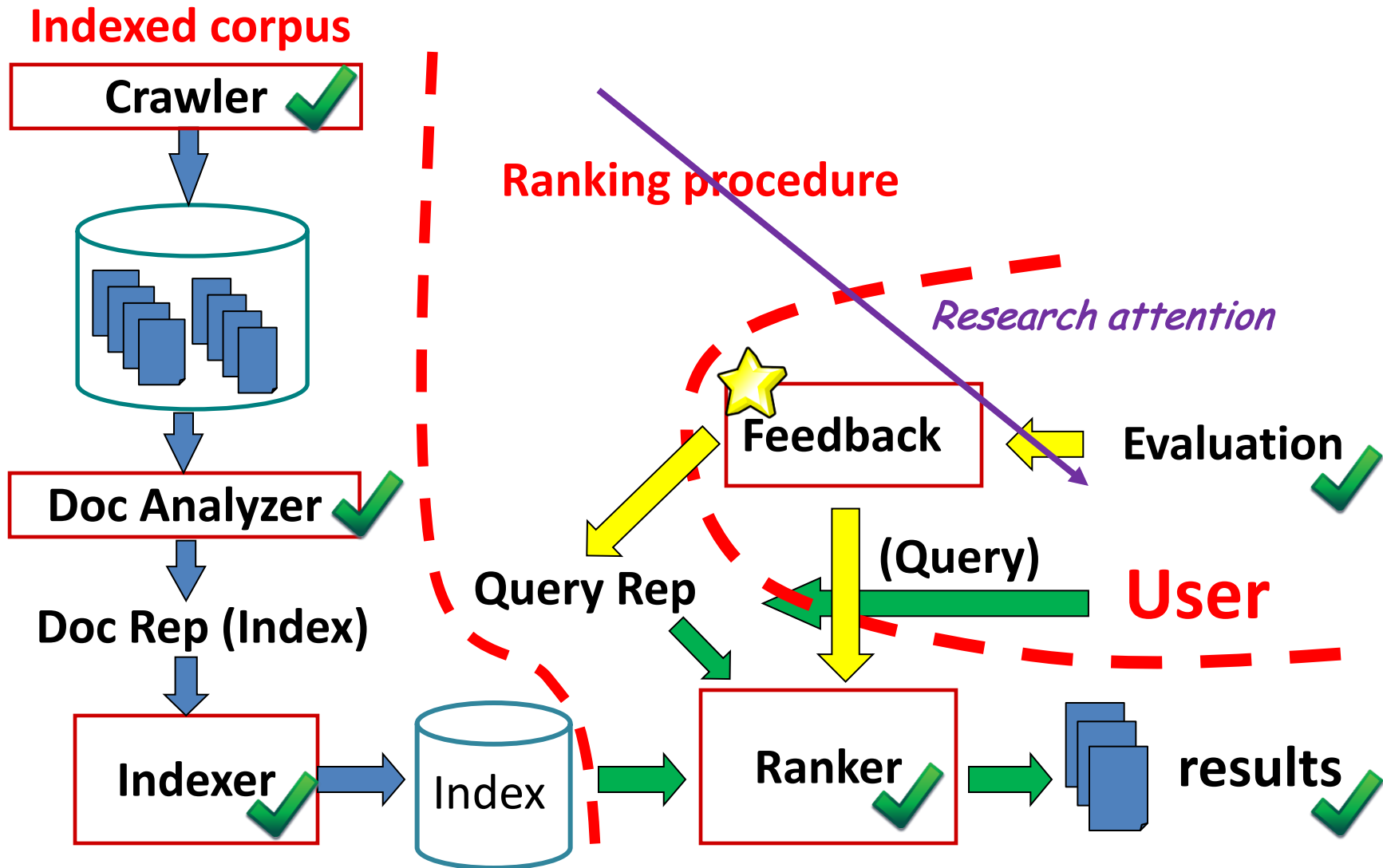


# Relevance Feedback

Hongning Wang

CS@UVa

# What we have learned so far

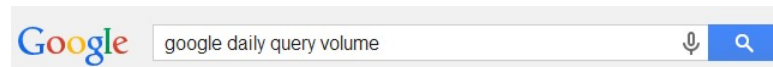


# User feedback

should be

- An IR system ~~is~~ an interactive system

Query



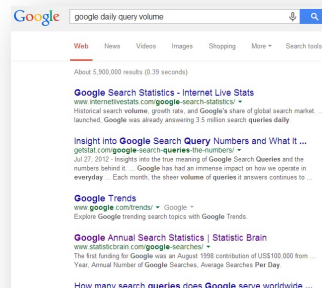
*Information need*



**Feedback**

**GAP!**

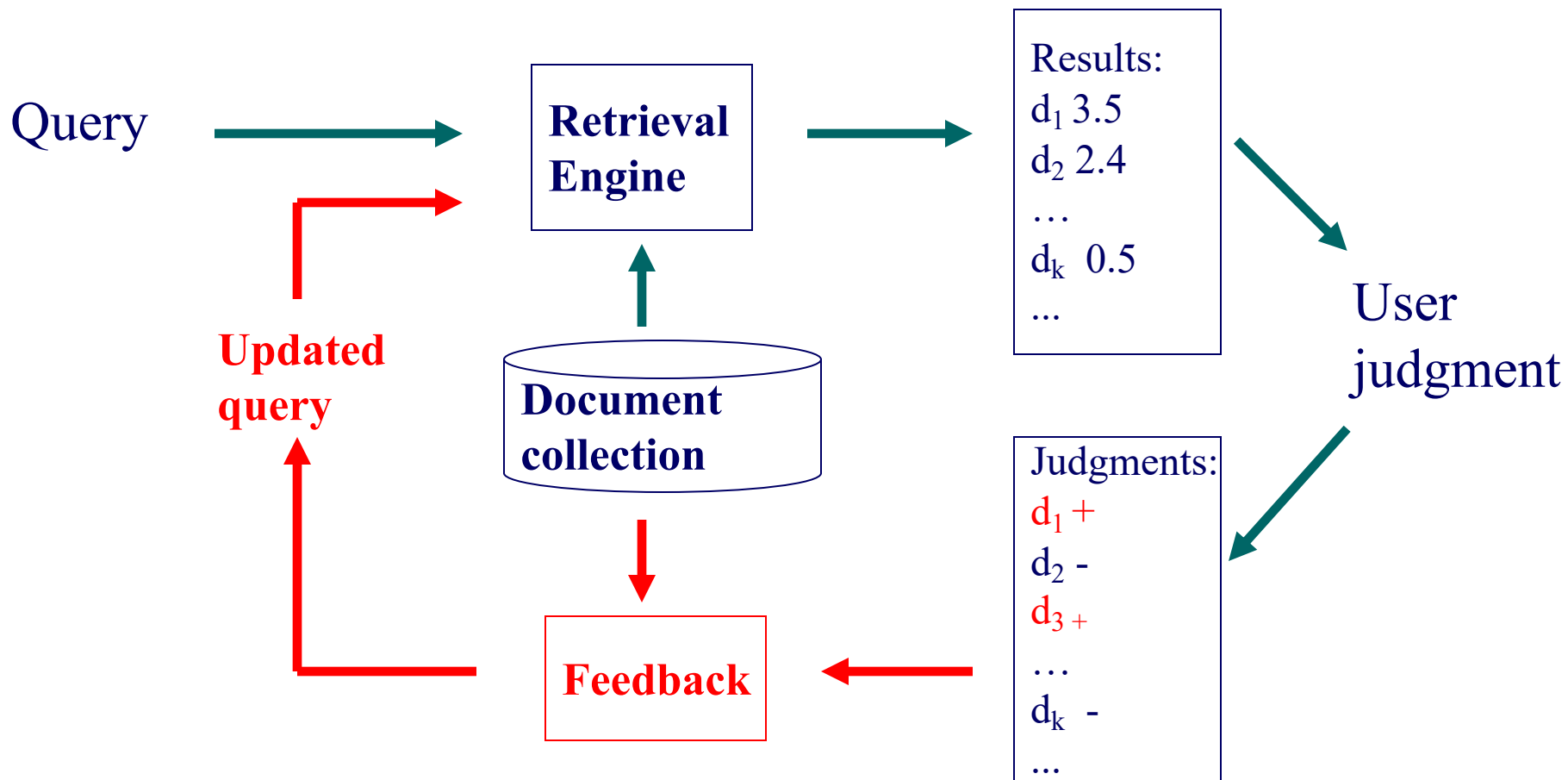
Ranked documents



*Inferred information need*



# Relevance feedback



# Basic idea in feedback

- Query expansion
  - Feedback documents can help discover related query terms
  - E.g., query=“information retrieval”
    - Relevant docs may likely share very related words, such as “search”, “search engine”, “ranking”, “query”
    - Expand the original query with such words will increase recall and sometimes also precision

# Basic idea in feedback


- Learning-based retrieval
  - Feedback documents can be treated as supervision for ranking model update
  - Covered in the lecture of “learning-to-rank”

# Relevance feedback in real systems

- Google used to provide such functions

[Personalization](#) - Wikipedia, the free encyclopedia  

**Personalization** involves using technology to accommodate the differences between individuals. Once confined mainly to the Web, it is increasingly becoming a ...

[en.wikipedia.org/wiki/Personalized](#) - 42k - [Cached](#) - [Similar pages](#) - 

Relevant

[Personalized Gifts from Personalization Mall](#)  

It shows you went out of your way to find the perfect gift and to **personalize** it to make it theirs alone! At PersonalizationMall.com, we design most of our ...

[www.personalizationmall.com/Default.aspx?&did=111028](#) - 47k -

[Cached](#) - [Similar pages](#) - 

Nonrelevant

[What is personalization?](#) - a definition from Whatis.com  

Mar 6, 2007 ... On a Web site, **personalization** is the process of tailoring pages to individual users' characteristics or preferences.

