Performance Metrics

Accuracy

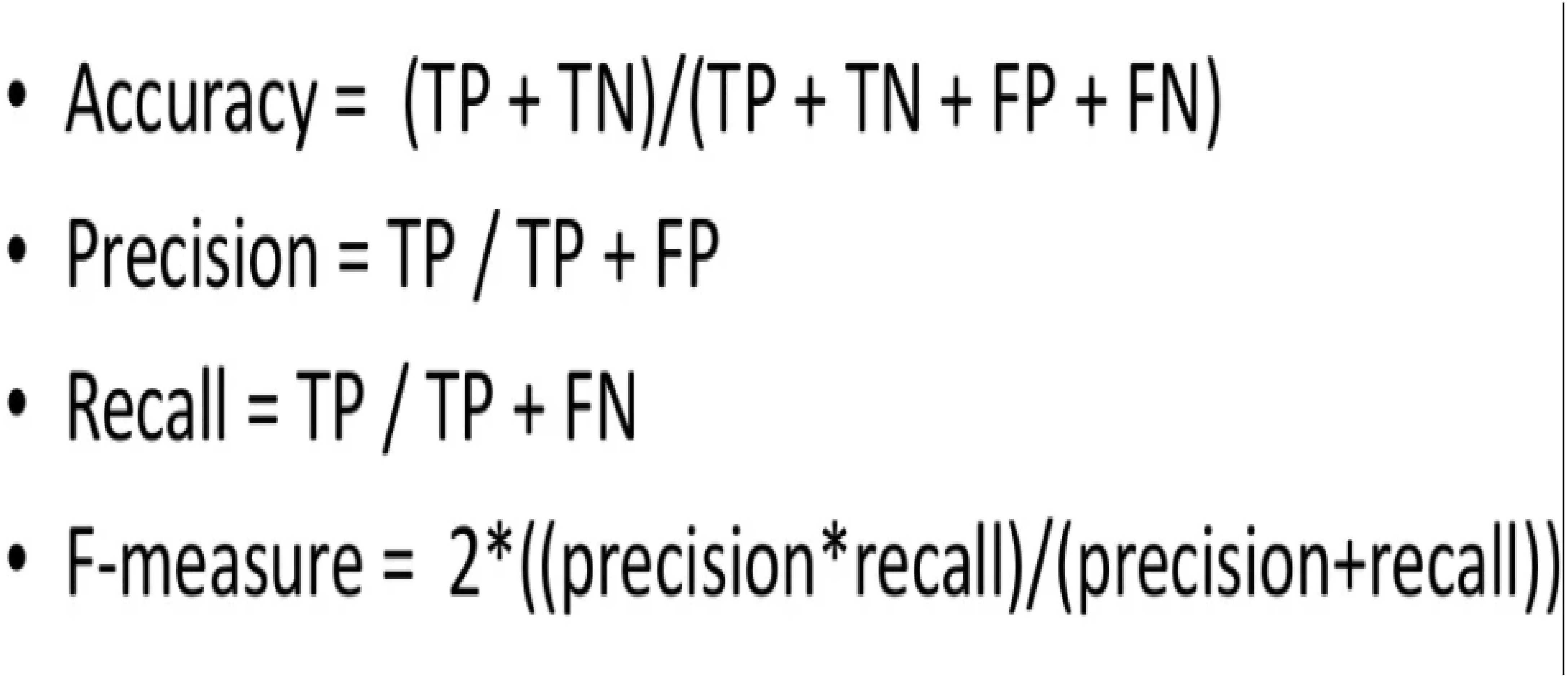
- Measures the percentage of correct results that a classifier has achieved.

➢ True and False Positives and Negatives

- True positives are relavant items that we correctly identified as relevant
- True Negative are irrelevant items that we correctly identified as irrelevant
- False Positive are irrelevant items that we incorrectly identified as relevant
- False Negative are relevant items that that we incorrectly identified as irrelevant

Given these four numbers, we can define the following metrics:

- **Precision**, which indicates how many of the items that we identified were relevant, is *TP/(TP+FP)*.
- **Recall**, which indicates how many of the relevant items that we identified, is *TP/(TP+FN)*.
- The **F-Measure** (or **F-Score**), which combines the precision and recall to give a single score, is defined to be the harmonic mean of the precision and recall: (2 × *Precision* × *Recall*) / (*Precision* + *Recall*).

