INFORMATION RETRIEVAL AND SEMANTIC WEB 16B1NCI648

Lecture 3 and 4

CONTENTS TO BE COVERED

- Information retrieval Models
- Term Document Matrix
- Boolean retrieval incidence model
- Inverted Index Creation

Q1 : Consider the following documents,

Doc1: Data analysis deals with data sets

Doc2: Data mining in various fields

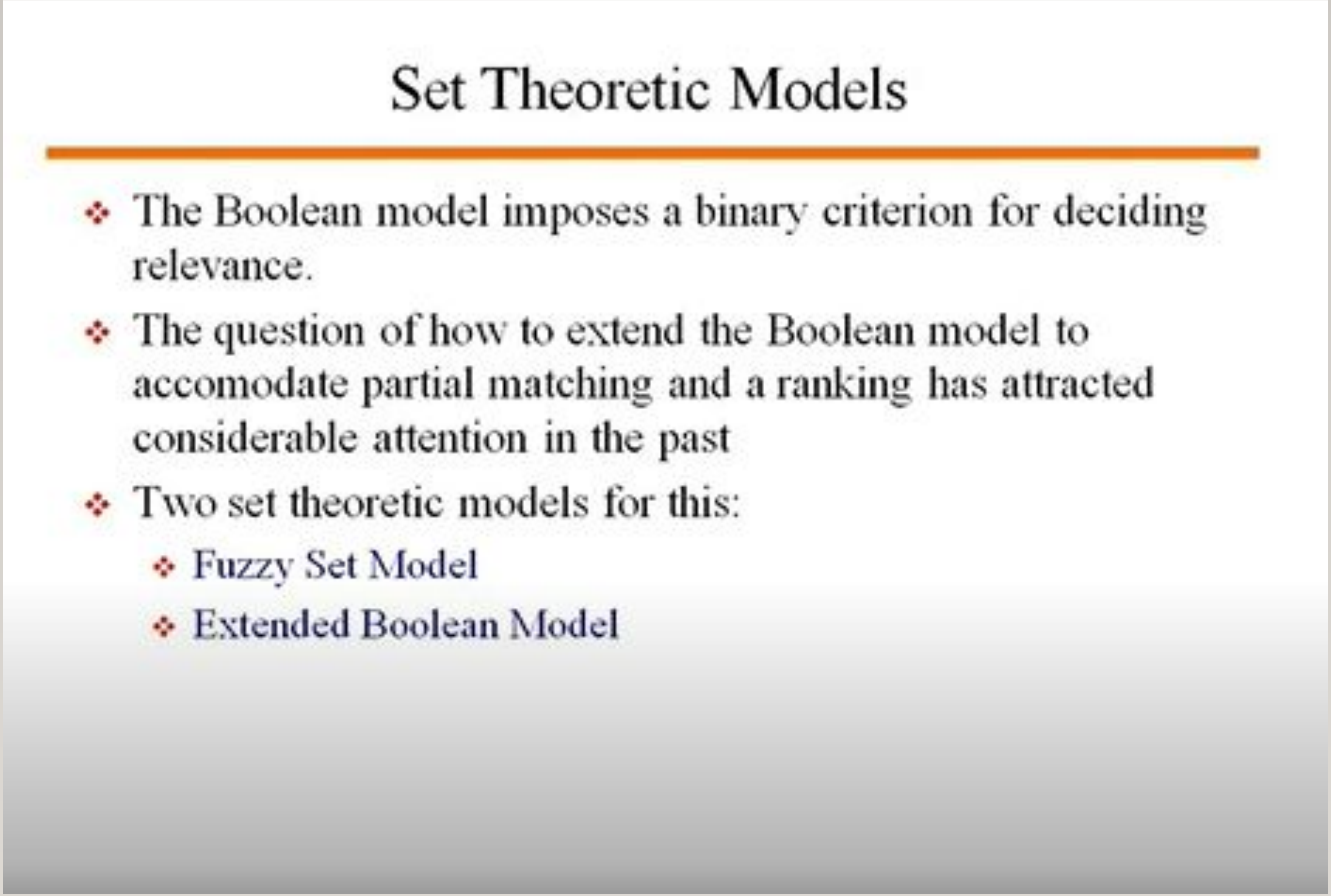
Doc3: Data analysis techniques and algorithms

Doc4: Data sets with temporal attributes

Construct Term Document incidence matrix and execute the query and find the resultant document that will be retrieved.