[searchcrm.techtarget.com/sDefinition/0,,sid11\\_gci532341,00.html](#) - 72k -







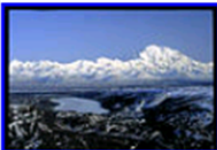



[Cached](#) - [Similar pages](#) - 

– Guess why?

# Relevance feedback in real systems

- Popularly used in image search systems

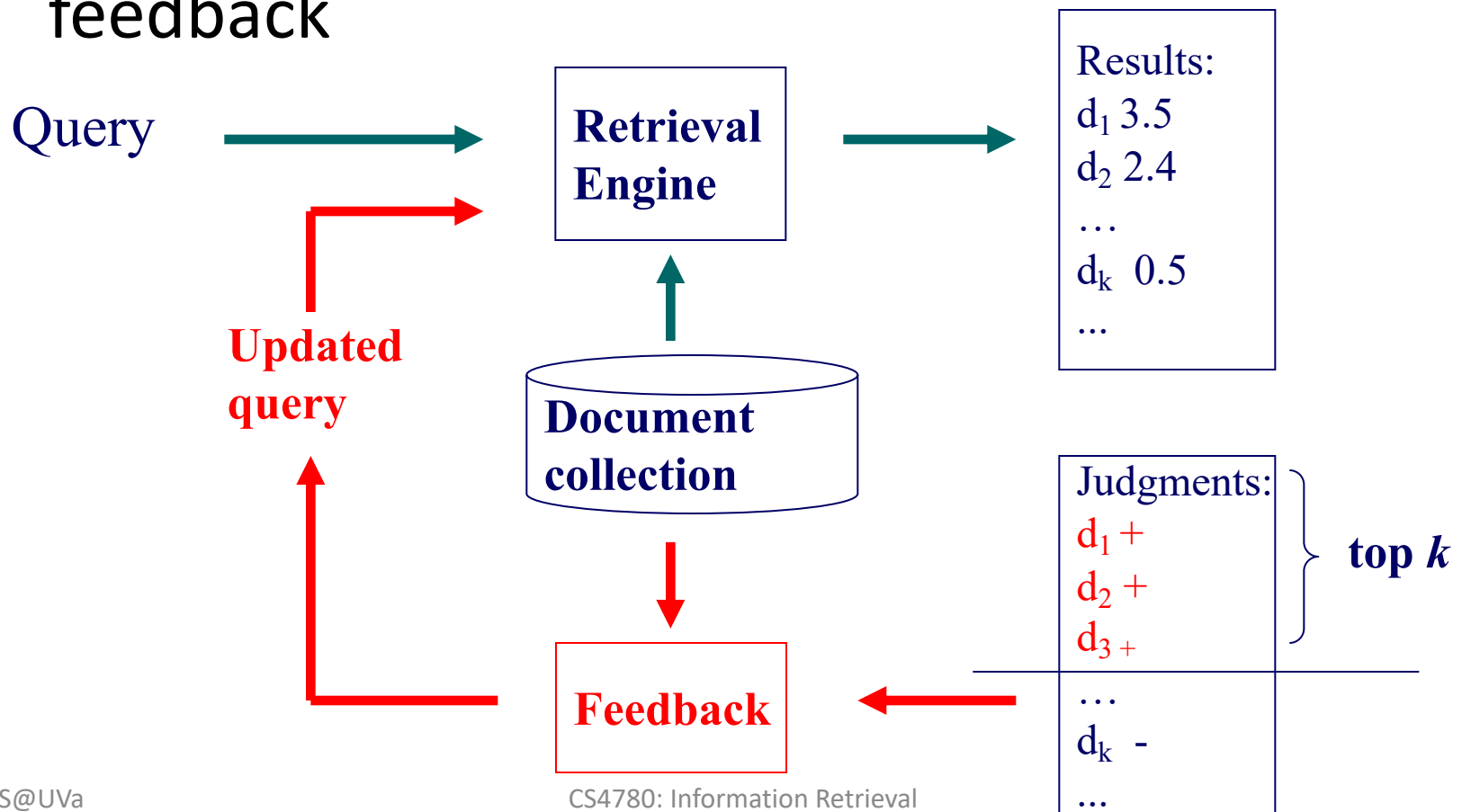
**Result:**

 <p>Similarity: 1.387633 <b>Query Image</b></p> <p><a href="#">top</a></p>	 <p>Similarity: 0.440483</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.352732</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.346222</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.345664</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>
 <p>Similarity: 0.340732</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.332161</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.329942</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.325042</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>	 <p>Similarity: 0.323497</p> <p><input type="button" value="neutral"/> ▾</p> <p><a href="#">top</a></p>



