Information Retrieval Overview

Modern Information Retrieval by **R. Baeza-Yates and B. Ribeiro-Neto**
(Chapter 2)
Introduction to information retrieval by **Manning, Christopher D. et. al.**
(Chapter 1)

Information Retrieval

Usually, Information retrieval (IR) is defined as finding documents of an unstructured text that satisfies an information need from large document collections usually stored on computers

- As defined in this way, IR used to be an activity that only a few people engaged in: reference librarians, paralegals(변호(리)사보조원), and similar professional searchers
- Now, hundreds of millions of people engage in information retrieval every day when they use a web search engine or search their email

“Unstructured data” vs. “Structured data”

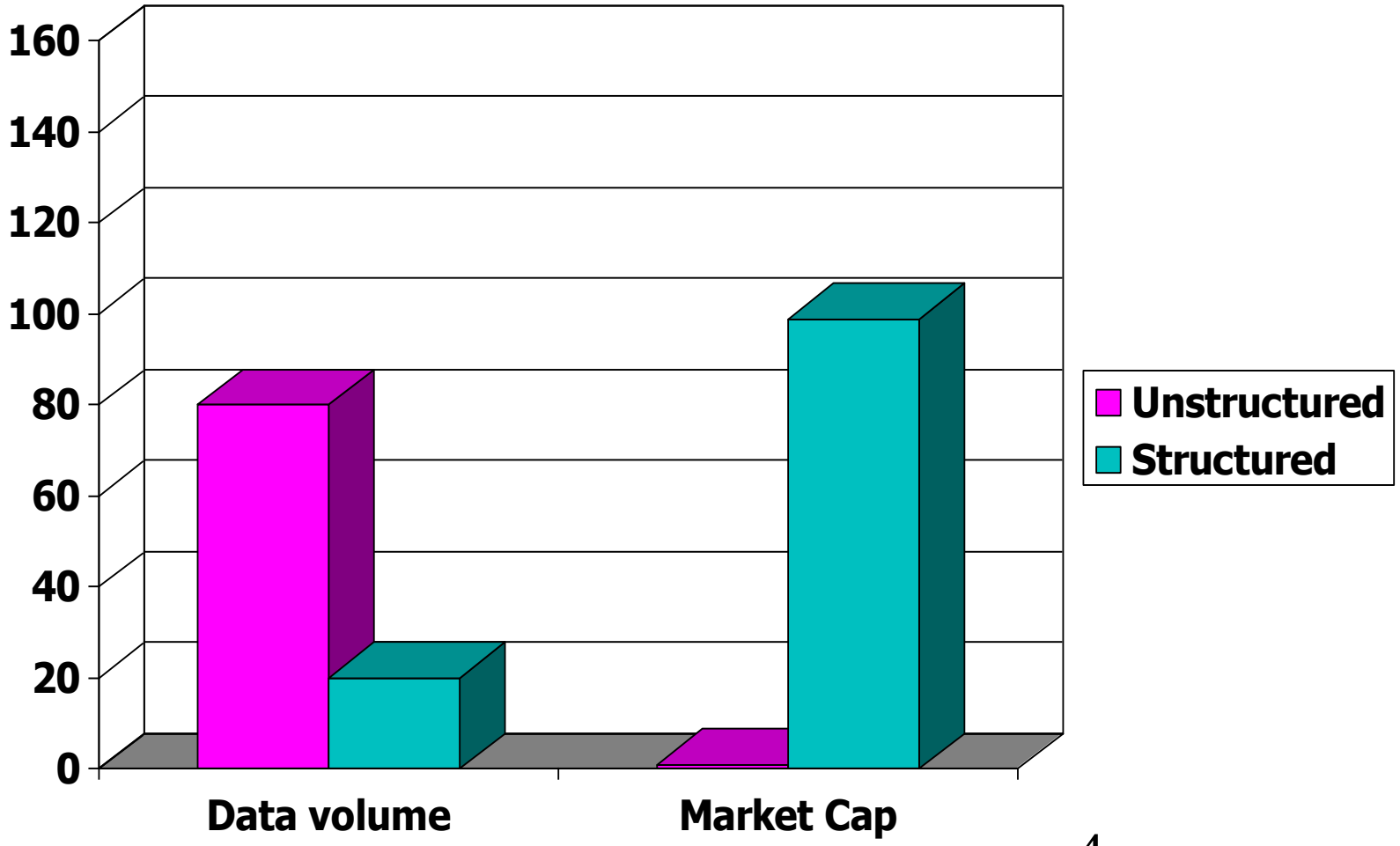
“Unstructured data” refers to data which does not have clear, semantically overt, easy-for-a-computer structure

