

Introduction to  
**Information Retrieval**

Lecture 10: Relevance Feedback & Query  
Expansion

# Overview

---

- ① Motivation
- ② Relevance feedback: Basics
- ③ Relevance feedback: Details
- ④ Query expansion

# Outline

---

- ① Motivation
- ② Relevance feedback: Basics
- ③ Relevance feedback: Details
- ④ Query expansion

# How can we improve recall in search?

---

- As an example consider query  $q$ : [aircraft] ...
- ... and document  $d$  containing “plane”, but not containing “aircraft”
- A simple IR system will not return  $d$  for  $q$ .
- Even if  $d$  is the most relevant document for  $q$ !
- We want to change this:
- Return relevant documents even if there is no term match with the (original) query

# Recall

---

- Loose definition of recall in this lecture: “increasing the number of relevant documents returned to user”
- Two ways of improving recall: **relevance feedback** and **query expansion**

# Options for improving recall

---

- Local: Do a “local”, on-demand analysis for a user query
  - Main local method: [relevance feedback](#)
  - Part 1
  
- Global: Do a global analysis once (e.g., of collection) to produce [thesaurus](#)
  - Use thesaurus for [query expansion](#)
  - Part 2

# Outline

---

- ① Motivation
- ② Relevance feedback: Basics
- ③ Relevance feedback: Details
- ④ Query expansion

# Relevance feedback: Basic idea

---

- The user issues a (short, simple) query.
- The search engine returns a set of documents.
- **User marks some docs as relevant, some as nonrelevant.**
- Search engine computes a new representation of the information need, based on the marked documents. Hope: better than the initial query.
- Search engine runs new query and returns new results.
- New results have (hopefully) better recall.

# Relevance feedback

---

- We can iterate this: several rounds of relevance feedback.
- We will use the term **ad hoc retrieval** to refer to regular retrieval without relevance feedback.
- We will now look at an example of relevance feedback.

# Example: A real (non-image) example

Initial query:

[new space satellite applications] Results for initial query: ( $r = \text{rank}$ )

	$r$		
+	1	0.539	NASA Hasn't Scrapped Imaging Spectrometer
+	2	0.533	NASA Scratches Environment Gear From Satellite Plan
	3	0.528	Science Panel Backs NASA Satellite Plan, But Urges Launches of Smaller Probes
	4	0.526	A NASA Satellite Project Accomplishes Incredible Feat: Staying Within Budget
	5	0.525	Scientist Who Exposed Global Warming Proposes Satellites for Climate Research
	6	0.524	Report Provides Support for the Critics Of Using Big Satellites to Study Climate
	7	0.516	Arianespace Receives Satellite Launch Pact From Telesat Canada
+	8	0.509	Telecommunications Tale of Two Companies

User then marks relevant documents with “+”.

# Expanded query after relevance feedback

2.074	new	15.106	space	
30.816	satellite	5.660	application	
5.991	nasa	5.196	eos	
4.196	launch	3.972	aster	
3.516	instrument	3.446	arianespace	Compare to original
3.004	bundespost	2.806	ss	
2.790	rocket	2.053	scientist	
2.003	broadcast	1.172	earth	
0.836	oil	0.646	measure	

query: [new space satellite applications]

# Results for expanded query

---

	<i>r</i>		
*	1	0.513	NASA Scratches Environment Gear From Satellite Plan
*	2	0.500	NASA Hasn't Scrapped Imaging Spectrometer
	3	0.493	When the Pentagon Launches a Secret Satellite, Space Sleuths Do Some Spy Work of Their Own
	4	0.493	NASA Uses 'Warm' Superconductors For Fast Circuit
*	5	0.492	Telecommunications Tale of Two Companies
	6	0.491	Soviets May Adapt Parts of SS-20 Missile For Commercial Use
	7	0.490	Gaping Gap: Pentagon Lags in Race To Match the Soviets In Rocket Launchers
	8	0.490	Rescue of Satellite By Space Agency To Cost \$90 Million

# Outline

---

- ① Motivation
- ② Relevance feedback: Basics
- ③ Relevance feedback: Details
- ④ Query expansion