# Pseudo relevance feedback

- What if the users are reluctant to provide any feedback



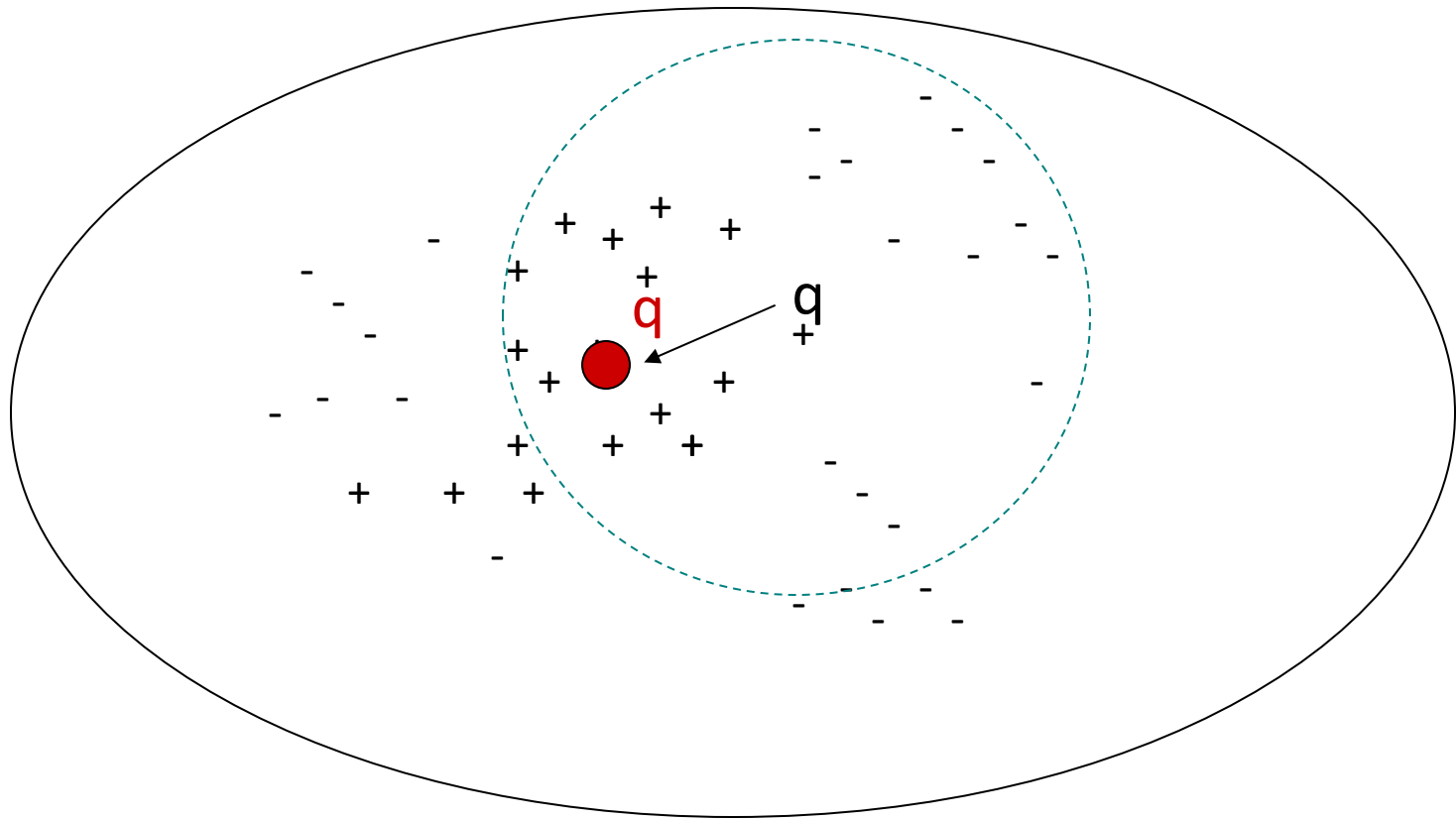
# Feedback techniques

- Feedback as query expansion
  - Step 1: Term selection
  - Step 2: Query expansion
  - Step 3: Query term re-weighting
- Feedback as training signal
  - Covered in learning to rank

# Relevance feedback in vector space models

- General idea: query modification
  - Adding new (weighted) terms
  - Adjusting weights of old terms
- The most well-known and effective approach is Rocchio [Rocchio 1971]

# Illustration of Rocchio feedback



# Formula for Rocchio feedback

- Standard operation in vector space

**Modified query**

**Parameters**

$$\vec{q}_m = \alpha \vec{q} + \frac{\beta}{|D_r|} \sum_{\forall \vec{d}_i \in D_r} \vec{d}_i - \frac{\gamma}{|D_n|} \sum_{\forall \vec{d}_j \in D_n} \vec{d}_j$$

**Original query**

**Rel docs**

**Non-rel docs**

# Rocchio in practice

- Negative (non-relevant) examples are not very important (why?)
- Efficiency concern
  - Restrict the vector onto a lower dimension (i.e., only consider highly weighted words in the centroid vector)
- Avoid “training bias”
  - Keep relatively high weight on the original query
- Can be used for relevance feedback and pseudo feedback
- Usually robust and effective

