IR Systems

(Note: Many slides from this slide set were adapted from an IR course taught by Prof. Ray Mooney at UT Austin)

CS4731 – Information Retrieval and Extraction

Vasudeva Varma
www.iiit.ac.in/~vasu

What is Information Retrieval?

- Concerned firstly with retrieving *relevant* documents to a query.
- Concerned secondly with retrieving from *large* sets of documents *efficiently*.
- What does that generally mean?
  - Search
  - Filtering
  - Organization
  - Multiple languages
  - Multiple media

Approaches to IR

- Two types of retrieval
  - By metadata (subject headings, keywords, etc.)
  - By content
- Metadata as manually assigned information
  - Human agreement is not good
  - Expensive for most data
- Metadata assigned automatically
  - Quality is reasonable, but not high for many applications
- Metadata in general
  - Requires *a priori* prediction of headings, keywords, …
- Most successful IR approaches are content-based

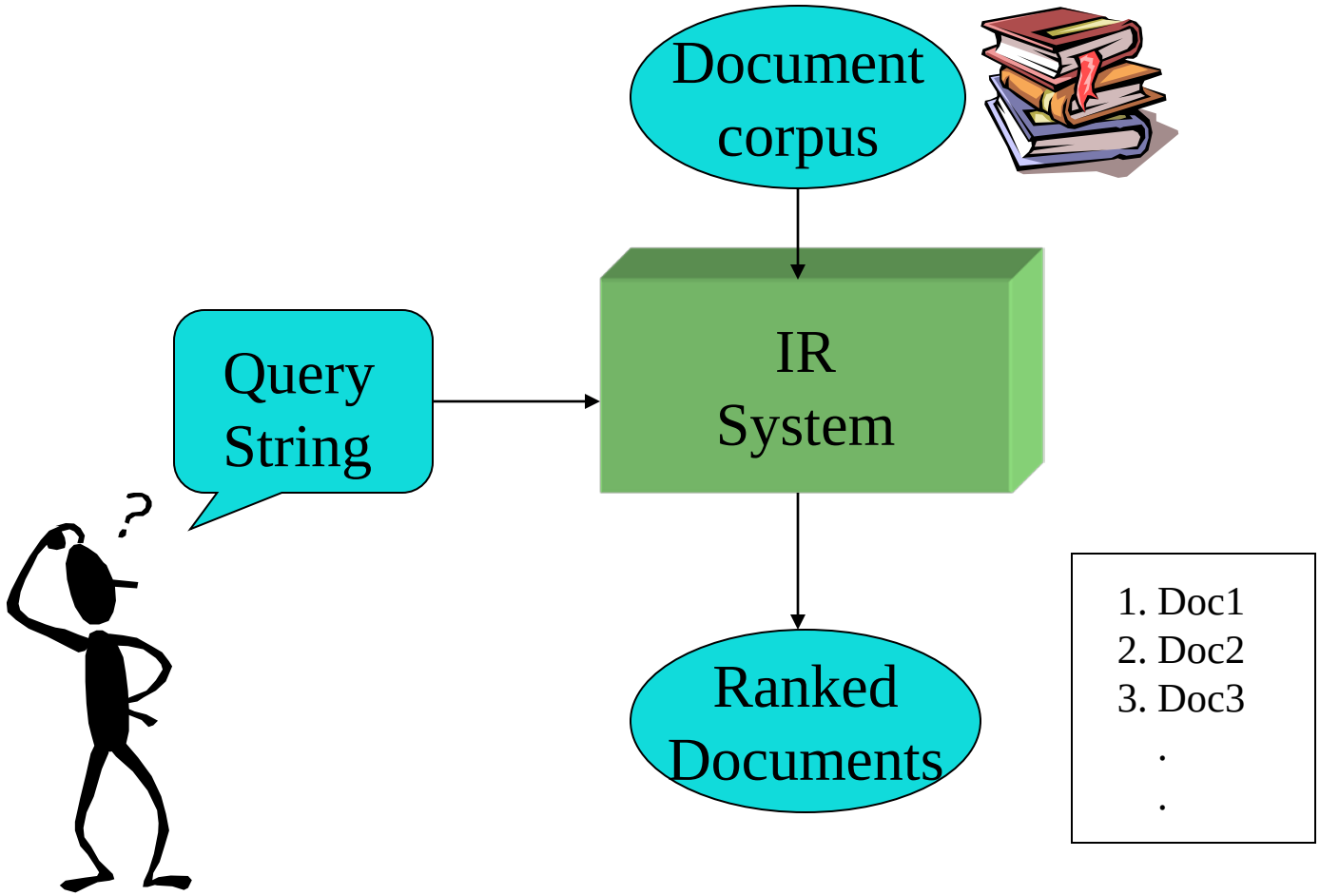
Basic approach to IR

- Successful content-based approaches are statistical
  - Rather than actual "understanding" of text
  - Text understanding effectiveness is very poor
    - Exception: works better in some restricted domains
- IR statistics used in different ways
  - Past/concurrent queries and relevance judgments
    - Collaborative systems
  - Document and query similarities