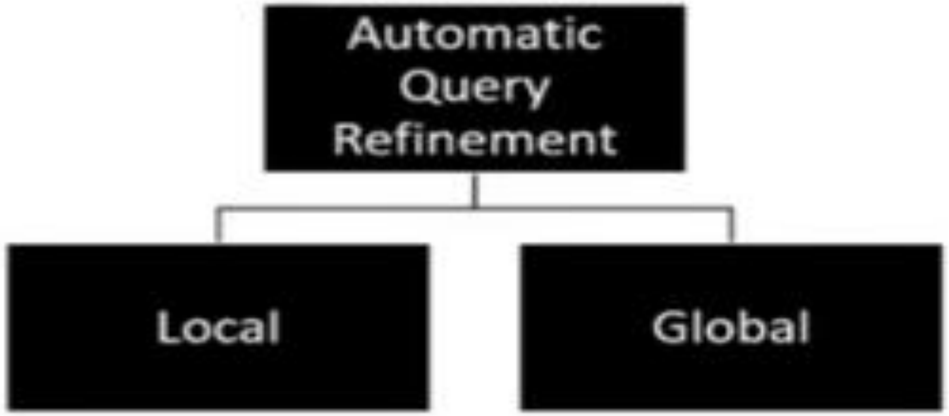


- Accuracy = (TP + TN)/(TP + TN + FP + FN)
- Precision = TP / TP + FP
- Recall = TP / TP + FN
- F-measure = 2*((precision*recall)/(precision+recall))

The Problem of Synonymy

- What result do you expect for a query, “plane”?
- What if plane appears in this query, “plane from Delhi to Goa”?
- So many synonyms which will work for web search…
  - Flight
  - Aircraft
  - Airplane
  - Aeroplane
  - By Air
  - Fly
  - Flgt

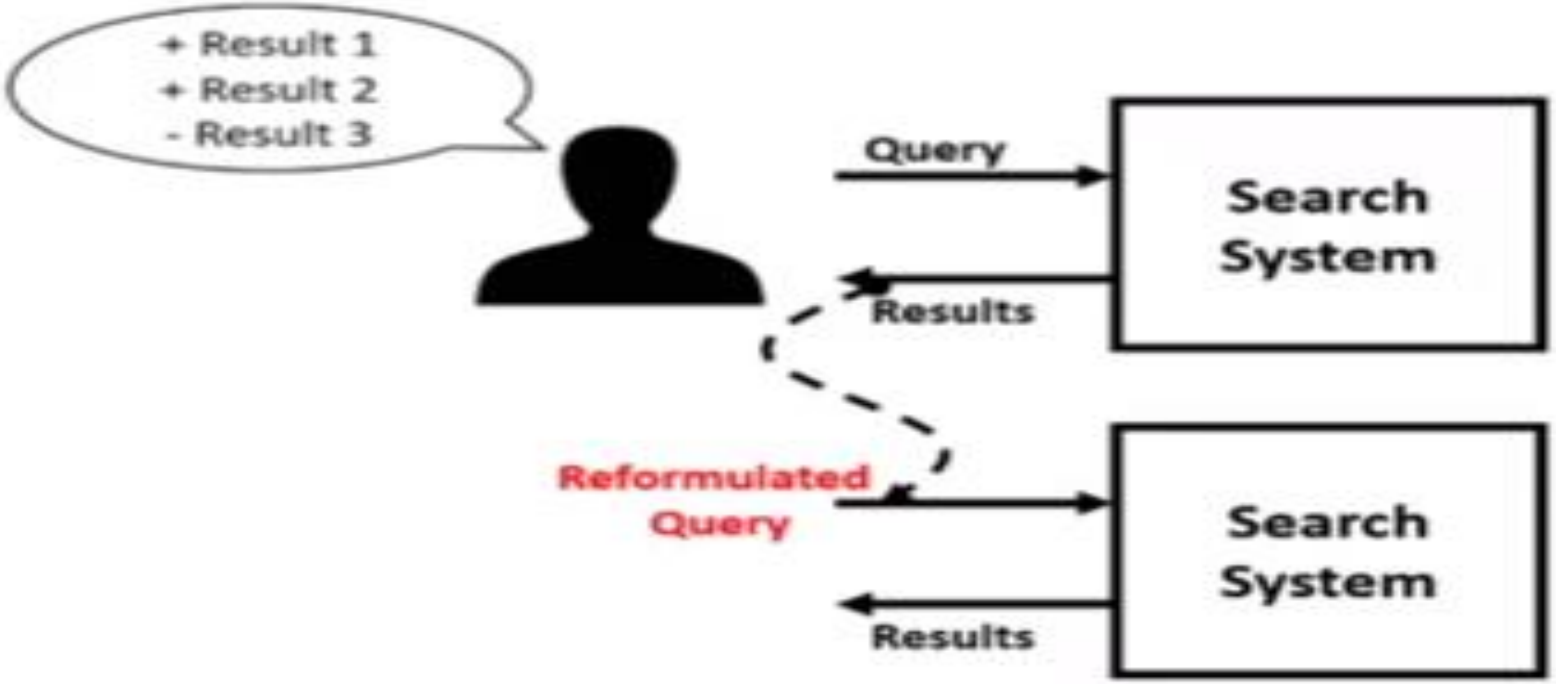
How to ensure good results?