# Feedback in probabilistic models

**Classic Prob. Model**  $O(R=1|Q,D) \propto \frac{P(D|Q,R=1)}{P(D|Q,R=0)}$  ← **Rel. doc model**  
 ← **NonRel. doc model**

**Language Model**  $O(R=1|Q,D) \propto P(Q|D,R=1)$  ← **“Rel. query” model**

**Parameter Estimation**

$\left. \begin{matrix} (q_1, d_1, 1) \\ (q_1, d_2, 1) \\ (q_1, d_3, 1) \end{matrix} \right\} P(D|Q, R=1)$   
 $\left. \begin{matrix} (q_1, d_4, 0) \\ (q_1, d_5, 0) \end{matrix} \right\} P(D|Q, R=0)$   
 $\left. \begin{matrix} (q_3, d_1, 1) \\ (q_4, d_1, 1) \\ (q_5, d_1, 1) \\ (q_6, d_2, 1) \\ (q_6, d_3, 0) \end{matrix} \right\} P(Q|D, R=1)$

## Feedback:

- $P(D|Q, R=1)$  can be improved for the **current query** and **future doc**
- $P(Q|D, R=1)$  can be improved for the **current doc** and **future query**

# Robertson-Sparck Jones Model

(Robertson & Sparck Jones 76)

$$\log O(R=1|Q,D) \stackrel{\text{Rank}}{\approx} \sum_{i=1, d_i=q_i=1}^k \log \frac{p_i(1-u_i)}{u_i(1-p_i)} = \sum_{i=1, d_i=q_i=1}^k \log \frac{p_i}{1-p_i} + \log \frac{1-u_i}{u_i} \quad (\text{RSJ model})$$

Two parameters for each term  $A_i$ :

$p_i = P(A_i=1 | Q, R=1)$ : prob. that term  $A_i$  occurs in a relevant doc

$u_i = P(A_i=1 | Q, R=0)$ : prob. that term  $A_i$  occurs in a non-relevant doc

How to estimate these parameters?

Suppose we have relevance judgments,

$$\hat{p}_i = \frac{\#(\text{rel. doc with } A_i) + 0.5}{\#(\text{rel.doc}) + 1} \quad \hat{u}_i = \frac{\#(\text{nonrel. doc with } A_i) + 0.5}{\#(\text{nonrel.doc}) + 1}$$

“+0.5” and “+1” can be justified by Bayesian estimation as priors

$P(D|Q, R=1)$  can be improved for  
the *current query* and *future doc*

**Per-query estimation!**



# Feedback in language models

- Recap of language model

- Rank documents based on *query likelihood*

$$\log p(q | d) = \sum_{w_i \in q} \log p(w_i | d)$$

where,  $q = w_1 w_2 \dots w_n$

Document language model



- Difficulty

- Documents are given, i.e.,  $p(w|d)$  is fixed

# Feedback in language models

- Approach
  - Introduce a probabilistic query model
  - Ranking: measure distance between query model and document model
  - Feedback: query model update

*Q: Back to vector space model?*

*A: Kind of, but in a different perspective.*

# Kullback-Leibler (KL) divergence based retrieval model

- Probabilistic similarity measure

- $sim(q; d) \propto -KL(\theta_q || \theta_d)$

$$-\sum_w p(w|\theta_q) \log p(w|\theta_q) + \sum_w p(w|\theta_q) \log p(w|\theta_d)$$

*Query-specific quality, ignored for ranking*

*Query language model, need to be estimated*

*Document language model, we know how to estimate*

# Background knowledge

- Kullback-Leibler divergence
  - A non-symmetric measure of the difference between two probability distributions P and Q

- $KL(P||Q) = \int P(x) \log \frac{P(x)}{Q(x)} dx$

- It measures the expected number of extra bits required to code samples from P when using a code based on Q
- P usually refers to the “true” data distribution, Q refers to the “approximated” distribution