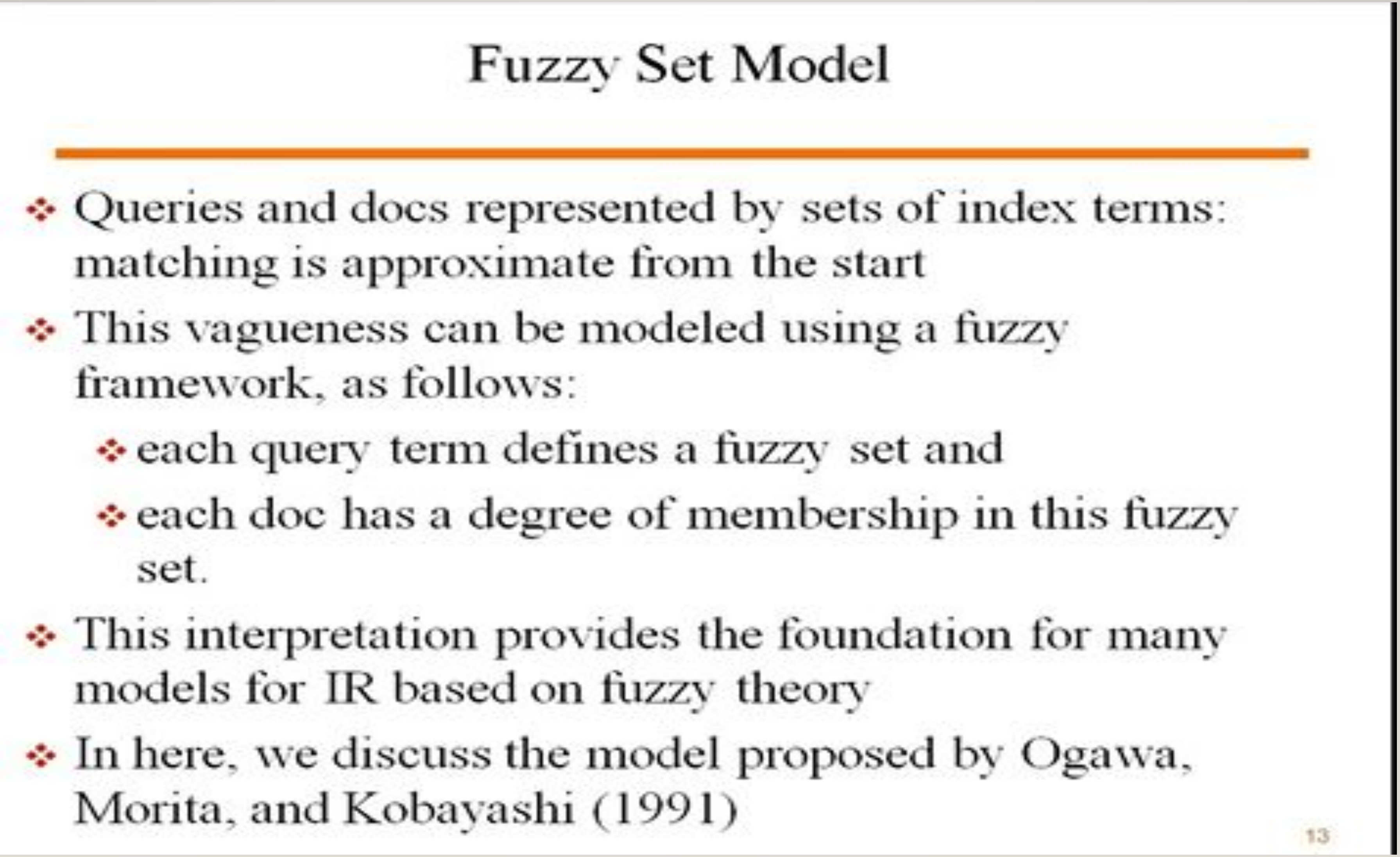
data AND (sets OR analysis)

data AND NOT (sets OR analysis)