- e.g. Texts

“Structured data” refers to data which does have clear, semantically overt, easy-for-a-computer structure

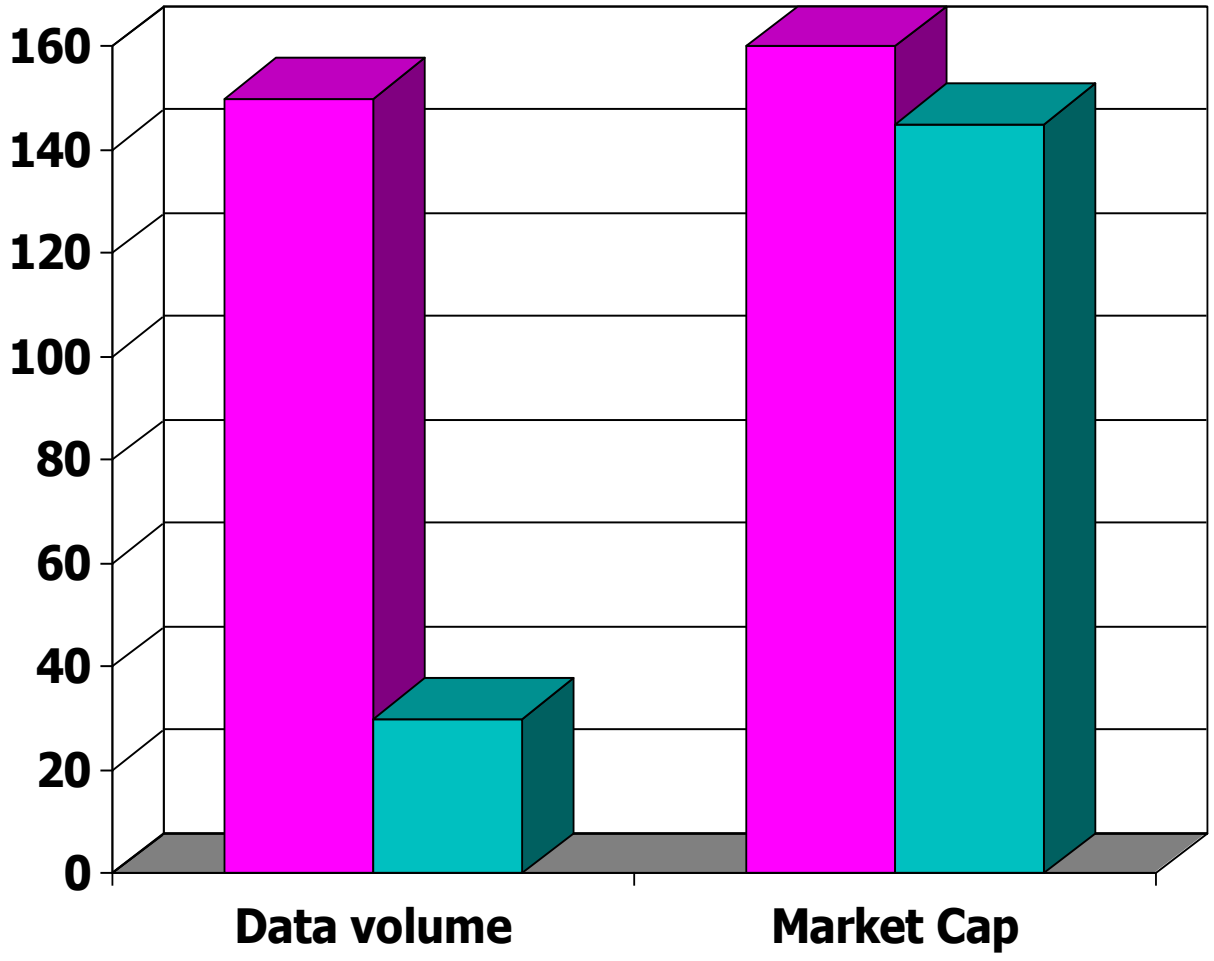
- e.g. Relational DB

Unstructured (text) vs. structured (database) data in 1996



4

Unstructured (text) vs. structured (database) data in 2006



5

DB usually searches in Structured data

Structured data tends to refer to information in “tables”

Employee	Manager	Salary
Smith	Jones	50000
Chang	Smith	60000
Ivy	Smith	50000

Typically DB allows numerical range and exact match (for discrete value) queries, e.g.,
find employees where Salary < 60,000 AND Manager = Smith.

6

IR often searches in Semi-structured data

In fact, many data are “semi-structured”

- e.g. this slide because it has distinctly identified zones such as the *Title* and *Bullets*
- e.g. html or xml documents
- e.g. Parsed text corpus

Typically, “semi-structured” search is facilitated such as

- *Title* contains data AND *Bullets* contain search
- <head> contains virus OR <body> contains computer
- <NP> contains Tom AND <VP> contains Jane

7

More sophisticated semi-structured search

Title begins with "Object Oriented" AND *Author contains* something like stro*rup

- where * is the wild-card operator

Issues:

- how do you process the wild-card operator?
- how do you rank retrieved results?
- Will be studied in later lectures

XML search or parsed text corpus search usually focus on semi-structured data

8

IR usually searches in Unstructured data

Typically refers to free text

Allows

- "keyword" queries (sometimes including operators (and, or))
- More sophisticated "concept" queries
  - e.g. find all web pages dealing with "*drug abuse"* irrespective of inclusion of "*drug*" and "*abuse*" words

Classic model for searching text documents usually focuses on unstructured data

9

Unstructured data: Example

Query: Which plays of Shakespeare contain the words ***Brutus*** *AND* ***Caesar*** but *NOT* ***Calpurnia***?

One could grep all of Shakespeare's plays for ***Brutus*** and ***Caesar,*** then strip out lines containing ***Calpurnia***?

- Slow (for large corpora)
- *NOT* ***Calpurnia*** is non-trivial

Other Query: find the word ***Romans*** near ***countrymen***)