- Properties

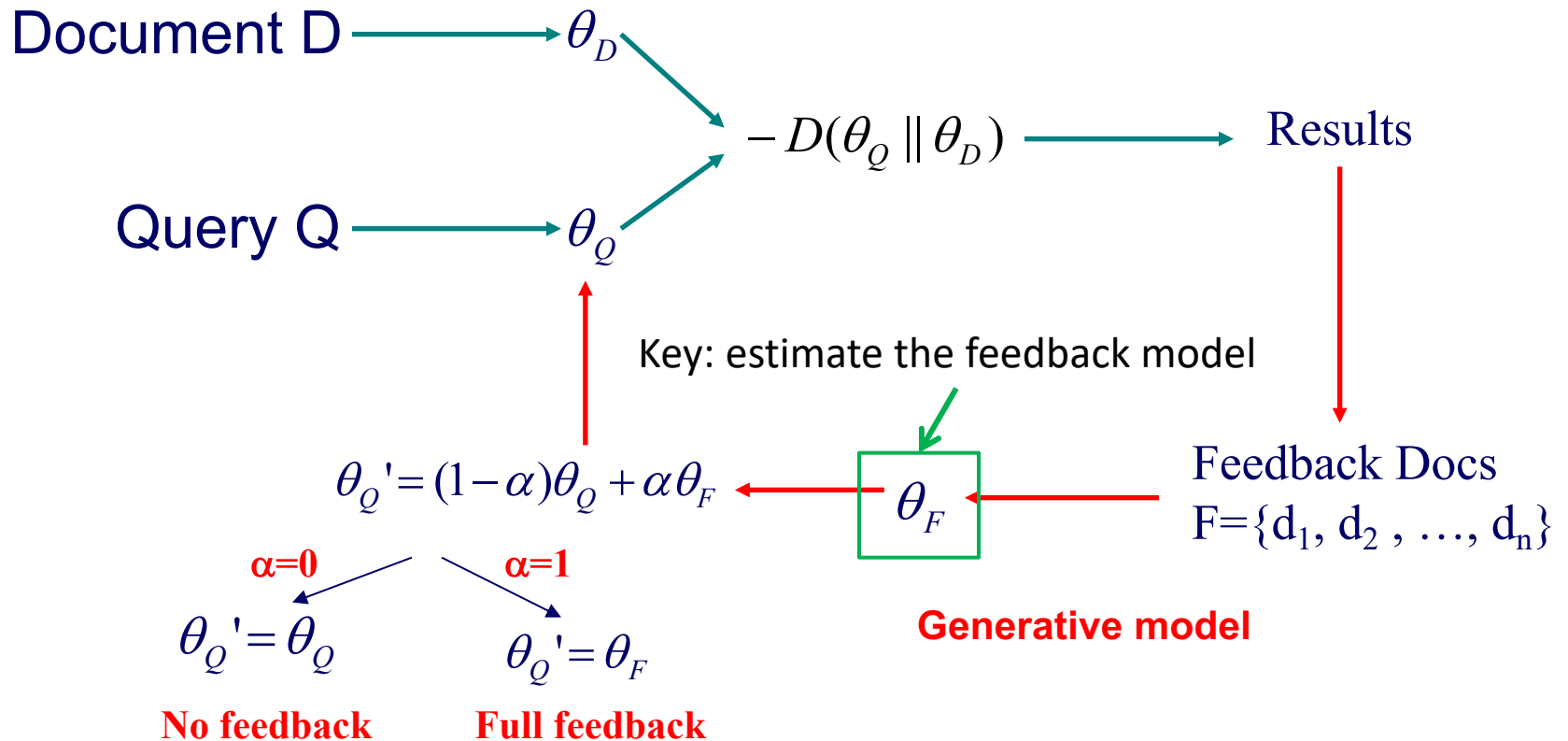
- Non-negative
- $KL(P||Q) = 0$ , iff  $P = Q$  almost everywhere

*Explains why  $sim(q; d) \propto -D(\theta_q || \theta_d)$*

# Kullback-Leibler (KL) divergence based retrieval model

- Retrieval  $\approx$  estimation of  $\theta_q$  and  $\theta_d$ 
  - $Rel(q; d) \propto$   
 $\sum_{w \in d, p(w|\theta_q) > 0} p(w|\theta_q) \log \frac{p(w|d)}{\alpha_d p(w|C)} + \log \alpha_d$ 
    - same smoothing strategy* (with arrow pointing to  $\alpha_d$ )
  - A generalized version of query-likelihood language model
    - $p(w|\theta_q)$  is the empirical distribution of words in a query


# Feedback as model interpolation



*Q: Rocchio feedback in vector space model?*

*A: Very similar, but with different interpretations.*

# Feedback in language models

airport security 

## [Transportation Security Administration - Official Site](#)

[www.tsa.gov](#) Official site  
Charged with providing effective and efficient security for passenger and freight transportation in the United States. Mission, press releases, employment, milestones ...

### [Prohibited Items](#)

The My TSA mobile application provides 24/7 access to helpful ...

### [TSA Precheck Ad](#)

Learn about TSA Pre™ expedited screening! No longer remove ...

### [Careers](#)

TSA is comprised of nearly 50,000 security officers, inspectors, air ...

See results only from tsa.gov

### [3-1-1 for Carry-ons](#)

Consolidating these containers in the small bag separate from your ...

### [Traveler Information](#)

One of the primary goals of the Transportation Security ...

### [Acceptable IDs](#)

Adult passengers (18 and over) must show a valid U.S. federal or state ...



## [Airport security - Wikipedia, the free encyclopedia](#)

[en.wikipedia.org/wiki/Airport\\_security](#)  
Airport security refers to the techniques and methods used in protecting passengers, staff and aircraft which use the airports from accidental/malicious harm, crime ...  
Airport enforcement ... - Process and equipment - Notable incidents



## [An Overview of Airport Security Rules - About studenttravel.about.com](#)

Student Transportation Options  
Airport security rules are a travel drag: get through airport security and get to the fun part (travel!) faster by knowing what the airport security rules are in advance.