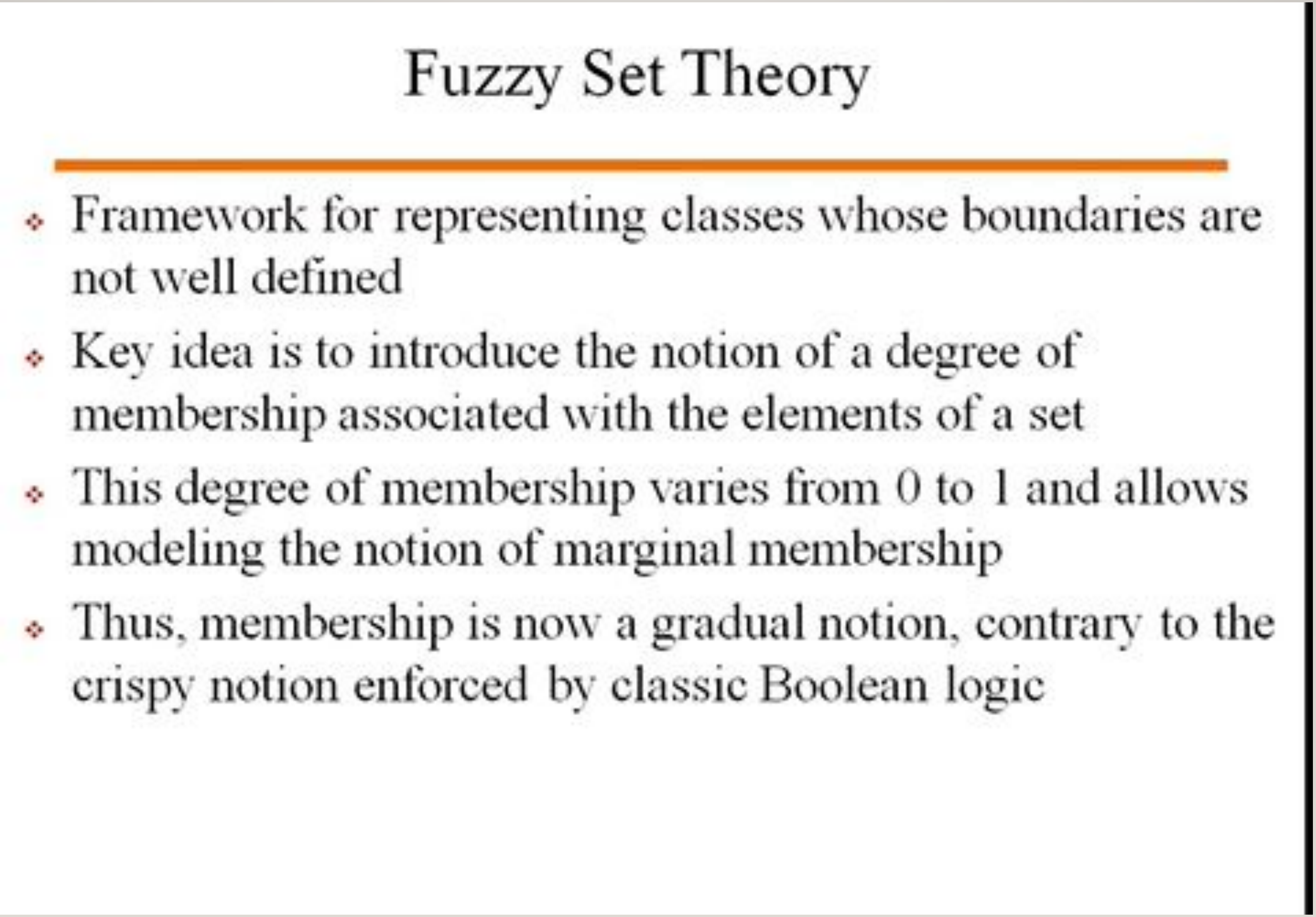
Set Theoretic Models

- The Boolean model imposes a binary criterion for deciding relevance.
- The question of how to extend the Boolean model to accomodate partial matching and a ranking has attracted considerable attention in the past
- Two set theoretic models for this:
  - Fuzzy Set Model
  - Extended Boolean Model