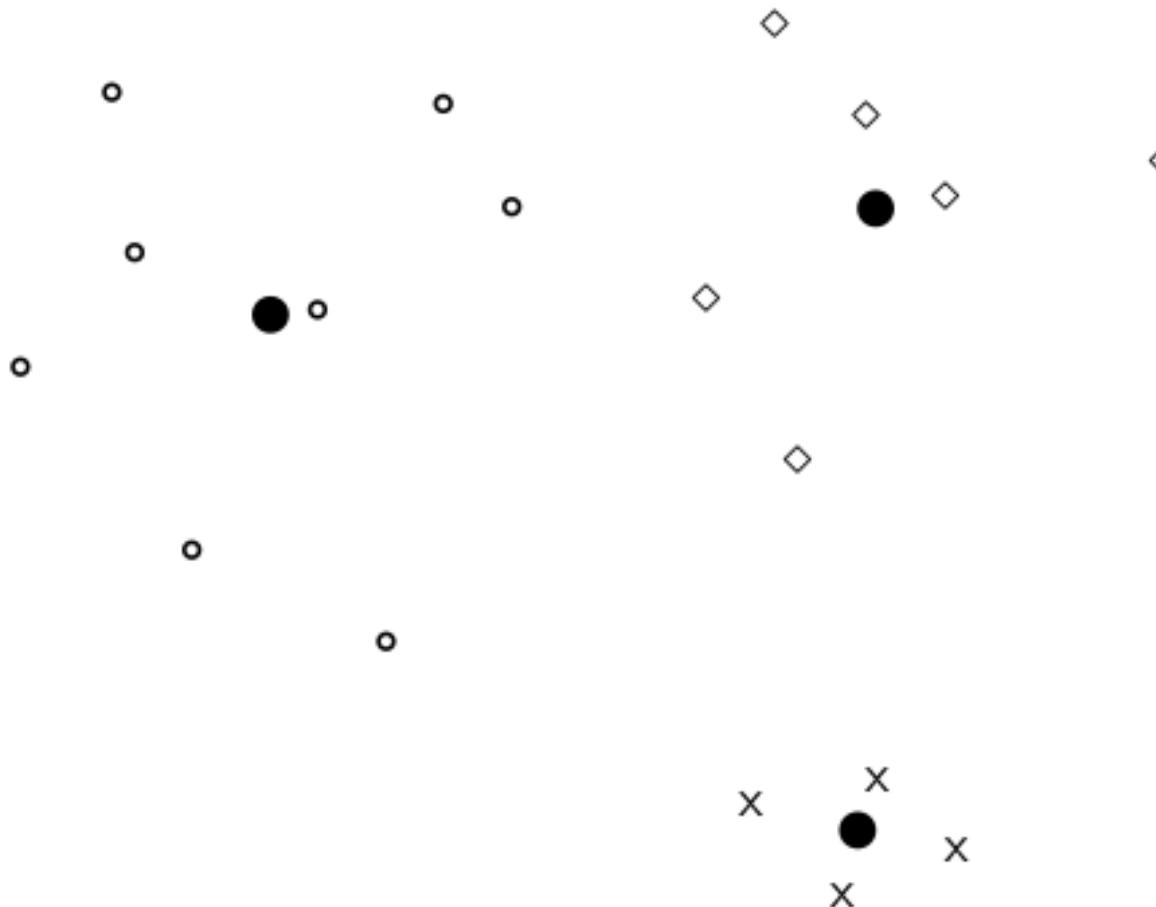
# Key concept for relevance feedback: Centroid

- The centroid is the center of mass of a set of points.
- Remember that we represent documents as points in a high-dimensional space.
- Thus: we can compute centroids of documents.
- Definition:

$$\vec{\mu}(D) = \frac{1}{|D|} \sum_{d \in D} \vec{v}(d)$$

where  $D$  is a set of documents and  $\vec{v}(d) = \vec{d}$  is the vector we use to represent document  $d$ .

# Centroid: Example



## Rocchio' algorithm

- The Rocchio' algorithm implements relevance feedback in the vector space model.
- Rocchio' chooses the query  $\vec{q}_{opt}$  that maximizes

$$\vec{q}_{opt} = \arg \max_{\vec{q}} [\text{sim}(\vec{q}, \mu(D_r)) - \text{sim}(\vec{q}, \mu(D_{nr}))]$$

$D_r$  : set of relevant docs;  $D_{nr}$  : set of nonrelevant docs

- Intent:  $q_{opt}$  is the vector that separates relevant and nonrelevant docs maximally.
- Making some additional assumptions, we can rewrite  $\vec{q}_{opt}$  as:

$$\vec{q}_{opt} = \mu(D_r) + [\mu(D_r) - \mu(D_{nr})]$$

# Rocchio' algorithm

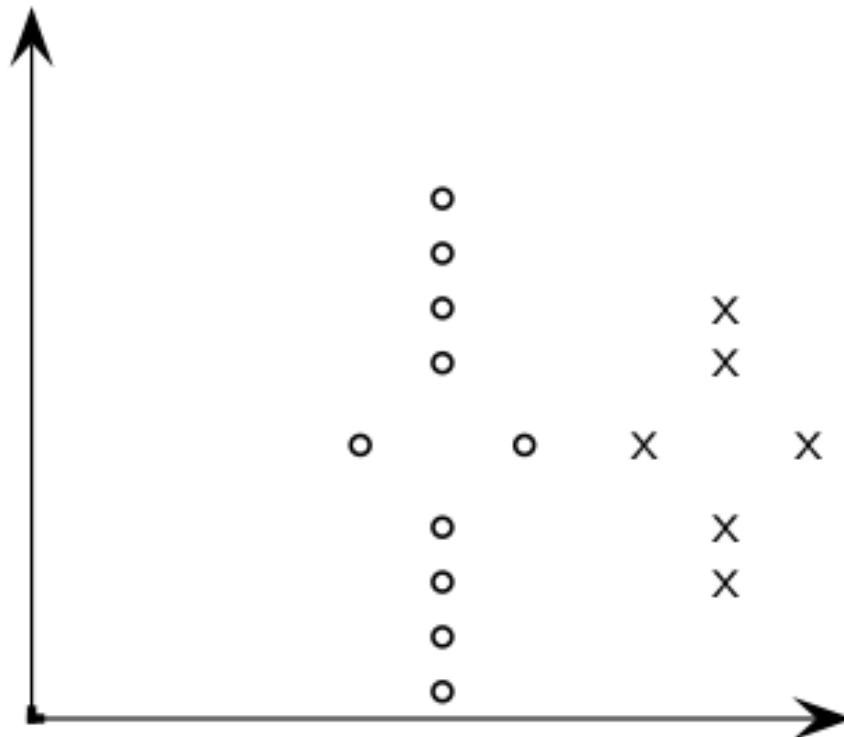
---

- The optimal query vector is:

$$\begin{aligned}\vec{q}_{opt} &= \mu(D_r) + [\mu(D_r) - \mu(D_{nr})] \\ &= \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j + [\frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j]\end{aligned}$$