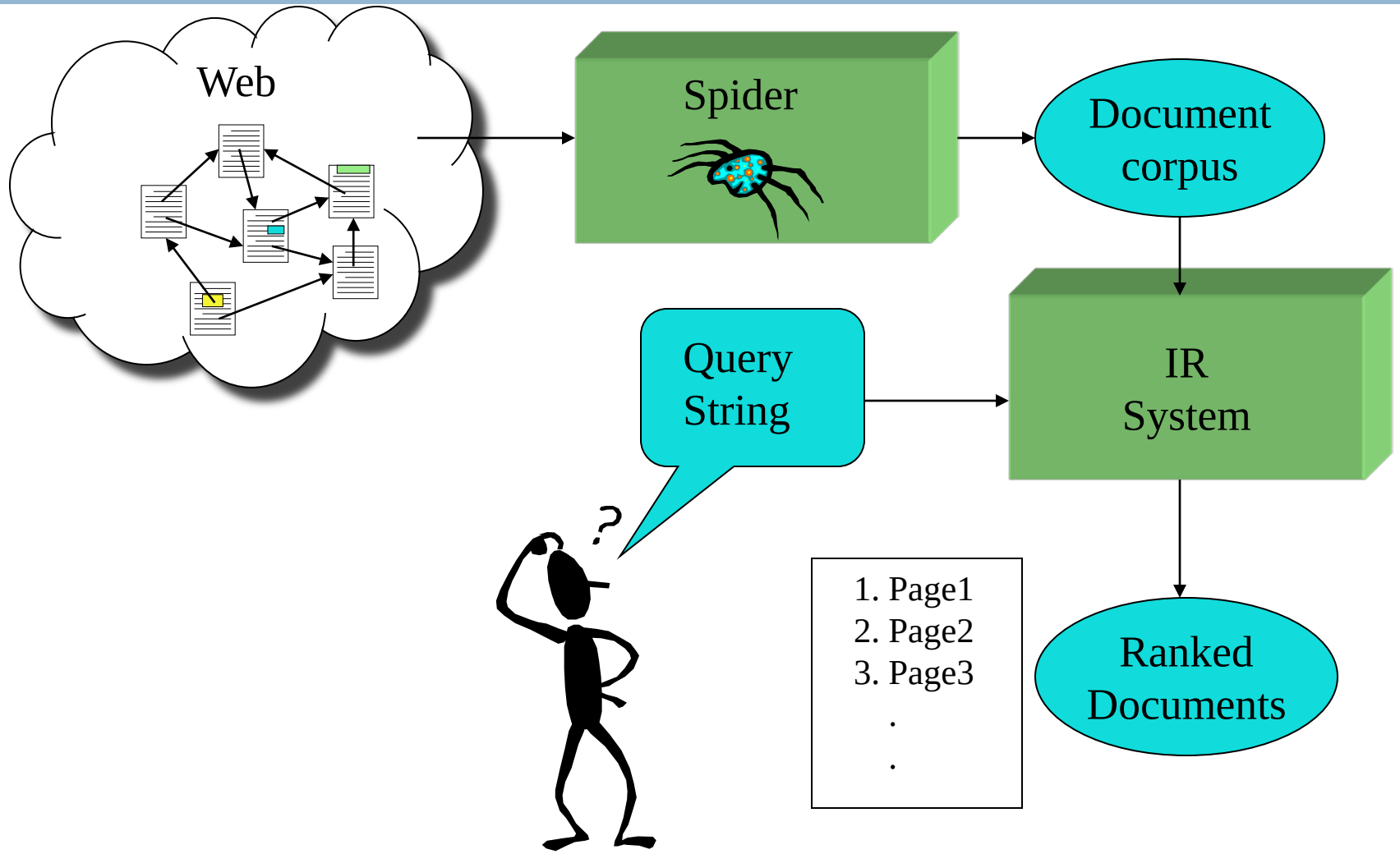
Typical IR Task

- Given:
  - A corpus of textual natural-language documents.
  - A user query in the form of a textual string.
- Find:
  - A ranked set of documents that are relevant to the query.

