Monsoon 2020

9 - Vector Space Models

Information

Retrieval

by

Dr. Rajendra Prasath

Indian Institute of Information Technology, Sri City, Chittoor

Sri City – 517 646, Andhra Pradesh, India

Lecture – 09

Topics Covered So Far

- Bi-Word Index
- Wild Card Queries
- Permuterm Index
- K-gram Index (k = 2 □ Bigram Index)
- Spell Correction
- Term Weighting

Now: Vector Space Model

Recap: Overview

- Why Ranked Retrieval?
- Term Frequency
- Term Weighting
- TF-IDF Weighting
- The Vector Space Model

Recap: Ranked Retrieval

- Our Queries have all been Boolean
  - Documents either match or don’t
- Good for expert users with precise understanding of their needs and of the collection.
- Also good for applications: Applications can easily consume 1000s of results.
- Not good for the majority of users
- Most users don’t want to wade through 1000s of results.
- This is particularly true of web search.

4

Scoring as the basis of ranked retrieval

- Rank documents such that more relevant documents higher than less relevant document

- How do we do follow?
  - Accomplish a ranking of the documents in the collection with respect to a query?

- Assign a score to each query-document pair, say in [0, 1]
- This score measures how well document and query "**match**"

5

Query – Docs matching scores

- How do we compute the score of
  a query - document pair?

- Let’s start with a one-term query.
- If the query term does not occur in the document: score should be 0.
- The more frequent the query term in the document, the higher the score
- We will look at a number of alternatives for doing this.

6

Term Frequency (tf)

- The term frequency $tf_{t,d}$ of term t in document d:
  - The number of times that t occurs in d

- Use tf to compute query-doc. match scores
- Raw term frequency is not what we want

- A document with tf = 10 occurrences of the term is more relevant than a document
  with tf = 1 occurrence of the term
- But not 10 times more relevant
- Relevance does not increase proportionally with term frequency

7

Exercise

- Compute Jaccard matching score & TF matching score for the following query-document pairs

  q: [information on cars]

  d: “all you’ve ever wanted to know about cars”
- q: [information on cars]

  d: “information on trucks, information on planes, information on trains”
- q: [red cars and red trucks]

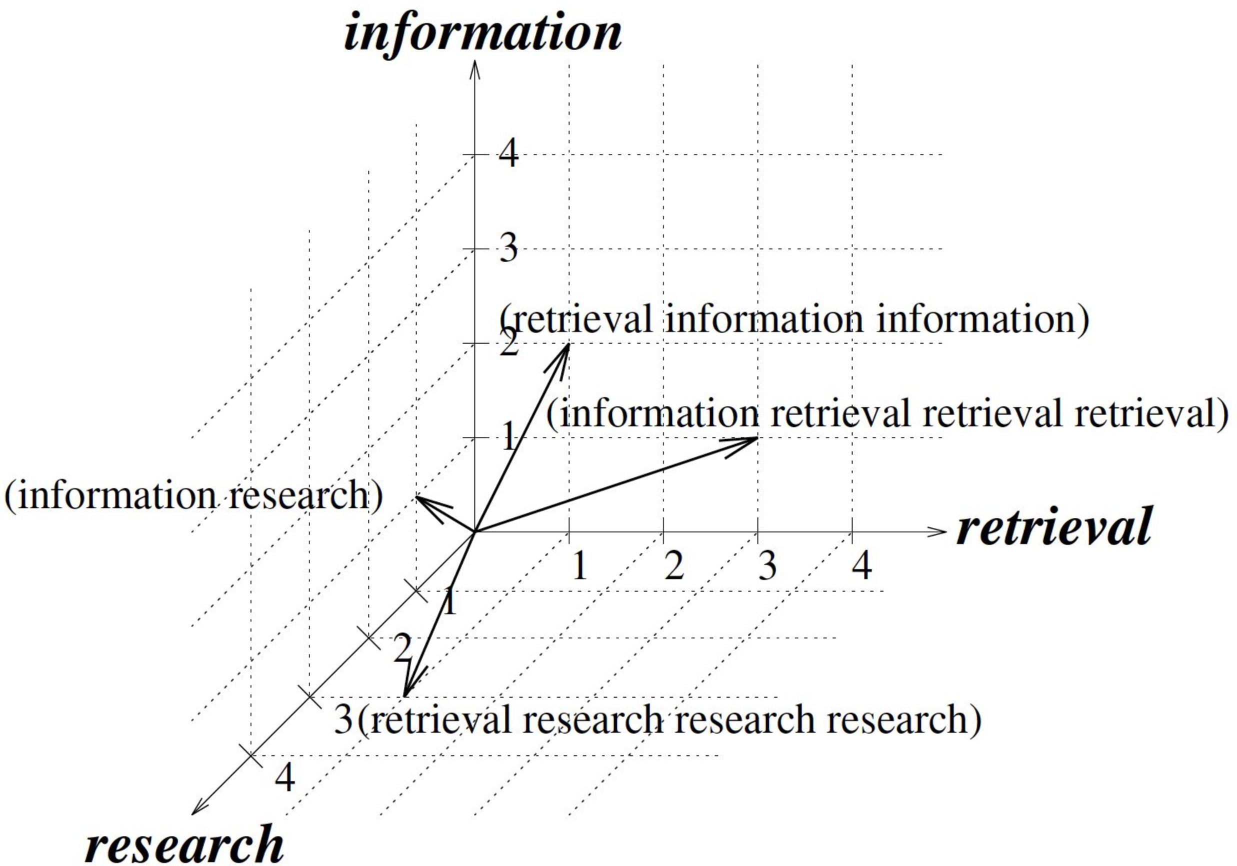
  d: “cops stop red cars more often”