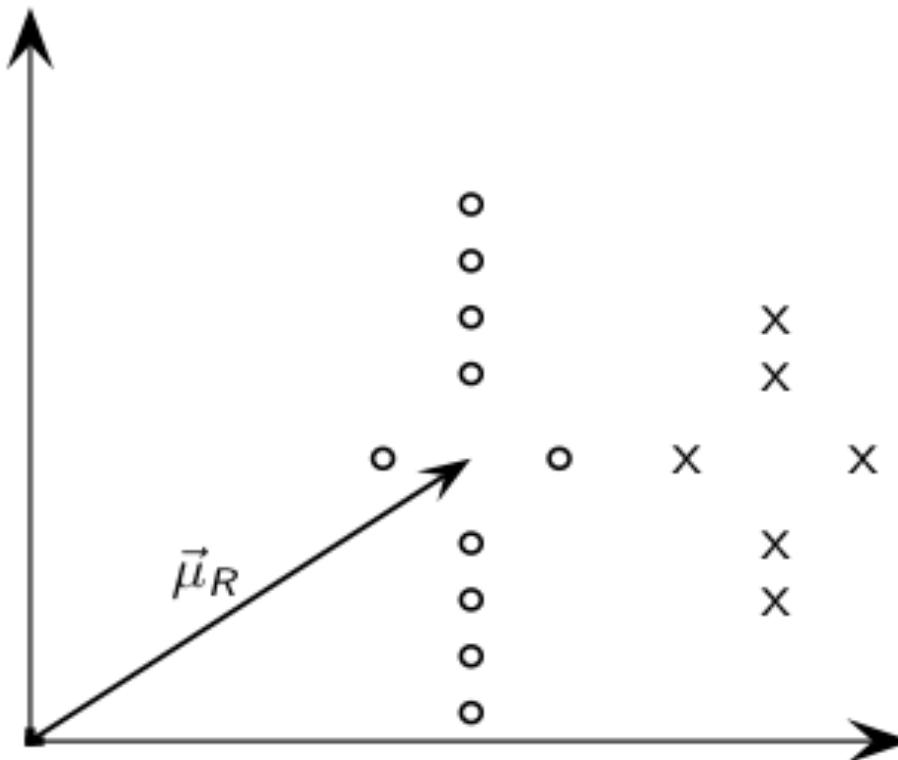
- We move the centroid of the relevant documents by the difference between the two centroids.

## Exercise: Compute Rocchio' vector



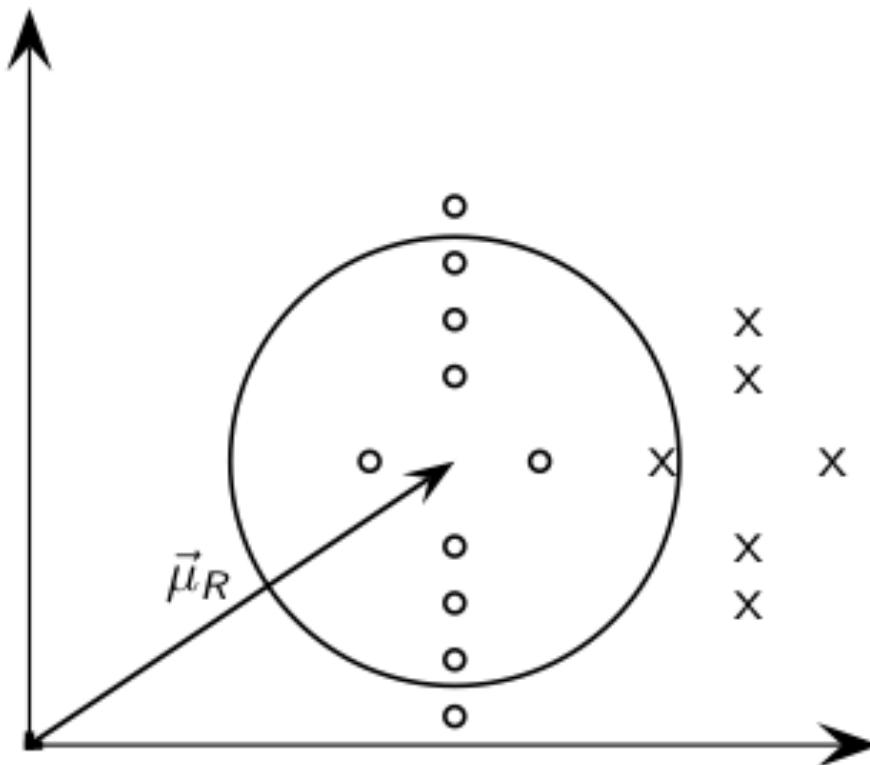
circles: relevant documents, Xs: nonrelevant documents

# Rocchio' illustrated



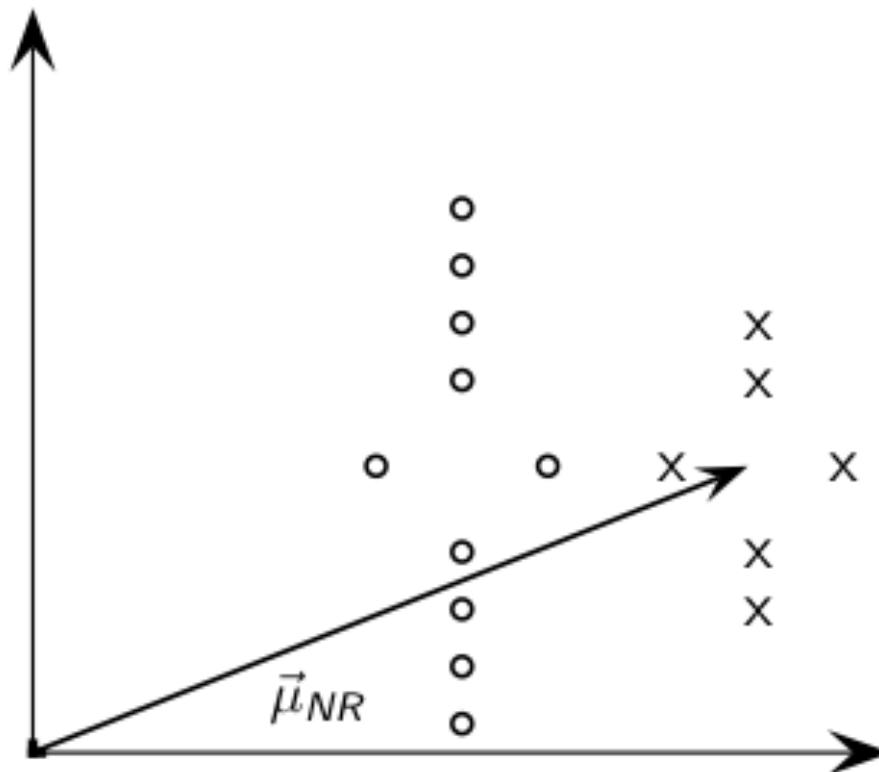
$\vec{\mu}_R$  : centroid of relevant documents

