="radio"/> N	New	http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/jair/pub/volume10/joslin99a.html/
<input type="radio"/> Y <input type="radio"/> N	New	http://www.internetnews.com/IAR/article.php/2217391
<input type="radio"/> Y <input type="radio"/> N	New	http://smw.internet.com/symm/voices/bband/
<input type="radio"/> Y <input type="radio"/> N	Chg	http://www.channelseven.com/newsbeat/execs/execs_accounts0903.shtml

(Update profile with selected relevant documents)

[Home](#)

Data Collected

- For each disseminated document:
 - Query used to retrieve: **original** or **generated**
 - Dissemination method: **rank**, **query**, **tfidf**, or **nbc**
- For each combination of query type and dissemination method:
 - hits (RF) – relevant documents disseminated (found)
 - false pos. (NF) – non-rel. documents disseminated
 - false neg. (RM) – relevant documents missed
 - Note: False negatives are relative to other methods!

Performance metrics

- Recall

$$\text{recall} = \frac{RF}{RF + RM}$$

- Precision

$$\text{precision} = \frac{RF}{RF + NF}$$

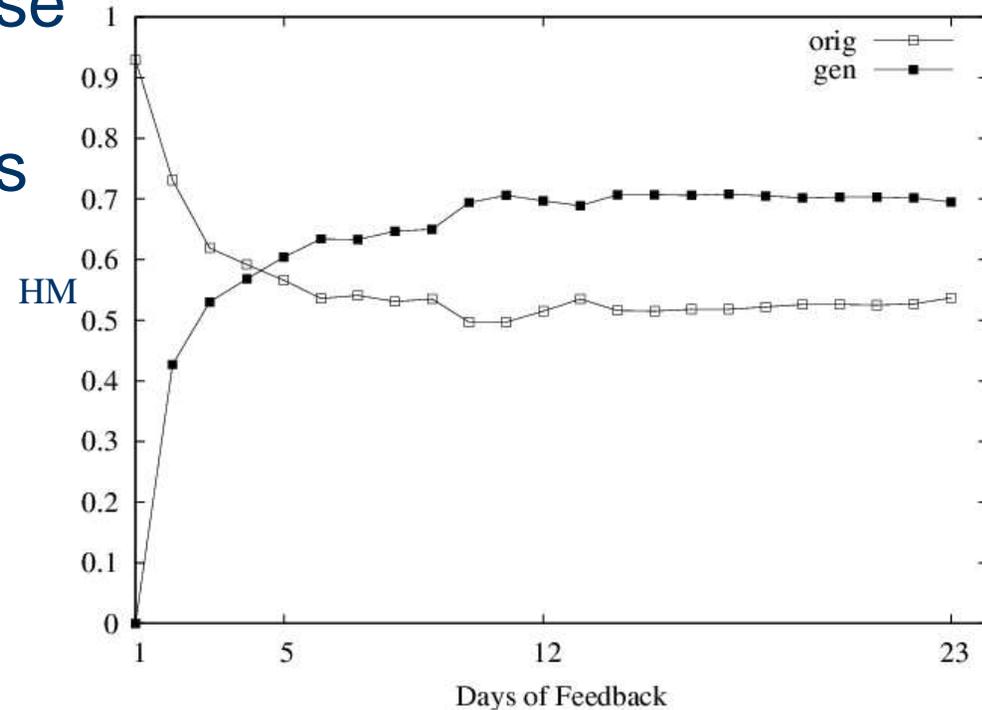
- Harmonic Mean

$$\text{HM} = \frac{2 \cdot \text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}}$$

Does Query Generation Help?

- Overall, generated queries perform worse than original ones (esp. when original is well-focused)
- But ... produced hits not found by original query

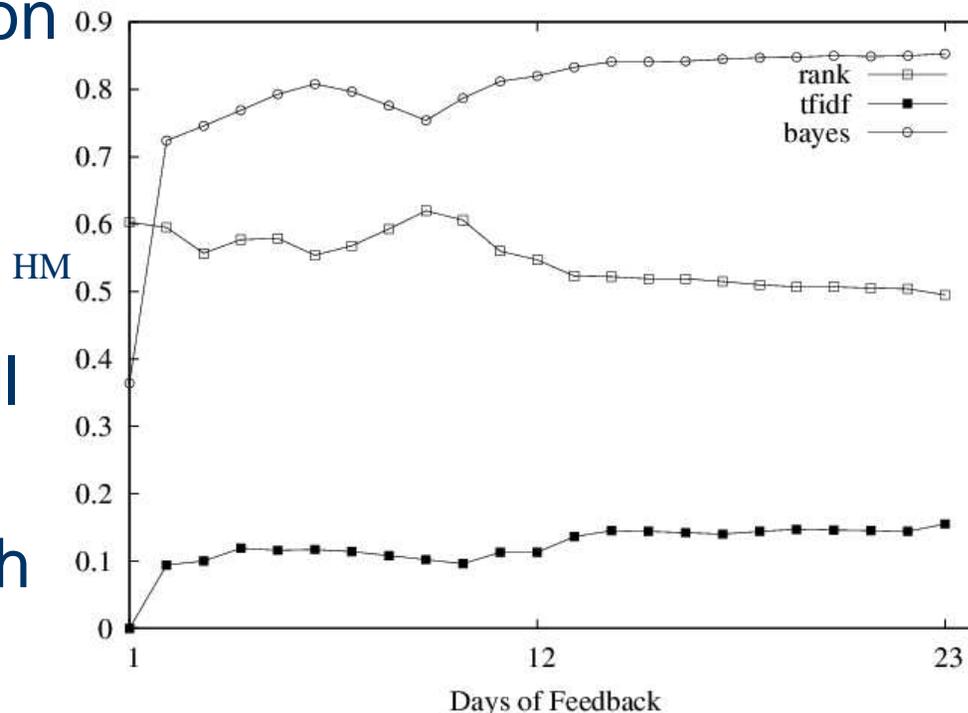
Counterexample: query #10



Which Dissemination Method Works Best?

- Comparable precision
- Search rank is mediocre but consistent
- TFIDF has low recall due to slow learning
- Bayes is fastest, with best performance

Example (avg. over q. gen.):



Conclusions

- Query generation
 - Helps less focused queries
 - Requires significant feedback to become useful
 - Improves recall
- Filtering/Dissemination
 - Rank is OK in the absence of feedback
 - TFIDF is slow-learning
 - Bayes works well but requires negative feedback
- Working on Bayes-based query generation
- Working on positive/negative TFIDF profile

You can use QueryTracker, too!



<http://www.cs.colostate.edu/~somlo/QueryTracker/>