8

Vector Space Model

Consider Three Words Model

"information retrieval research"



9

Term Frequency Factor

- What is Term Frequency Factor?
  - The function of the term frequency used to compute a term’s importance
  - Some commonly used factors are:
    - Raw TF factor
    - Logarithmic TF factor
    - Binary TF factor
    - Augmented TF factor
    - Okapi’s TF factor

10

Measure of Closeness of Vectors

- **Measure the closeness between two vectors**
- Two texts are semantically related if they share some vocabulary
  - More Vocabulary they share, the stronger is the relationship
- This implies that the measure of closeness increases with the number of words matches between two texts
- If matching terms are important then vectors should be considered closer to each other

11

Modern Vector Space Models

- The length of the sub-vector in dimension - i is used to represent the importance or the weigh of word – i in a text
- Words that are absent in a text get a weight – 0 (zero)
- Apply **Vector Inner Product** measure between two vectors:
- This vector inner product increases:
  - # words match between two texts
  - Importance of the matching terms

12

Finding closeness between texts

- Given two texts in T dimensional vector space:

$$\vec{P} = (p_1, p_2, \ldots, p_T) \text{ and } \vec{Q} = (q_1, q_2, \ldots, q_T)$$

- The inner product between these two vectors:

$$\vec{P} \cdot \vec{Q} = \sum_{i=1}^{T} \sum_{j=1}^{T} p_i \times \vec{u_i} \cdot q_j \times \vec{u_j}$$

- Vectors u_i and u_j are unit vectors in dimensions i *and* j (Here $u_i \cdot u_j = 0$, if $i \neq j$ - orthogonal)
- Vector Similarity: Closeness between two texts

$$similarity(\vec{P}, \vec{Q}) = \sum_{i=1}^{T} p_i \times q_i$$

13

Recap: Exercise – Ex08

- Consider a collection of n documents
- Let n be sufficiently large (at least 100 docs)
- Find two lists:
  - The most frequent words and
  - The least frequent words
- Form k (=10) queries each with exactly 3-words taken from above lists (at least one from each)
- Compute Similarity between each query and and documents

14

Inverse Document Frequency

- Using the TF factors to estimate the term importance does not suffice
- Why?
  - Consider common words that occur with very high frequency across numerous articles.
  - Such words are not very informative
  - A match between a query and a document on words like “put” or “the’ does not mean much in terms of the semantic relationship between the query and the document.

15

Summary

In this class, we focused on:

(a) Words / Terms / Lexical Units

(b) Preparing Term – Document matrix

(c) Boolean Retrieval

(d) Inverted Index Construction

  i. Computational Cost

  ii. Managing Bigger Collections

  iii. How much storage is required?