## [News about Airport Security](#)

[bing.com/news](#)

[No need to beef up airport security: govt](#)

YahooNews · 1 minute ago

Airport security doesn't need to be strengthened because 30 to 40 New Zealanders are being monitored over links to terrorist groups, the government says. Prime Minister John Key on Wednesday revealed the existence of...

## Feedback documents

### [Airport security - Wikipedia, the free encyclopedia](#)

[en.wikipedia.org/wiki/Airport\\_security](#)

Airport security refers to the techniques and methods used in protecting passengers, staff and aircraft which use the airports from accidental/malicious harm, crime ...

Airport enforcement ... - Process and equipment - Notable incidents

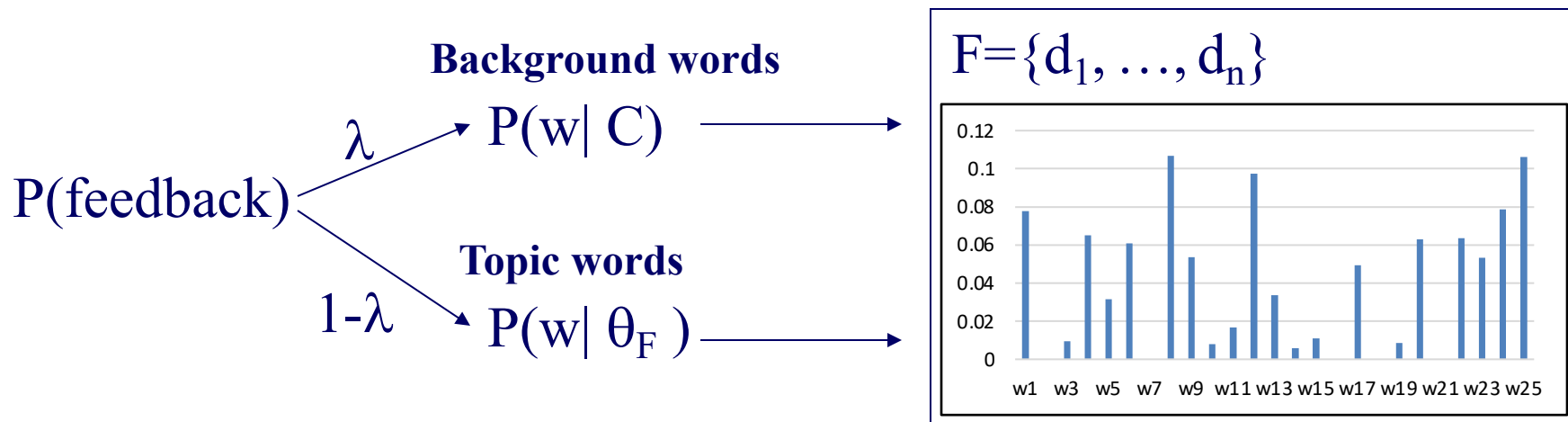
### [An Overview of Airport Security Rules - About studenttravel.about.com](#)

Student Transportation Options

Airport security rules are a travel drag: get through airport security and get to the fun part (travel!) faster by knowing what the airport security rules are in advance.

*protect passengers,  
accidental/malicious  
harm, crime, rules*

# Generative mixture model of feedback



$$\log p(d_F) = \sum_{d,w} c(w,d) \log[(1-\lambda)p(w|\theta_F) + \lambda p(w|C)]$$

$\lambda$  = **Noise ratio in feedback documents**

**Maximum Likelihood**  $\bar{\theta}_F = \operatorname{argmax}_{\theta} \log p(d_F)$



# How to estimate $\theta_F$ ?

**Known**  
Background  
 $p(w|C)$

the 0.2  
a 0.1  
we 0.01  
to 0.02  
...  
flight 0.0001  
company  
0.00005  
...

**Unknown**  
query topic  
 $p(w|\theta_F)=?$

“airport security”

...  
accident =?  
regulation =?  
passenger =?  
rules =?  
...

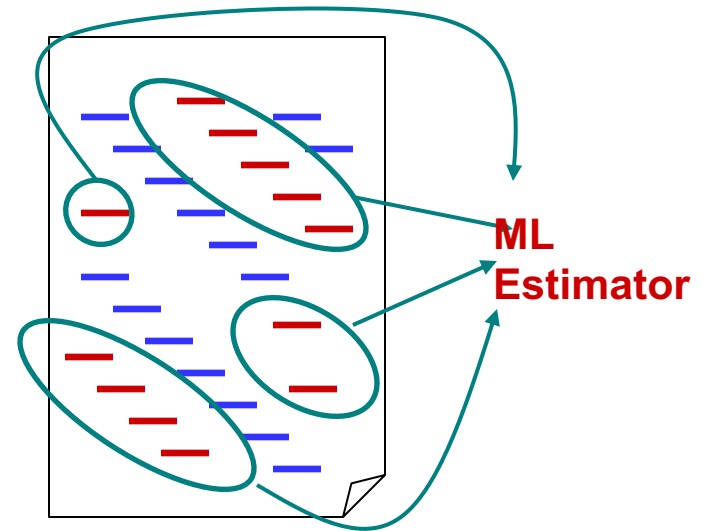
fixed



$\lambda=0.7$



**Feedback**  
Doc(s)



$\lambda=0.3$



Suppose, we know the identity of each word; **but we don't...**

# Appeal to Expectation Maximization algorithm

Identity (“hidden”) variable:  $z_i \in \{1 \text{ (background)}, 0 \text{ (topic)}\}$