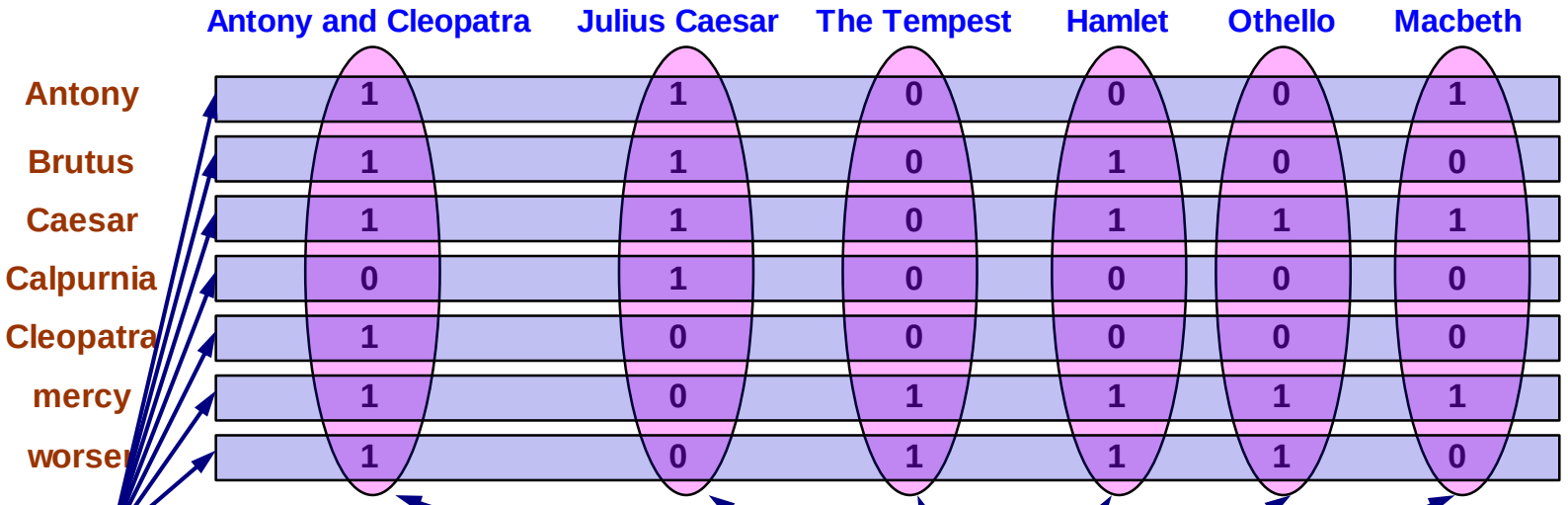
- not feasible when using grep

Another Problem: How to rank retrieved documents (Which are best documents and which are worst?)

- Will be studied in later lectures

10

Term-document incidence matrix



Term vectors

1 if term appears in play,
0 otherwise.

Document vectors

1 if play contains term,
0 otherwise.

Query: *Brutus* *AND* ***Caesar*** but *NOT* ***Calpurnia***

Term vectors

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	1	1	0	0	0	1
Brutus	1	1	0	1	0	0
Caesar	1	1	0	1	1	1
Calpurnia	0	1	0	0	0	0
Cleopatra	1	0	0	0	0	0
mercy	1	0	1	1	1	1
worser	1	0	1	1	1	0

We have a 0/1 vector for each term, called term vector.

To answer query: take the term vectors for ***Brutus, Caesar*** and NOT ***Calpurnia*** (complemented) ➔ bitwise *AND* and *NOT*.

110100 *AND* 110111 *AND* ~(010000) = 100100.

101111

Answers to query

Antony and Cleopatra, Act III, Scene ii

Agrippa [Aside to DOMITIUS ENOBARBUS]: Why, Enobarbus,
When Antony found Julius ***Caesar*** dead,
He cried almost to roaring; and he wept
When at Philippi he found ***Brutus*** slain.

Hamlet, Act III, Scene ii

Lord Polonius: I did enact Julius ***Caesar*** I was killed i' the
Capitol; ***Brutus*** killed me.