# Rocchio' illustrated



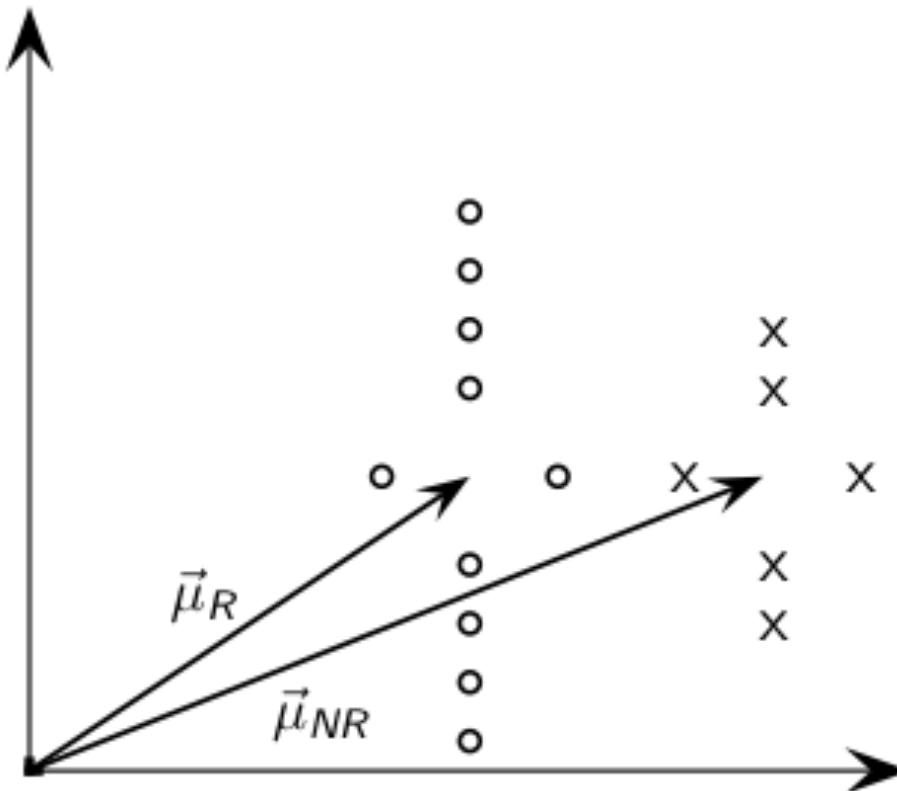
$\vec{\mu}_R$  does not separate relevant / nonrelevant.

# Rocchio' illustrated

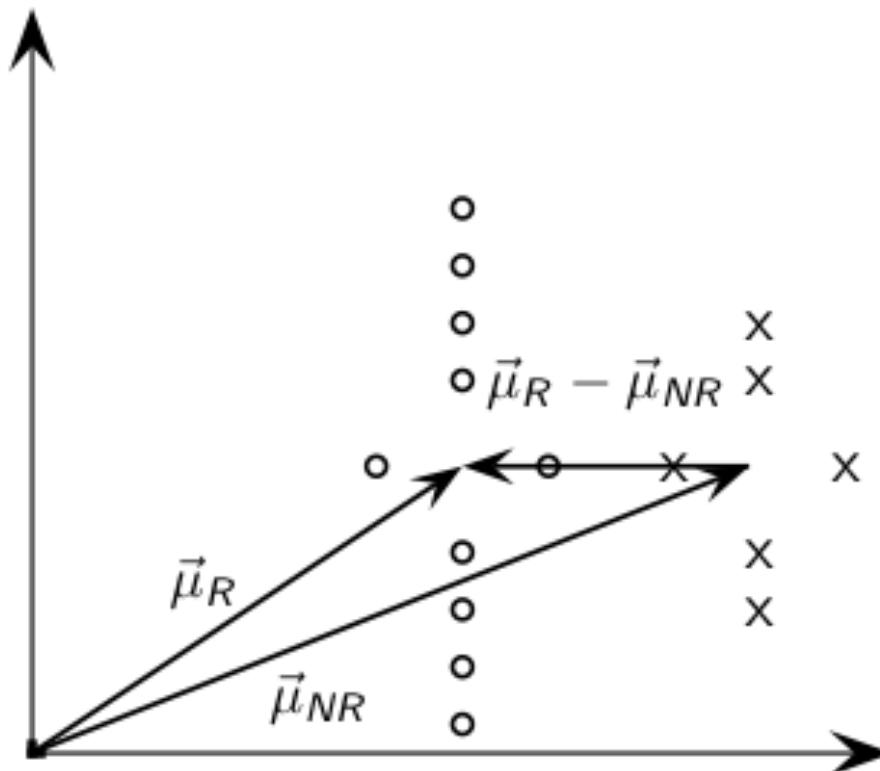


$\vec{\mu}_{NR}$ : centroid of nonrelevant documents.