IR System


Document corpus
Query String
IR System
Ranked Documents
1. Doc1
2. Doc2
3. Doc3
.
.

Web Search System



Relevance

- Relevance is a subjective judgment and may include:
  - Being on the proper subject.
  - Being timely (recent information).
  - Being authoritative (from a trusted source).
  - Satisfying the goals of the user and his/her intended use of the information (*information need*).

Keyword Search

- Simplest notion of relevance is that the query string appears verbatim in the document.
- Slightly less strict notion is that the words in the query appear frequently in the document, in any order (*bag of words*).

Problems with Keywords

- May not retrieve relevant documents that include synonymous terms.
  - "restaurant" vs. "café"
  - "PRC" vs. "China"
- May retrieve irrelevant documents that include ambiguous terms.
  - "bat" (baseball vs. mammal)
  - "Apple" (company vs. fruit)
  - "bit" (unit of data vs. act of eating)

Relevant items are similar

- Much of IR depends upon idea that
  similar → relevant to same queries
- Usually measure query-document similarity
  - Can consider document-document similarity
- "Similar" can be measured in many ways
  - String matching
  - Same vocabulary
  - Probability arise from same model
  - Same meaning
  - …

“Bag of words”

- An effective and popular approach
- Compares words without regard to order

- Consider reordering words in a headline
  - Stocks fall on inflation fears
  - inflation stocks fall on fears
  - fall inflation stocks on fears
  - fall fears inflation stocks on
  - fall fears inflation on stocks

How effective is IR?

- Anecdotal evidence from Web search says…
  - Sometimes it works really well
  - Most of the time it fails impressively
- Research systems at about 25-50% accuracy
  - Controlled queries, document sets
  - Near-complete relevance judgments
  - Ad-hoc, 2-3 word queries (typical Web search)
  - Depends what you measure (best systems, TREC 1999)
    - 50% accuracy in top 5 documents
    - 40% accuracy in top 20 documents
    - 30% accuracy when all relevant could have been retrieved
  - Goes up 10% with much longer queries
- Seems like it could be much better…

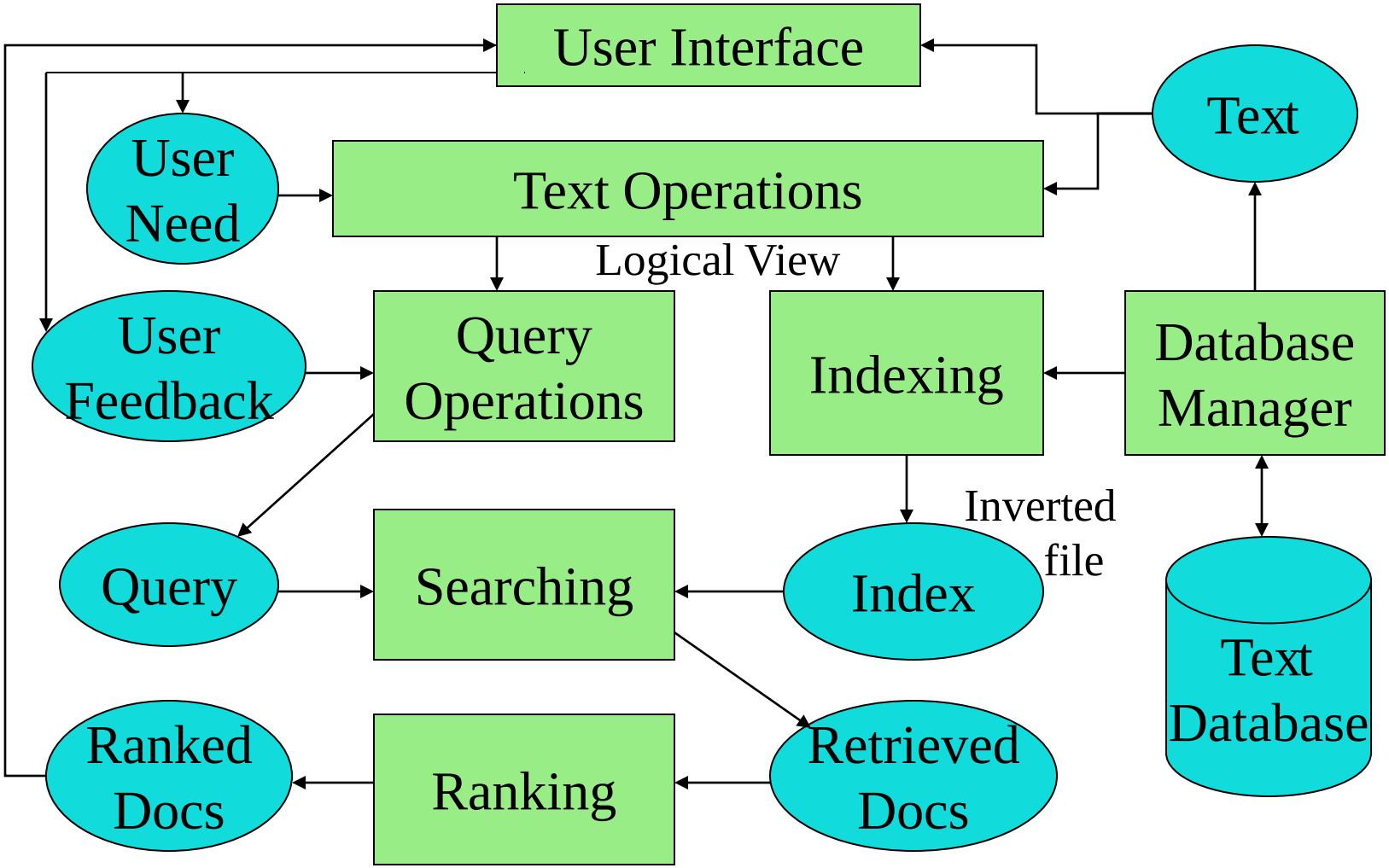
What could be missing?