13

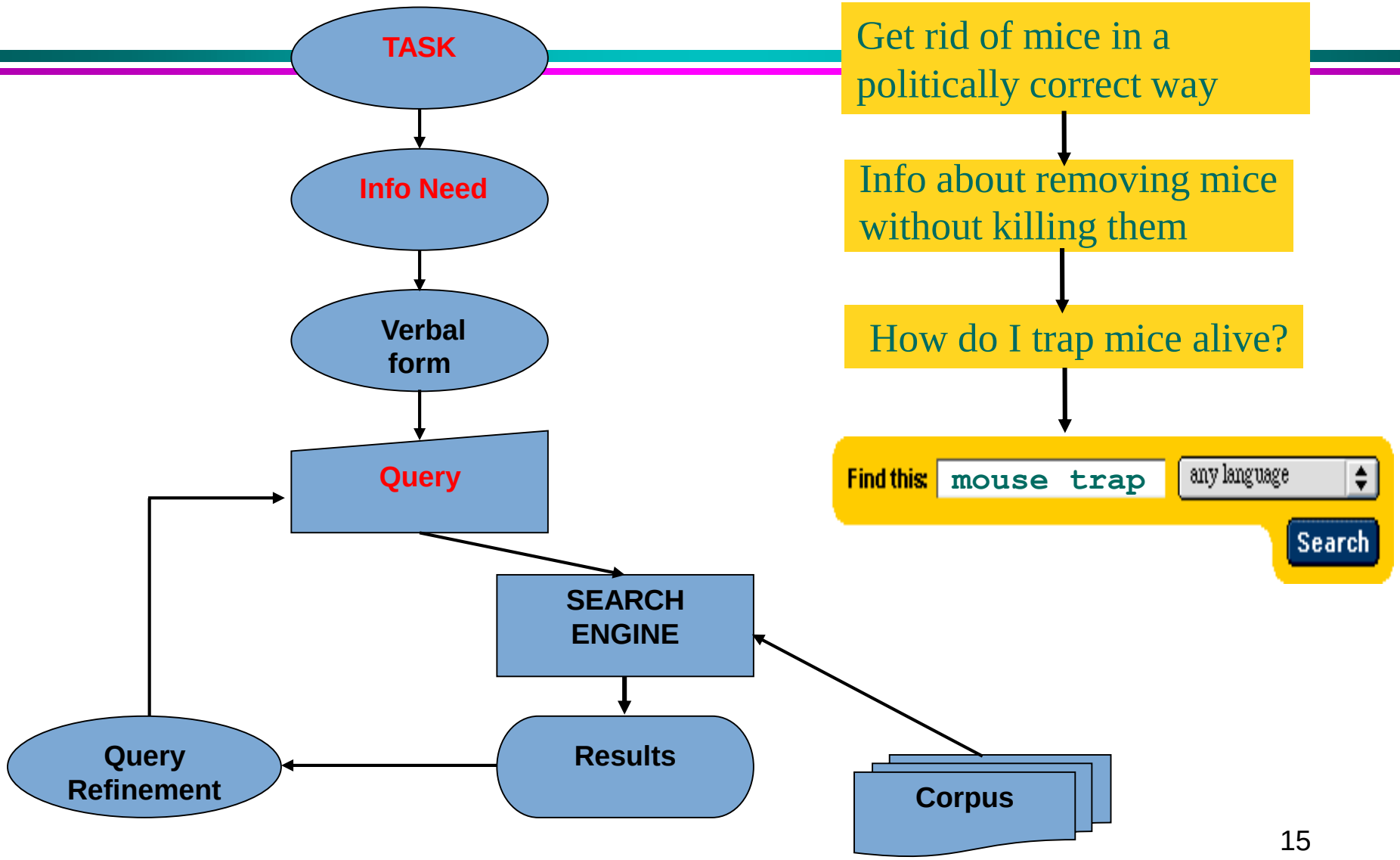
Basic assumptions of Information Retrieval

Document Collection: Fixed set of documents

Goal: Retrieve documents with information that is relevant to user's information need and helps him complete a task

- Generally, user's information need is formulated as a query as explained in the previous slides

The classic search procedure



15

Information Retrieval

Generic information retrieval systems

select and return to the user desired documents from a large set of documents in accordance with information need specified by the user

functions

- document search
  selects documents from an existing collection of documents for a query
- document routing/filtering
  disseminate incoming documents to appropriate users on the basis of user interest/preference profile

Information Need

Definition

a set of criteria specified by the user which describes the kind of information need.

- queries in document search task
- profiles in document routing task

forms

- keywords (used in most search engines)
- keywords with Boolean operators (used in most search engines)
- free text (used for finding most similar documents to the free text given.
  - e.g. detecting homework copies or cheating answer sheets of exams
- example documents (used for filtering incoming e-mails)
  - e.g. spam-filtering a large number of incoming e-mails on the basis of desired e-mail examples and undesired e-mail examples.

search vs. routing

The document search process generally matches a single Information Need against the stored corpus to return a subset of documents.

Document Routing generally matches a single document against a group of Profiles to determine which users are interested in that document.