# Rocchio' illustrated

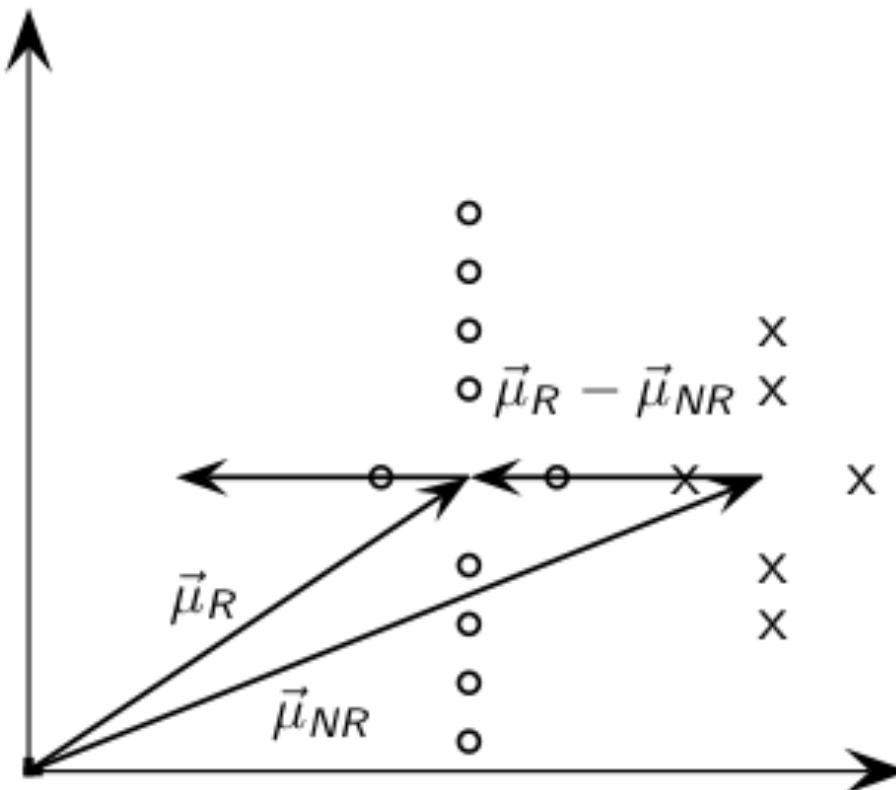


# Rocchio' illustrated



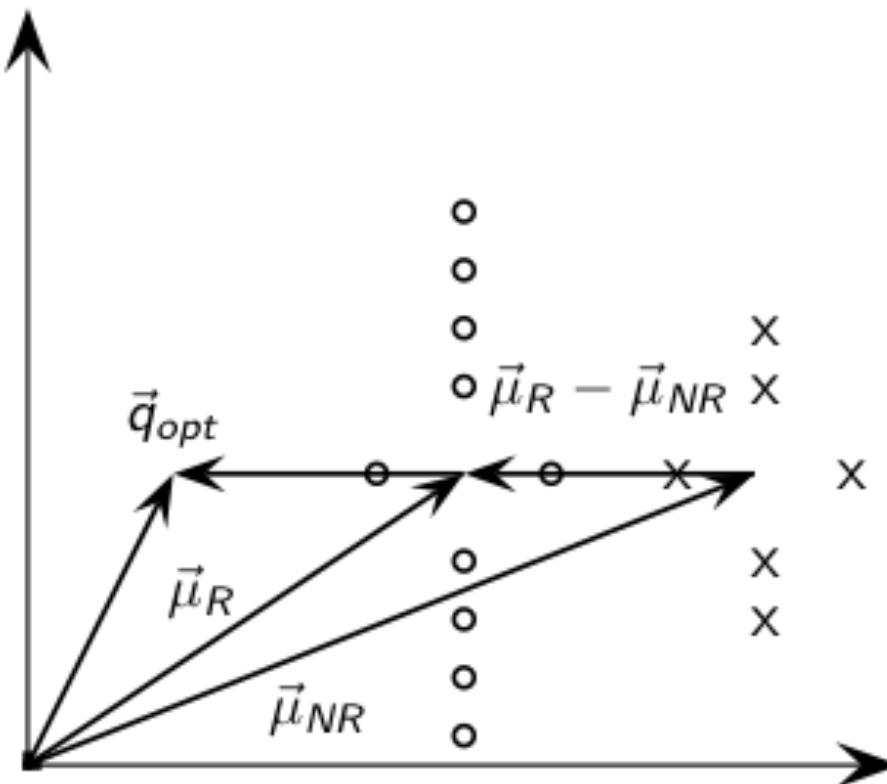
$\vec{\mu}_R - \vec{\mu}_{NR}$ : difference vector