Use the **query** or the **results** for reformulating the query
We will study:
Relevance Feedback
Pseudorelevance
Indirect Relevance Feedback

Do not use the **query** or the **results** for reformulating the query.
Eg:
Use *Thesaurus.*
Do Spelling Correction.

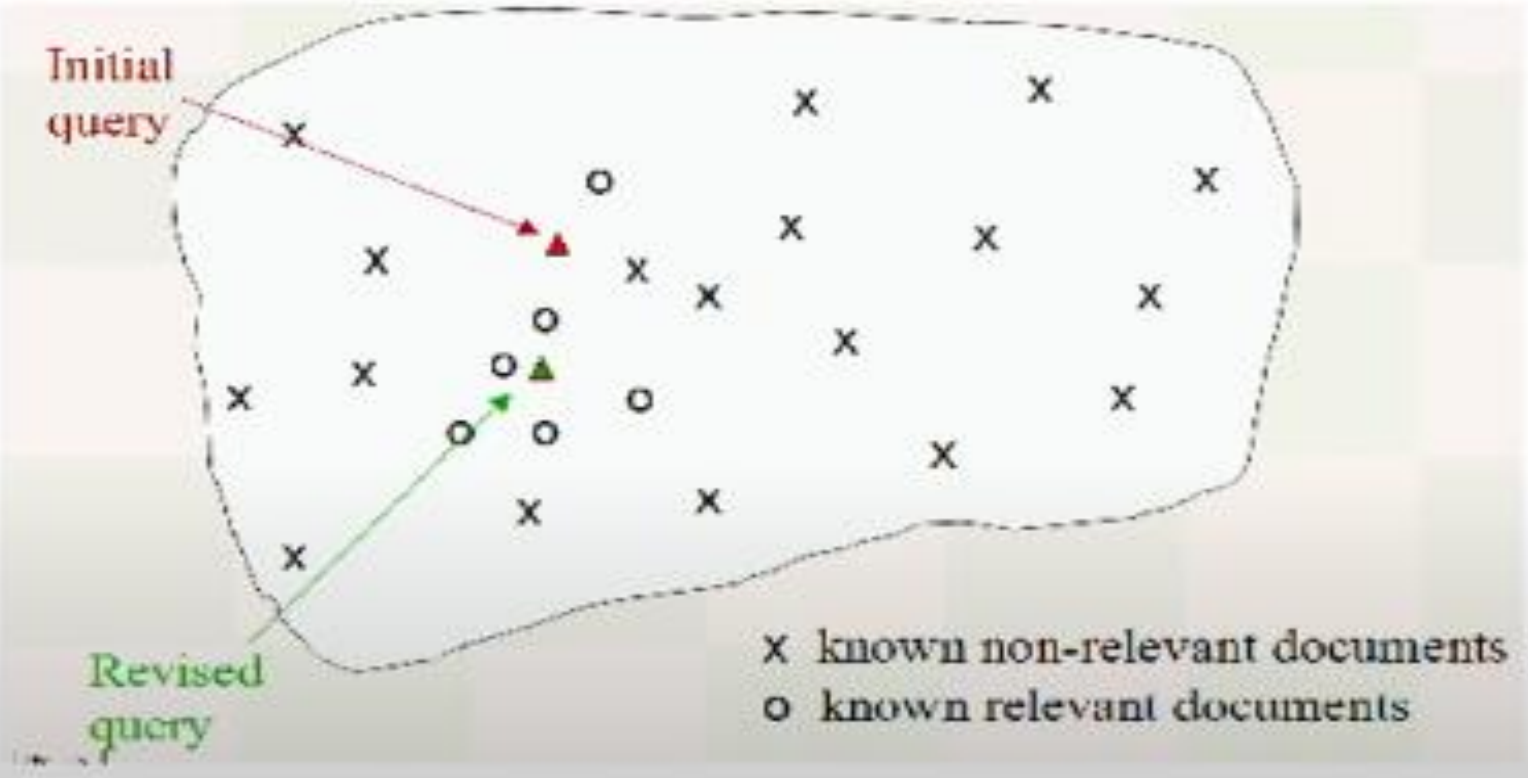
Relevance Feedback



Interesting Characteristics

- Indexed content is unknown to the user.
- “Information Need” changes after looking at the results.
    - User visits youtube to listen to a specific set of songs.
    - After the first song, he changes his mind and listens to something else!

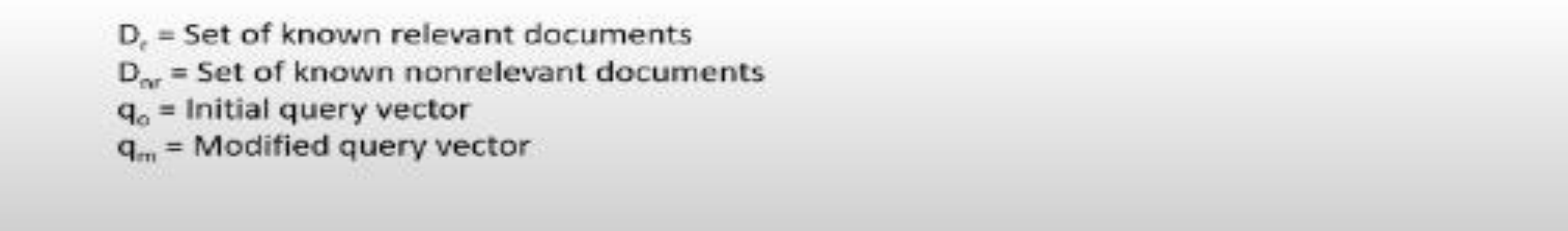
Rocchio Algorithm for Relevance Feedback



Moving the Centroid!

Modify the query (and therefore, the query vector from q0 to qm):

$$\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$$



D_r = Set of known relevant documents

D_{nr} = Set of known nonrelevant documents

q_o = Initial query vector

q_m = Modified query vector

Rocchio relevance feedback - Example

- Given:
  - Initial query = “cheap CDs cheap DVDs extremely cheap CDs”.
  - $d_{1\,=}$ “CDs cheap software cheap CDs” is judged as relevant.
  - $d_{2\,=}$ “cheap thrills DVDs” is judged as nonrelevant
- What would the revised query vector be after relevance feedback?

Representing Initial Query in Vector Space

Initial query = "cheap CDs cheap DVDs extremely cheap CDs".

	cheap	CDs	DVDs	extremely	software	thrills
q_0	3	2	1	1	0	0

Rocchio relevance feedback - Example

Quiz: Can you complete the following table?

q_0 = "cheap CDs cheap DVDs extremely cheap CDs".