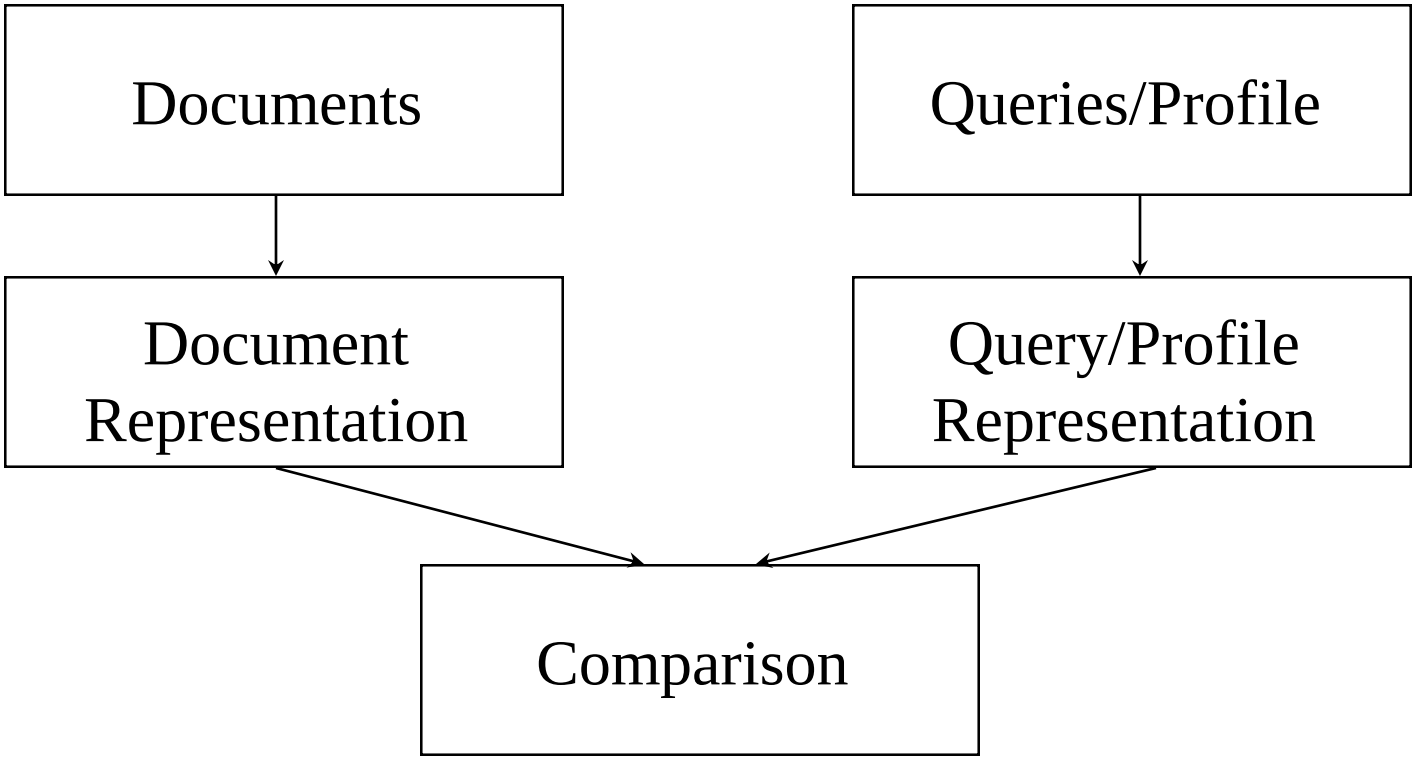
- (application examples) Mail Filtering, Content recommendation

Routing profiles stand for long-term expressions of user's information needs.

Search queries are ad-hoc(short-term criteria about user's information needs) in nature.

However, a basic search architecture can be often used for both the document search and document routing

Summary: Basic Architecture of an Information Retrieval System



Summary: Basic Architecture of an Information Retrieval System

Generate a representation of the meaning or content of *each document* based on its description.

Generate a representation (*as query or profile)* of the meaning of the *information need*.

Compare these two representations to select those documents that are most likely to match the information need.