# Rocchio' illustrated



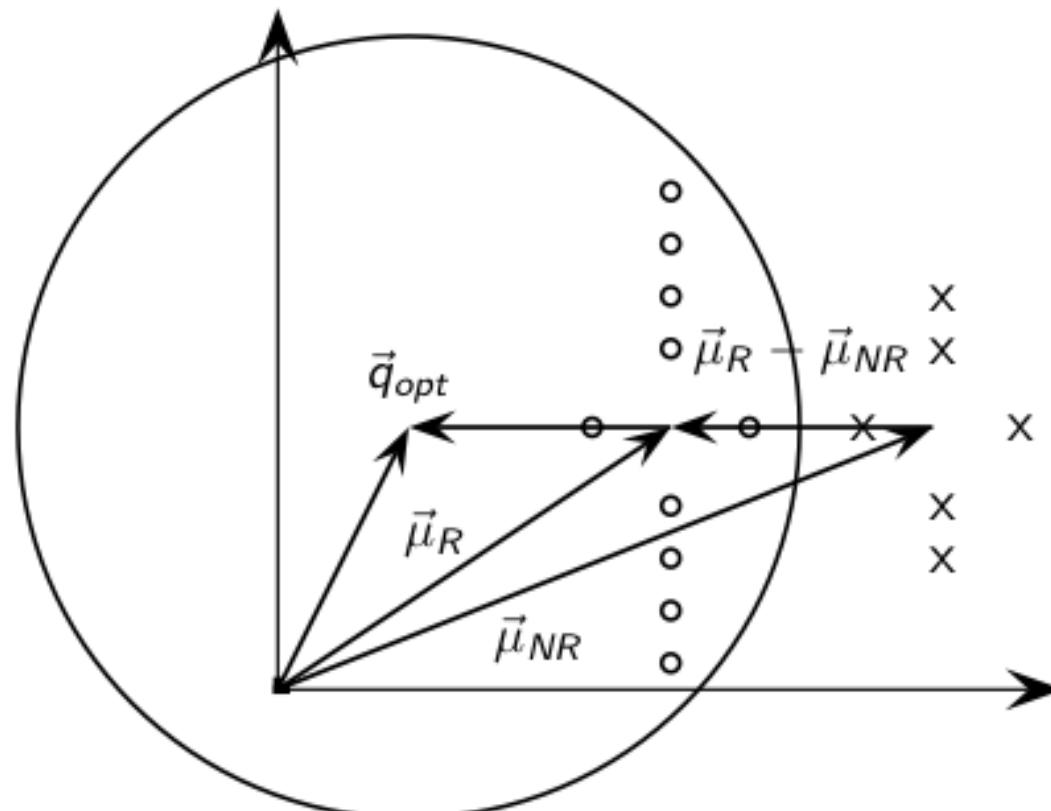
Add difference vector to  $\vec{\mu}_R$  ...

# Rocchio' illustrated



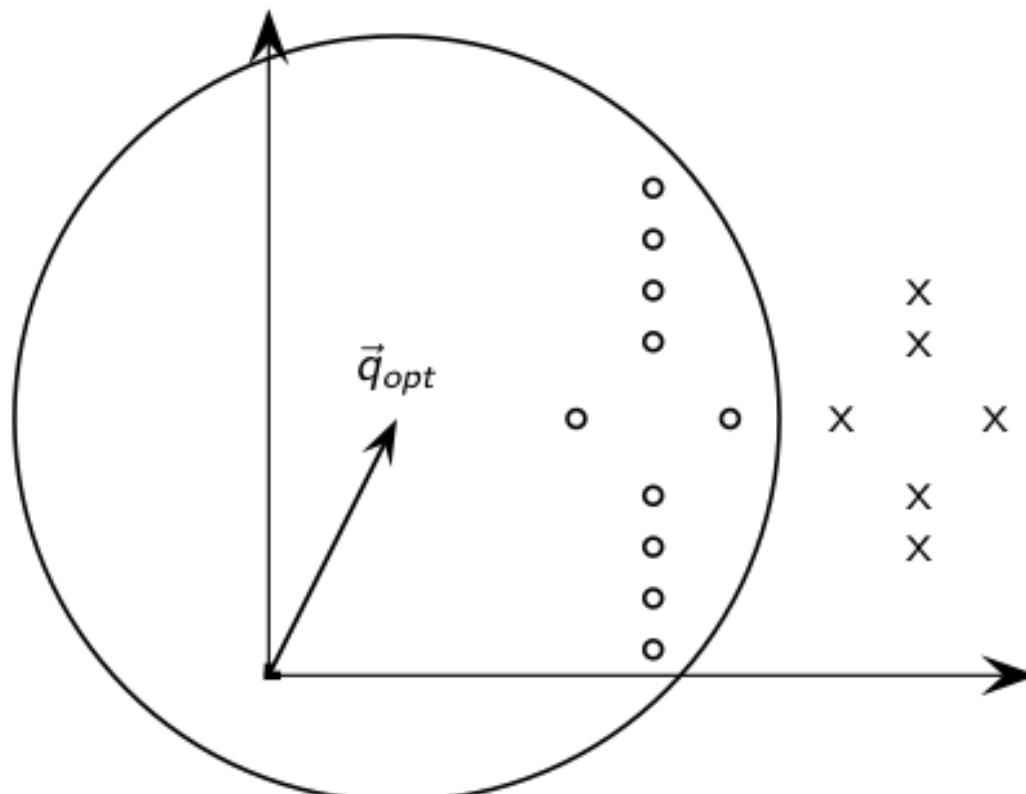
... to get  $\vec{q}_{opt}$

# Rocchio' illustrated



$\vec{q}_{opt}$  separates relevant / nonrelevant perfectly.

# Rocchio' illustrated



$\vec{q}_{opt}$  separates relevant / nonrelevant perfectly.

# Terminology

---

- We use the name Rocchio' for the theoretically better motivated original version of Rocchio.
- The implementation that is actually used in most cases is the SMART implementation – we use the name Rocchio (without prime) for that.