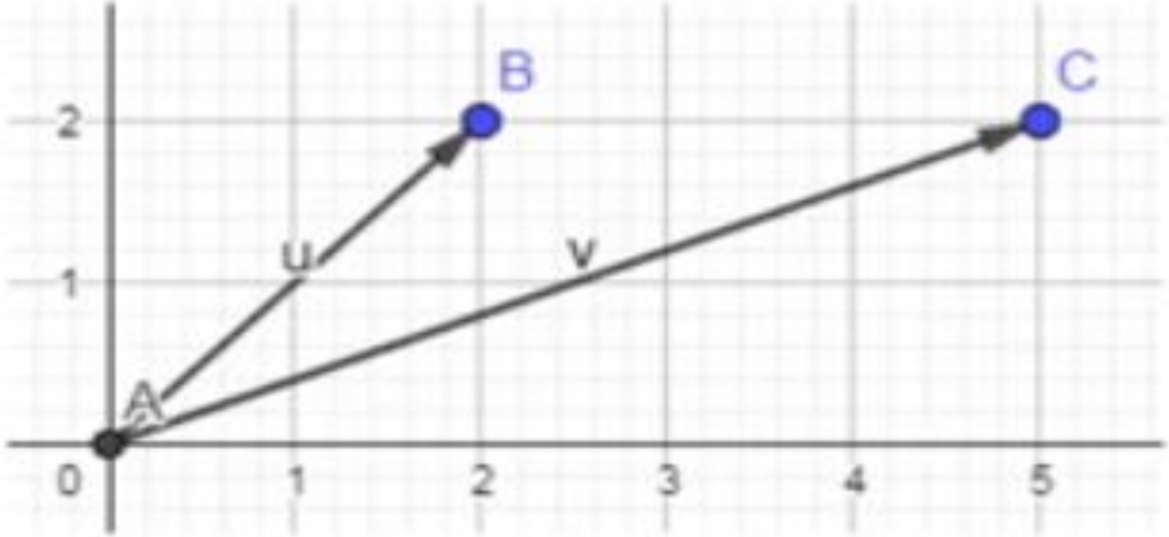
d_1 = "CDs cheap software cheap CDs".

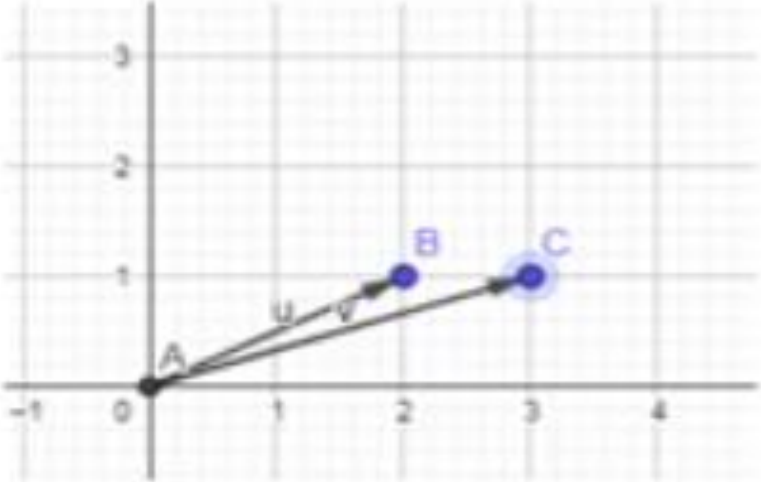
d_2 = "cheap thrills DVDs".

	cheap	CDs	DVDs	extremely	software	thrills
q_0	3	2	1	1	0	0
d_1	2	2	0	0	1	0
d_2	1	0	1	0	0	1

Moving Vectors

- Move (2,2) to (5,2) by adding 3 to x.





Rocchio relevance feedback - Example

Quiz: How to calculate the modified query vector, q_m?

d_1 is judged as **relevant**. d_2 is judged as **non-relevant**.

Assume $\alpha = 1$, $\beta = 0.75$, $\gamma = 0.25$.

	cheap	CDs	DVDs	extremely	software	thrills
q_0	3	2	1	1	0	0
d_1	2	2	0	0	1	0
d_2	1	0	1	0	0	1

q_m						

$$\vec{q}_m = \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j$$

Rocchio relevance feedback - Example

Quiz: How to calculate the modified query vector, q_m?

d_1 is judged as relevant. d_2 is judged as nonrelevant.
Assume $\alpha = 1$, $\beta = 0.75$, $\gamma = 0.25$.