- No effort to understand the text
- Simple word comparison seems flawed
  - *vice president* is not the same as *president* and *vice*
  - *Bill* as a name is not the same as a duck's *bill*
  - *George Bush* is the same as *President Bush*
  - A *run* in a stocking is different from a *5*-mile *run*
  - A *pen* for writing is not the same as a pig *pen*
- Surely IR effectiveness can be improved by addressing some of those problems

Intelligent IR

- Taking into account the *meaning* of the words used.
- Taking into account the *order* of words in the query.
- Adapting to the user based on direct or indirect feedback.
- Taking into account the *authority* of the source.

IR System Architecture



IR System Components

- Text Operations forms index words (tokens).
  - Tokenization
  - Stopword removal
  - Stemming
- Indexing constructs an *inverted index* of word to document pointers.
  - Mapping from keywords to document ids

I did enact Julius Caesar I was killed i' the Capitol; Brutus killed me.

Doc 1

So let it be with Caesar. The noble Brutus hath told you Caesar was ambitious

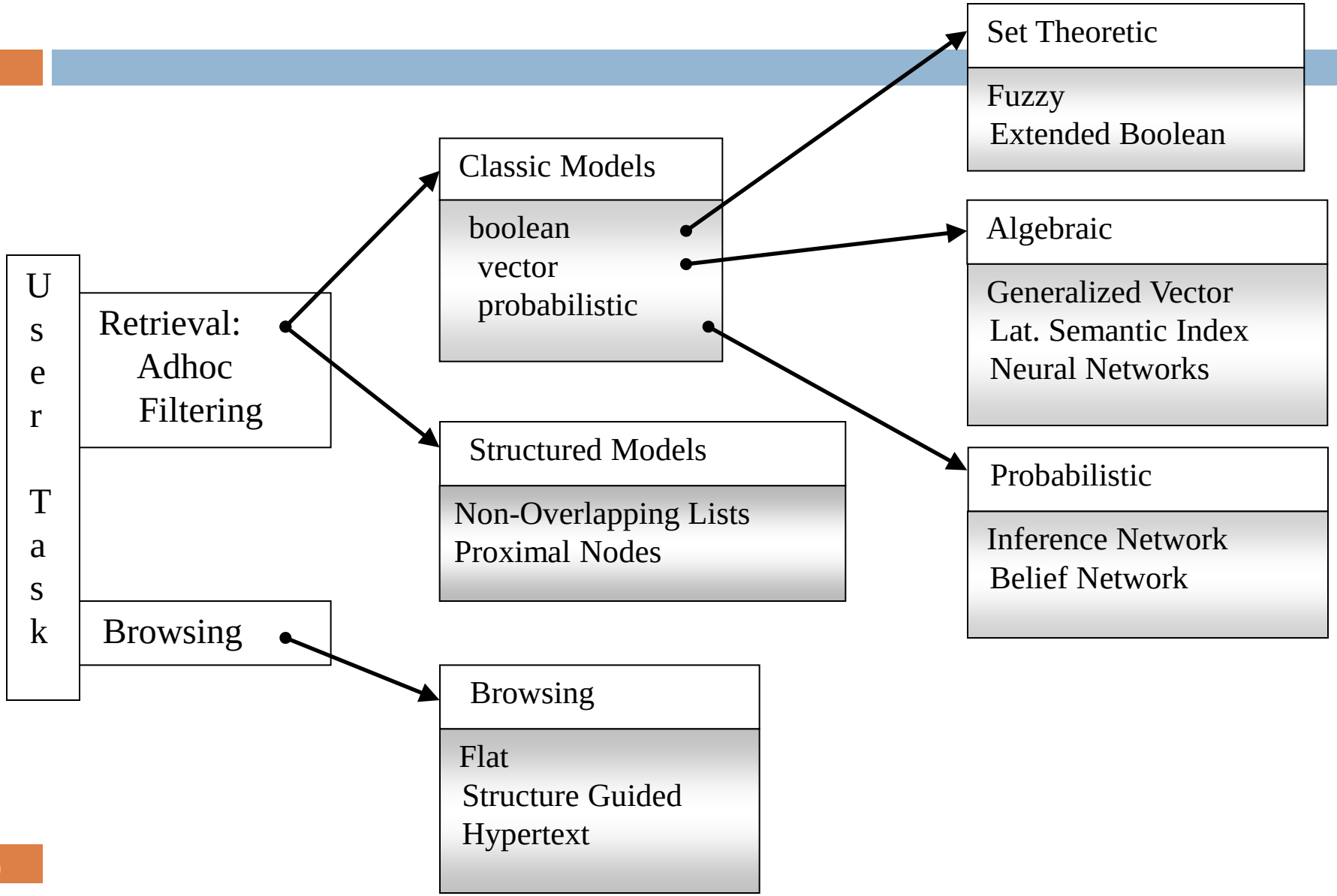
Doc 2

Term	Doc #
I	1
did	1
enact	1
julius	1
caesar	1
I	1
was	1
killed	1
i'	1
the	1
capitol	1
brutus	1
killed	1
me	1
so	2
let	2
it	2
be	2
with	2
caesar	2
the	2
noble	2
brutus	2
hath	2
told	2
you	2
caesar	2
was	2
ambitious	2

IR System Components

- Searching retrieves documents that contain a given query token from the inverted index.
- Ranking scores all retrieved documents according to a relevance metric.
- User Interface manages interaction with the user:
  - Query input and document output.
  - Relevance feedback.
  - Visualization of results.