	$z_i$
the	1
paper	1
presents	1
a	1
text	0
mining	0
algorithm	0
the	1
paper	0
...	...

Suppose the parameters are all known, what’s a reasonable guess of  $z_i$ ?

- depends on  $\lambda$  (why?)
- depends on  $p(w|C)$  and  $p(w|\theta_F)$  (how?)

$$p(z_i = 1 | w_i) = \frac{p(z_i = 1)p(w_i | z_i = 1)}{p(z_i = 1)p(w_i | z_i = 1) + p(z_i = 0)p(w_i | z_i = 0)}$$

$$= \frac{\lambda p(w_i | C)}{\lambda p(w_i | C) + (1 - \lambda)p(w_i | \theta_F)} \quad \text{E-step}$$

$$p^{new}(w_i | \theta_F) = \frac{c(w_i, F)(1 - p^{(n)}(z_i = 1 | w_i))}{\sum_{w_j \in \text{vocabulary}} c(w_j, F)(1 - p^{(n)}(z_j = 1 | w_j))} \quad \text{M-step}$$

# A toy example of EM computation

$$p^{(n)}(z_i = 1 | w_i) = \frac{\lambda p(w_i | C)}{\lambda p(w_i | C) + (1 - \lambda) p^{(n)}(w_i | \theta_F)}$$

Expectation-Step:

Augmenting data by guessing hidden variables

$$p^{(n+1)}(w_i | \theta_F) = \frac{c(w_i, F)(1 - p^{(n)}(z_i = 1 | w_i))}{\sum_{w_j \in \text{vocabulary}} c(w_j, F)(1 - p^{(n)}(z_j = 1 | w_j))}$$

Maximization-Step

With the “augmented data”, estimate parameters using maximum likelihood

Assume  $\lambda=0.5$

Word	#	P(w C)	Iteration 1		Iteration 2		Iteration 3	
			P(w  $\theta_F$ )	P(z=1)	P(w  $\theta_F$ )	P(z=1)	P(w  $\theta_F$ )	P(z=1)
The	4	0.5	<b>0.25</b>	0.67	<b>0.20</b>	0.71	<b>0.18</b>	0.74
Paper	2	0.3	<b>0.25</b>	0.55	<b>0.14</b>	0.68	<b>0.10</b>	0.75
Text	4	0.1	<b>0.25</b>	0.29	<b>0.44</b>	0.19	<b>0.50</b>	0.17
Mining	2	0.1	<b>0.25</b>	0.29	<b>0.22</b>	0.31	<b>0.22</b>	0.31
Log-Likelihood			-16.96		-16.13		-16.02	

*Why in Rocchio we did not distinguish a word's identity?*

# Example of feedback query model

*Open question: how do we handle negative feedback?*

- Query: “airport security”
  - Pseudo feedback with top 10 documents

$\lambda=0.7$

W	$p(W \theta_F)$
<b>the</b>	0.0405
<b>security</b>	0.0377
<b>airport</b>	0.0342
<b>beverage</b>	0.0305
<b>alcohol</b>	0.0304
<b>to</b>	0.0268
<b>of</b>	0.0241
<b>and</b>	0.0214
<b>author</b>	0.0156
<b>bomb</b>	0.0150
<b>terrorist</b>	0.0137
<b>in</b>	0.0135
<b>license</b>	0.0127
<b>state</b>	0.0127
<b>by</b>	0.0125

$\lambda=0.9$

W	$p(W \theta_F)$
<b>security</b>	0.0558
<b>airport</b>	0.0546
<b>beverage</b>	0.0488
<b>alcohol</b>	0.0474
<b>bomb</b>	0.0236
<b>terrorist</b>	0.0217
<b>author</b>	0.0206
<b>license</b>	0.0188
<b>bond</b>	0.0186
<b>counter-terror</b>	0.0173
<b>terror</b>	0.0142
<b>newsnet</b>	0.0129
<b>attack</b>	0.0124
<b>operation</b>	0.0121
<b>headline</b>	0.0121

# What you should know

- Purpose of relevance feedback
- Rocchio relevance feedback for vector space models
- Query model based feedback for language models

# Today's reading

- Chapter 9. Relevance feedback and query expansion
  - 9.1 Relevance feedback and pseudo relevance feedback
  - 9.2 Global methods for query reformulation