Research Issues

Given a set of description for documents in the collection and a description of a user's information need, we must consider

Issue 1: What makes a good *document representation*?

- How can a representation be generated from a description of the document?
- What are retrievable units and how are they organized?

Research Issues (*Continued*)

Issue 2: How can we represent the information need?

- how can we acquire this representation
  - from a description of the information need? Or
  - through interaction with the user?

Issue 3

How can we *compare* two (document and information need) representations to judge likelihood that a document matches an information need?

Issue 4

How can we evaluate the effectiveness of the retrieval process or the goodness of retrieved results(documents)?

Issue 4 in more details: How good are the retrieved documents?

Precision : Fraction of retrieved documents that are relevant to user's information need

Recall : Fraction of relevant documents in collection that are retrieved

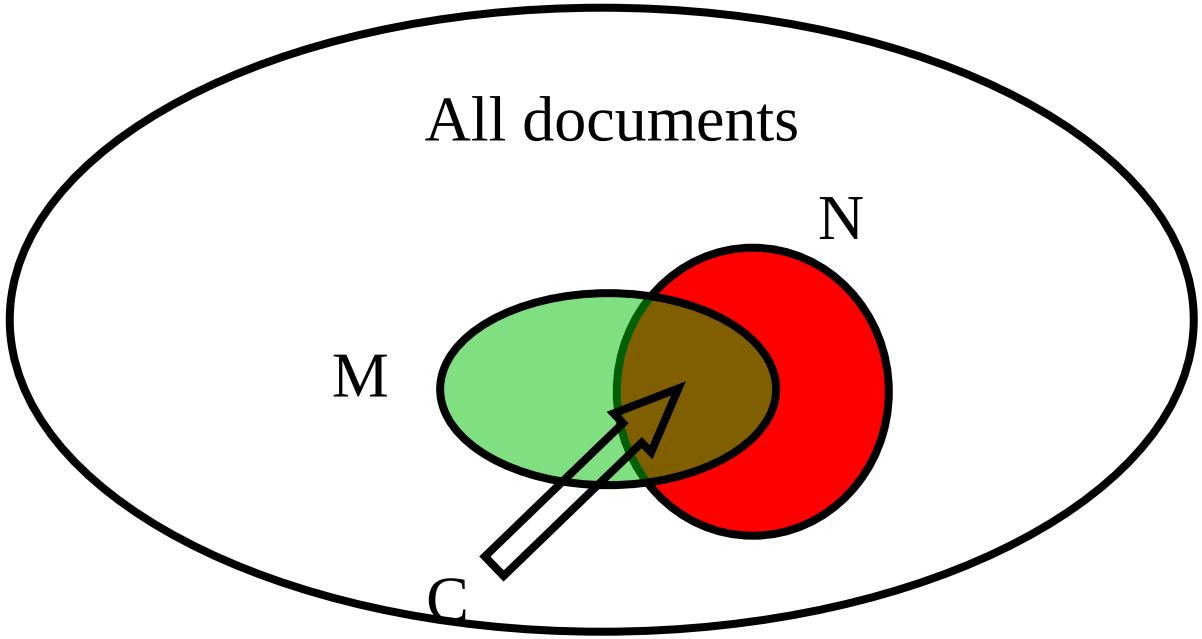
More precise definitions and measurements to follow in the next slide

23

N: Desired Documents

M: Retrieved Documents

C: Desired Documents that are actually retrieved



Precision(P): $\frac{C}{M}$

Recall(R): $\frac{C}{N}$

F-Value: $\frac{2P \cdot R}{P+R}$

More sophisticated information retrieval

Cross-language information retrieval

- e.g. query in Korean but retrieved documents in multi-languages

Question answering

- e.g. query: when is General Lee's birthday?

  answer: It is July 30

Summarization

- gets a document and produces its summary or theme automatically

Text mining

- finds(or mines) a number of important rules, patterns, or relationships from a large collection of text documents

Fortunately, we can apply the basic architecture of information retrieval systems or its variations for such more sophisticated information retrievals