- Query Operations transform the query to improve retrieval:
  - Query expansion using a thesaurus.
  - Query transformation using relevance feedback.

IR Models


User Task
Retrieval:
Adhoc
Filtering
Browsing
Classic Models
boolean
vector
probabilistic
Structured Models
Non-Overlapping Lists
Proximal Nodes
Browsing
Flat
Structure Guided
Hypertext
Set Theoretic
Fuzzy
Extended Boolean
Algebraic
Generalized Vector
Lat. Semantic Index
Neural Networks
Probabilistic
Inference Network
Belief Network

Classic IR Models - Basic Concepts

- Each document represented by a set of representative keywords or index terms
- An index term is a document word useful for remembering the document main themes
- Index terms may be selected to be only nouns, since nouns have meaning by themselves
  - Should reduce the size of the index
  - … But it requires the identification of nouns → Part of Speech tagger
- However, search engines assume that all words are index terms (full text representation)

Classic IR Models - Basic Concepts

- Not all terms are equally useful for representing the document contents: less frequent terms allow identifying a narrower set of documents
- The *importance* of the index terms is represented by weights associated to them
- Let
  - k_i be an index term
  - d_j be a document
  - w_{ij} is a weight associated with (k_i,d_j)
- The weight w_{ij} quantifies the importance of the index term for describing the document contents

Boolean Model

- Simple model based on set theory
- Queries specified as Boolean expressions
  - precise semantics
  - neat formalism
  - $q = ka \wedge (kb \vee \neg kc)$
- Terms are either present or absent. Thus, $w_{ij} \varepsilon \{0,1\}$