# Rocchio 1971 algorithm (SMART)

Used in practice:

$$\begin{aligned}\vec{q}_m &= \alpha \vec{q}_0 + \beta \mu(D_r) - \gamma \mu(D_{nr}) \\ &= \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j\end{aligned}$$

$\vec{q}_m$ : modified query vector;  $\vec{q}_0$ : original query vector;  $D_r$  and  $D_{nr}$  : sets of known relevant and nonrelevant documents respectively;  $\alpha$ ,  $\beta$ , and  $\gamma$ : weights

- New query moves towards relevant documents and away from nonrelevant documents.
- Tradeoff  $\alpha$  vs.  $\beta/\gamma$ : If we have a lot of judged documents, we want a higher  $\beta/\gamma$ .
- Set negative term weights (if any) to 0, since “negative weight” for a term doesn’t make sense in the vector space model.

# Positive vs. negative relevance feedback

---

- Positive feedback is more valuable than negative feedback.
- For example, set  $\beta = 0.75$ ,  $\gamma = 0.25$  to give higher weight to positive feedback.
- Many systems only allow positive feedback.

# Relevance feedback: Assumptions

---

- When can relevance feedback enhance recall?
- Assumption A1: The user knows the terms in the collection well enough for an initial query.
- Assumption A2: Relevant documents contain similar terms (so I can “hop” from one relevant document to a different one when giving relevance feedback).

# Violation of A1

---

- Assumption A1: The user knows the terms in the collection well enough for an initial query.
- Violation: Mismatch of searcher's vocabulary and collection vocabulary
- Example: cosmonaut / astronaut

# Violation of A2

---

- Assumption A2: Relevant documents contain similar terms
- Example for violation: if the relevant results have several unrelated “prototypes”, e.g.,
  - Subsidies for tobacco farmers vs. anti-smoking campaigns
  - Aid for developing countries vs. high tariffs on imports from developing countries
- Relevance feedback on tobacco docs will not help with finding docs on developing countries.

# Takeaway till now

---

- **Interactive relevance feedback:** improve initial retrieval results by telling the IR system which docs are relevant / nonrelevant
- Best known relevance feedback method: **Rocchio feedback**

# Relevance feedback: Evaluation

---

- Pick one of the evaluation measures, e.g., precision in top 10:  $P@10$
- Compute  $P@10$  for original query  $q_0$
- Compute  $P@10$  for modified relevance feedback query  $q_1$
- In most cases:  $q_1$  is spectacularly better than  $q_0$ !
- **Is this a fair evaluation?**

## Evaluation: Caveat

---

- True evaluation of usefulness **must compare to other methods taking the same amount of time.**
- Alternative to relevance feedback: User revises and resubmits query.
- Users may prefer revision/resubmission to having to judge relevance of documents.
- There is no clear evidence that relevance feedback is the “best use” of the user’s time.

# Relevance feedback: Problems

---

- Relevance feedback is expensive.
  - Relevance feedback creates long modified queries.
  - Long queries are expensive to process.
- Users are reluctant to provide explicit feedback.
- It's often hard to understand why a particular document was retrieved after applying relevance feedback.
  
- The search engine Excite had full relevance feedback at one point, but abandoned it later.

# Pseudo-relevance feedback

---

- Pseudo-relevance feedback automates the “manual” part of true relevance feedback.
- Pseudo-relevance algorithm:
  - Retrieve a ranked list of hits for the user’s query
  - Assume that the top  $k$  documents are relevant
  - Do relevance feedback (e.g., Rocchio)
- Works very well on average
- But can go horribly wrong for some queries.
- Several iterations can cause *query drift*.

# Pseudo-relevance feedback at TREC4

- Cornell SMART system
- Results show number of relevant documents out of top 100 for 50 queries (so total number of documents is 5000):

method	number of relevant documents
Inc.ltc	3210
Inc.ltc-PsRF	3634
Lnu.ltu	3709
Lnu.ltu-PsRF	4350

- Results contrast two length normalization schemes (L vs. I) and pseudo-relevance feedback (PsRF).
- The pseudo-relevance feedback method used added only 20 terms to the query (Rocchio will add many more)
- Demonstrates that pseudo-relevance feedback is effective on average

# Outline

---

- ① Motivation
- ② Relevance feedback: Basics
- ③ Relevance feedback: Details
- ④ Query expansion

# Query expansion

---

- Query expansion is another method for **increasing recall**.
- We use “global query expansion” to refer to “global methods for query reformulation”.
- In global query expansion, the **query is modified based on some global resource, i.e. a resource that is not query-dependent**.
- Main information we use: (near-)synonymy of terms
- A publication or database that collects (near-)synonyms is called a **thesaurus**.
- We will look at two types of thesauri: manually created and automatically created.

# Query expansion: Example

**YAHOO! SEARCH**

Web | Images | Video | Audio | Directory | Local | News | Shopping | More »

palm

Search

Answers My Web Search Services | Advanced Search Preferences

**Search Results** 1 - 10 of about 160,000,000 for palm - 0.07 sec. (About this page)

Also try: [palm springs](#), [palm pilot](#), [palm trees](#), [palm reading](#) [More...](#)

SPONSOR RESULTS

- [Official Palm Store](#)  
[store.palm.com](http://store.palm.com) Free shipping on all handhelds and more at the official Palm store.
- [Palms Hotel - Best Rate Guarantee](#)  
[www.vegas.com](http://www.vegas.com) Book the Palms Hotel Casino with our best rate guarantee at VEGAS.com, the official Vegas travel site.