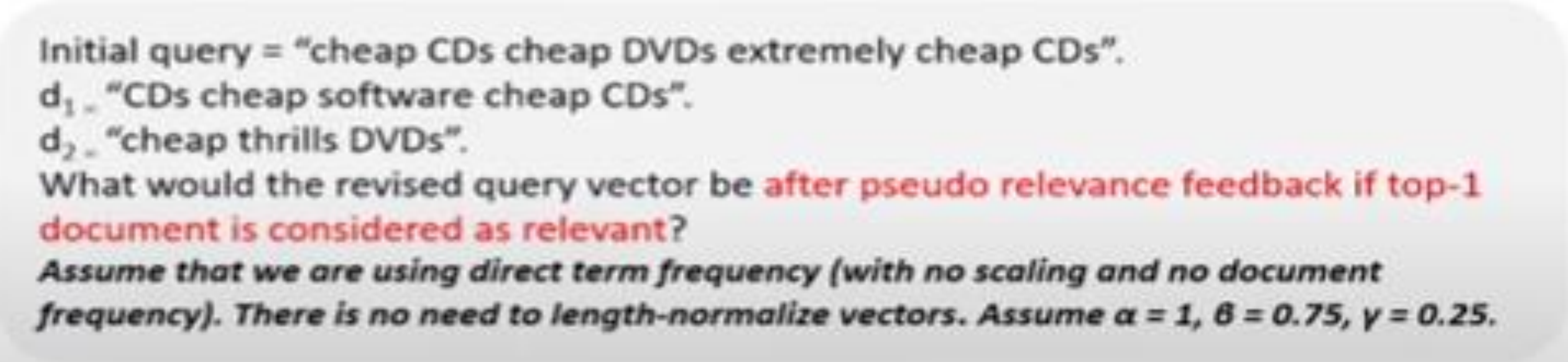
	cheap	CDs	DVDs	extremely	software	thrills
q_0	3	2	1	1	0	0
d_1	2	2	0	0	1	0
d_2	1	0	1	0	0	1
	$q_m = q_0 + 0.75*d_1 - 0.25*d_2$					
q_m	4.25	3.5	0.75	1	0.75	0

Negative weight does not make sense. So, leave them as zero.

$$\vec{q}_m = \alpha \vec{q}_0 + \beta \frac{1}{|D_r|} \sum_{\vec{d}_j \in D_r} \vec{d}_j - \gamma \frac{1}{|D_{nr}|} \sum_{\vec{d}_j \in D_{nr}} \vec{d}_j$$

Pseudo (Blind) Relevance Feedback

- No User Judgment.
- Assume that the top-k ranked documents are relevant.



Initial query = “cheap CDs cheap DVDs extremely cheap CDs”.
d_1 = “CDs cheap software cheap CDs”.
d_2 = “cheap thrills DVDs”.
What would the revised query vector be after pseudo relevance feedback if top-1 document is considered as relevant?
Assume that we are using direct term frequency (with no scaling and no document frequency). There is no need to length-normalize vectors. Assume α = 1, β = 0.75, γ = 0.25.

- May lead to query drift.

Indirect (Implicit) Relevance Feedback

- No asking for judgments from users.
- No automatic feedback such as assuming top-k documents as relevant.

Clickstream Mining

Global methods for query reformulation

➢Vocabulary tools for query reformulation

- Helps user to see how their searches are or are not working
- This includes information about words that were omitted from query because they were on stop list,words were stemmed to….
- The IR system might also suggest search terms by means of a thesaurus or a controlled vocabulary.

Global methods for query reformulation

➢Query Expansion

- In IR users give additional input on query words or phrases.
- Some search engines suggest related querries in response to a query
- The most common form of query expansion is global analysis,using some form of thesaurus.
- For each term t in a query,the query can be automatically expanded with synonyms and related words of t from thesaurus.

Methods for building a thesaurus for query expansion

- **Use of controlled vocabulary** that is maintained by human editors
- Example: Unified Medical Language System used with Medline for querying the biomedical research literature
- A manual thesaurus: Manual editors built up sets of synonymous names for concepts
- An automatically derived thesaurus: here word cooccurence statistics over a collection of documents in domain are used toautometically induce a thesaurus
- Query reformulation based on query log mining:we exploit the manual query formulatios of othr users to make suggestions to a new user.This require a huge query volume.

- Automatic thesaurus generation:
- There are two approaches:
- One is to exploit word cooccurence
- Other approach is to use a shallow grammatical analysis of the text andd to exploit grammatical relations or grammatical dependencies.
- shallow grammatical analysis :**analysis of a sentence** which first identifies constituent parts of sentences (nouns, verbs, adjectives, etc.) and then links them to higher order units that have discrete grammatical meanings (noun groups or phrases, verb groups, etc.).

Query Languages from yates