Drawbacks of the Boolean Model

- Retrieval based on binary decision criteria with no notion of partial matching
- No ranking of the documents is provided (absence of a grading scale)
- Information need has to be translated into a Boolean expression which most users find awkward
- The Boolean queries formulated by the users are most often too simplistic
- Frequently returns either too few or too many documents in response to a user query

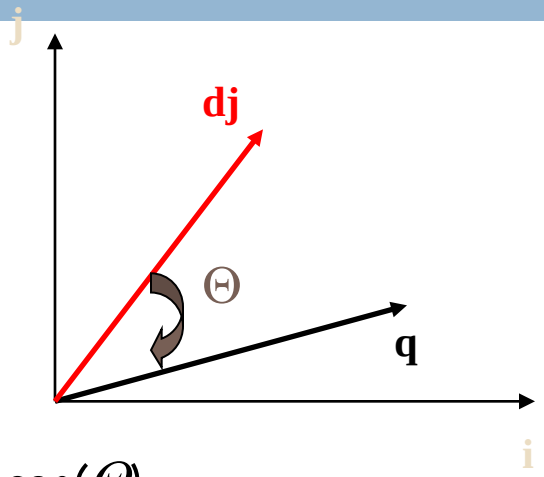
Vector-based Model

- Use of binary weights is too limiting
- Non-binary weights provide consideration for partial matches
- These term weights are used to compute a *degree of similarity* between a query and each document
- Ranked set of documents provides for better matching

Vector-based Model

- Define:
  - $w_{ij} > 0$ whenever $k_i \in d_j$
  - $w_{iq} >= 0$ associated with the pair (k_i, q)
  - $vec(d_j) = (w_{1j}, w_{2j}, ..., w_{tj})$
  - $vec(q) = (w_{1q}, w_{2q}, ..., w_{tq})$
  - To each term k_i is associated a unitary vector $vec(i)$
  - The unitary vectors $vec(i)$ and $vec(j)$ are assumed to be orthonormal (i.e., index terms are assumed to occur independently within the documents)
- The *t* unitary vectors $vec(i)$ form an orthonormal basis for a t-dimensional space
- In this space, queries and documents are represented as weighted vectors

Vector Based Model



- $Sim(q,dj) = cos(\Theta)$
  $= [vec(d_j) \bullet vec(q)] \ / \ |d_j| * |q|$
- $= [\Sigma \ w_{ij} * w_{iq}] \ / \ |d_j| * |q|$
- Since $w_{ij} > 0$ and $w_{iq} > 0$, $0 <= sim(q,d_j) <= 1$
- A document is retrieved even if it matches the query terms only partially

Vector Based Model

- $Sim(q,d_j) = [\Sigma\ w_{ij} * w_{iq}] \ /\ |d_j| * |q|$
- How to compute the weights w_{ij} and w_{iq} ?
- A good weight must take into account two effects:
  - quantification of intra-document contents (similarity)
    - *tf* factor, the *term frequency* within a document
  - quantification of inter-documents separation (dissimilarity)
    - *idf* factor, the *inverse document frequency*
  - $w_{ij} = tf(i,j) * idf(i)$

Probabilistic Model

- Objective: to capture the IR problem using a probabilistic framework
- Given a user query, there is an *ideal* answer set
- Guess at the beginning what they could be (i.e., guess initial description of ideal answer set)
- Improve by iteration

Probabilistic Model

- An initial set of documents is retrieved somehow
  - Can be done using vectorial model, boolean model
- User inspects these docs looking for the relevant ones (in truth, only top 10-20 need to be inspected)
- IR system uses this information to refine description of ideal answer set
- By repeting this process, it is expected that the description of the ideal answer set will improve
- Have always in mind the need to guess at the very beginning the description of the ideal answer set
- Description of ideal answer set is modeled in probabilistic terms

Probabilistic Ranking Principle

- Given a user query q and a document d_j, the probabilistic model tries to estimate the probability that the user will find the document d_j interesting (i.e., relevant).
- The model assumes that this probability of relevance depends on the query and the document representations only.
- Ideal answer set is referred to as R and should maximize the probability of relevance. Documents in the set R are predicted to be relevant.

Thank You – Questions?

Vasudeva Varma, IIIT Hyderabad

vv@iiit.ac.in or www.iiit.ac.in/~vasu