SPONSOR RESULTS

**Palm Memory**  
Memory Giant is fast and easy.  
Guaranteed compatible memory.  
Great...  
[www.memorygiant.com](http://www.memorygiant.com)

**The Palms, Turks and Caicos Islands**  
Resort/Condo photos, rates,  
availability and reservations....  
[www.worldwidereservationsystems.c](http://www.worldwidereservationsystems.c)

**The Palms Casino Resort, Las Vegas**  
Low price guarantee at the Palms Casino resort in Las Vegas. Book...  
[lasvegas.hotelscorp.com](http://lasvegas.hotelscorp.com)

**Palm Pilots - Palm Downloads**  
Yahoo! Shortcut - [About](#)

1. [Palm, Inc.](#)   
Maker of handheld PDA devices that allow mobile users to manage schedules, contacts, and other personal and business information.  
Category: [B2B](#) > [Personal Digital Assistants \(PDAs\)](#)  
[www.palm.com](http://www.palm.com) - 20k - [Cached](#) - [More from this site](#) - [Save](#)

# Types of user feedback

---

- User gives feedback on documents.
  - More common in relevance feedback
  
- User gives feedback on words or phrases.
  - More common in query expansion

# Types of query expansion

---

- Manually constructed thesaurus (maintained by editors, e.g., Unified Medical Language System)
- Automatically derived thesaurus (e.g., based on co-occurrence statistics of terms)
- Query-equivalence based on query log mining (common on the web as in the “palm” example few slides back)

# Thesaurus-based query expansion

- For each term  $t$  in the query, expand the query with words the thesaurus lists as semantically related with  $t$ .
- Example: HOSPITAL → MEDICAL
- Generally increases recall
- May significantly decrease precision, particularly with ambiguous terms: INTEREST RATE → INTEREST RATE FASCINATE
  
- Widely used in specialized search for science & engineering
- It's **very expensive to create a manual thesaurus and to maintain it over time.**
- A manual thesaurus has an effect roughly equivalent to annotation with a controlled vocabulary

# Automatic thesaurus generation

- Attempt to generate a thesaurus automatically by analyzing the distribution of words in documents
- Fundamental notion: similarity between two words
- Definition 1: Two words are **similar if they co-occur with similar words.**
  - “car” ≈ “motorcycle” because both occur with “road”, “gas” and “license”, so they must be similar.
- Definition 2: Two words are **similar if they occur in a given grammatical relation with the same words.**
  - You can harvest, peel, eat, prepare, etc. “apples” and “pears”, so “apples” and “pears” must be similar.
- Co-occurrence is more robust, grammatical relations are more accurate.

# Co-occurrence-based thesaurus construction

$$PMI(w_1, w_2) = \log_2 \frac{P_{corpus}(w_1, w_2)}{P_{corpus}(w_1)P_{corpus}(w_2)}$$

$$P_{corpus}(w_1, w_2) = \frac{freq(w_1, w_2)}{N} \quad P_{corpus}(w) = \frac{freq(w)}{N}$$

Statistically measure whether two words co-occur frequently (relative to their global frequencies)

# Co-occurrence-based thesaurus: Examples

<b>petroleum</b>	oil:0.032 gas:0.029 crude:0.029 barrels:0.028 exploration:0.027 barrel:0.026 opec:0.026 refining:0.026 gasoline:0.026 fuel:0.025 natural:0.025 exporting:0.025
<b>drug</b>	trafficking:0.029 cocaine:0.028 narcotics:0.027 fda:0.026 police:0.026 abuse:0.026 marijuana:0.025 crime:0.025 colombian:0.025 arrested:0.025 addicts:0.024
<b>insurance</b>	insurers:0.028 premiums:0.028 lloyds:0.026 reinsurance:0.026 underwriting:0.025 pension:0.025 mortgage:0.025 credit:0.025 investors:0.024 claims:0.024 benefits:0.024
<b>forest</b>	timber:0.028 trees:0.027 land:0.027 forestry:0.026 environmental:0.026 species:0.026 wildlife:0.026 habitat:0.025 tree:0.025 mountain:0.025 river:0.025 lake:0.025
<b>robotics</b>	robots:0.032 automation:0.029 technology:0.028 engineering:0.026 systems:0.026 sensors:0.025 welding:0.025 computer:0.025 manufacturing:0.025 automated:0.025

$$PMI(w_1, w_2) = \log_2 \frac{P_{corpus}(w_1, w_2)}{P_{corpus}(w_1)P_{corpus}(w_2)}$$

$$P_{corpus}(w_1, w_2) = \frac{freq(w_1, w_2)}{N} \quad P_{corpus}(w) = \frac{freq(w)}{N}$$

# Query Expansion: Examples

## *TREC Topic 104: catastrophic health insurance*

**Query Representation:** surtax:1.0 hcfa:0.97 medicare:0.93 hmos:0.83  
medicaid:0.8 hmo:0.78 beneficiaries:0.75 ambulatory:0.72 premiums:0.72  
hospitalization:0.71 hhs:0.7 reimbursable:0.7 deductible:0.69

- Broad expansion terms: **medicare, beneficiaries, premiums** ...
- Specific domain terms: **HCFA** (Health Care Financing Administration), **HMO** (Health Maintenance Organization), **HHS** (Health and Human Services)

## *TREC Topic 355: ocean remote sensing*

**Query Representation:** radiometer:1.0 landsat:0.97 ionosphere:0.94  
cnes:0.84 altimeter:0.83 nasda:0.81 meterology:0.81 cartography:0.78  
geostationary:0.78 doppler:0.78 oceanographic:0.76

- Broad expansion terms: **radiometer, landsat, ionosphere** ...
- Specific domain terms: **CNES** (Centre National d'Études Spatiales) and **NASDA** (National Space Development Agency of Japan)

# Query expansion at search engines

---

- Main source of query expansion at search engines: **query logs**
  
- Example 1: After issuing the query [herbs], users frequently search for [herbal remedies].
  - → “herbal remedies” is potential expansion of “herb”.
  
- Example 2: Users searching for [flower pix] frequently click on the URL **photobucket.com/flower**. Users searching for [flower clipart] frequently click on the **same URL**.
  - → “flower clipart” and “flower pix” are potential expansions of each other.