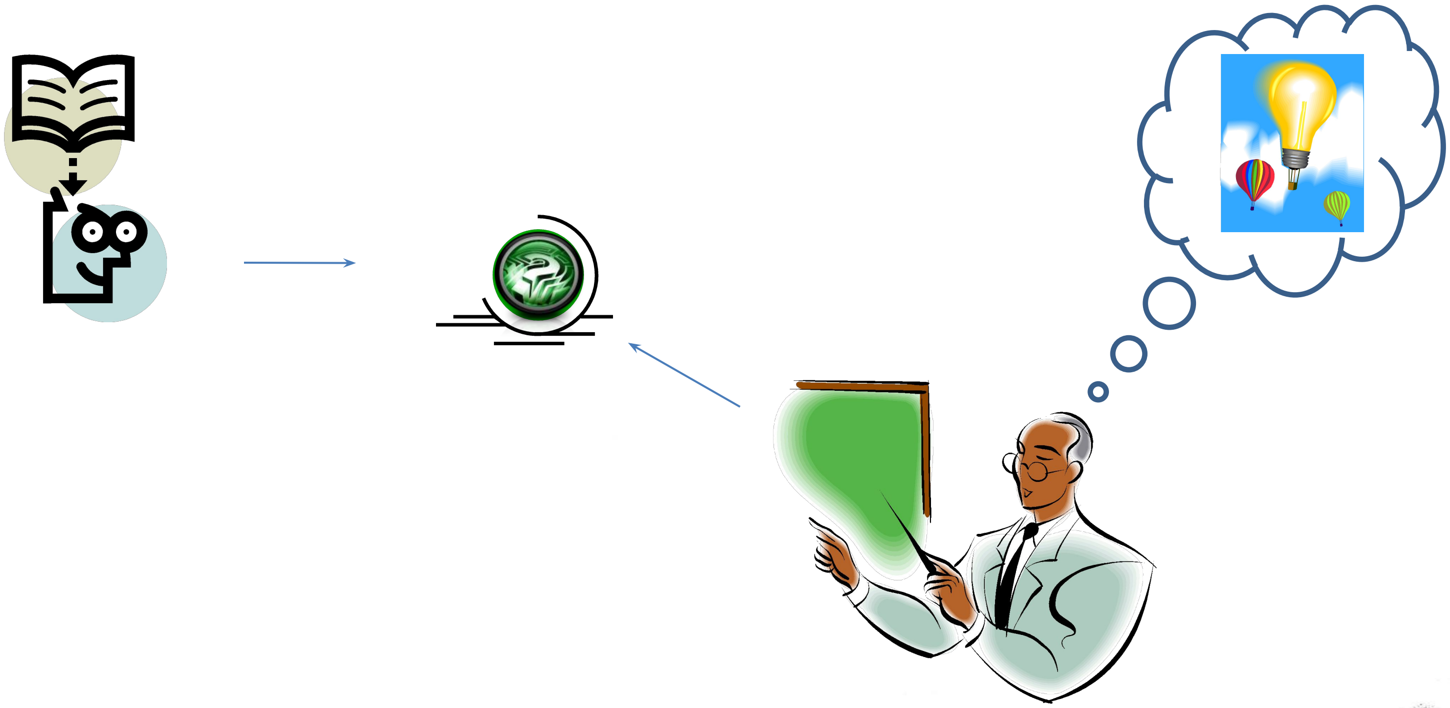
  iv. Boolean Queries: Exact match

Acknowledgements

Thanks to ALL RESEARCHERS:

1. Introduction to Information Retrieval Manning, Raghavan and Schutze, Cambridge University Press, 2008.
2. Search Engines Information Retrieval in Practice W. Bruce Croft, D. Metzler, T. Strohman, Pearson, 2009.
3. Information Retrieval Implementing and Evaluating Search Engines Stefan Büttcher, Charles L. A. Clarke and Gordon V. Cormack, MIT Press, 2010.
4. Modern Information Retrieval Baeza-Yates and Ribeiro-Neto, Addison Wesley, 1999.
5. Many Authors who contributed to SIGIR / WWW / KDD / ECIR / CIKM / WSDM and other top tier conferences
6. Prof. Mandar Mitra, Indian Statistical Institute, Kolkatata (https://www.isical.ac.in/~mandar/)

Thanks …



… Questions ???