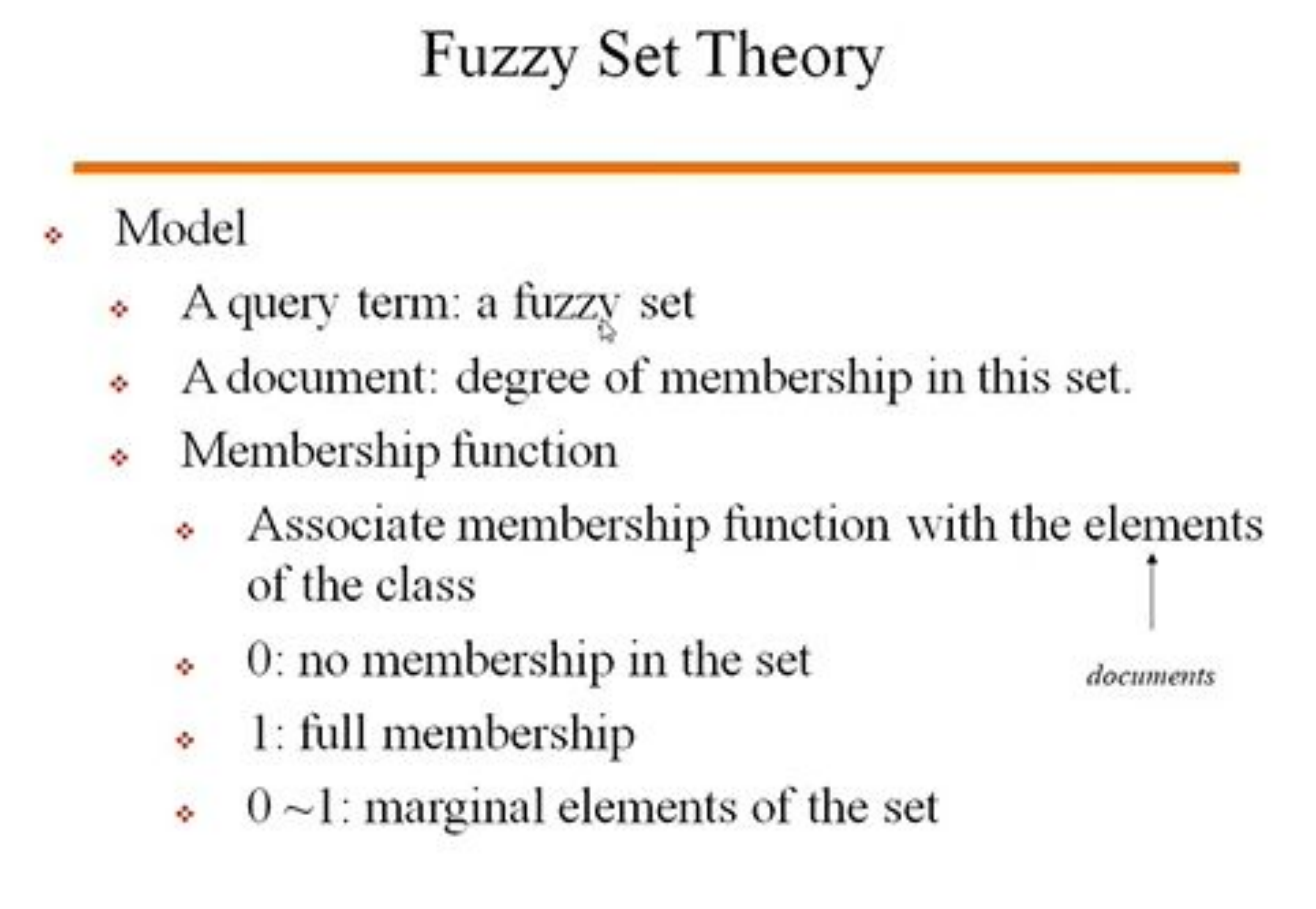
Fuzzy Set Model

- Queries and docs represented by sets of index terms: matching is approximate from the start
- This vagueness can be modeled using a fuzzy framework, as follows:
  - each query term defines a fuzzy set and
  - each doc has a degree of membership in this fuzzy set.
- This interpretation provides the foundation for many models for IR based on fuzzy theory
- In here, we discuss the model proposed by Ogawa, Morita, and Kobayashi (1991)



Fuzzy Set Theory

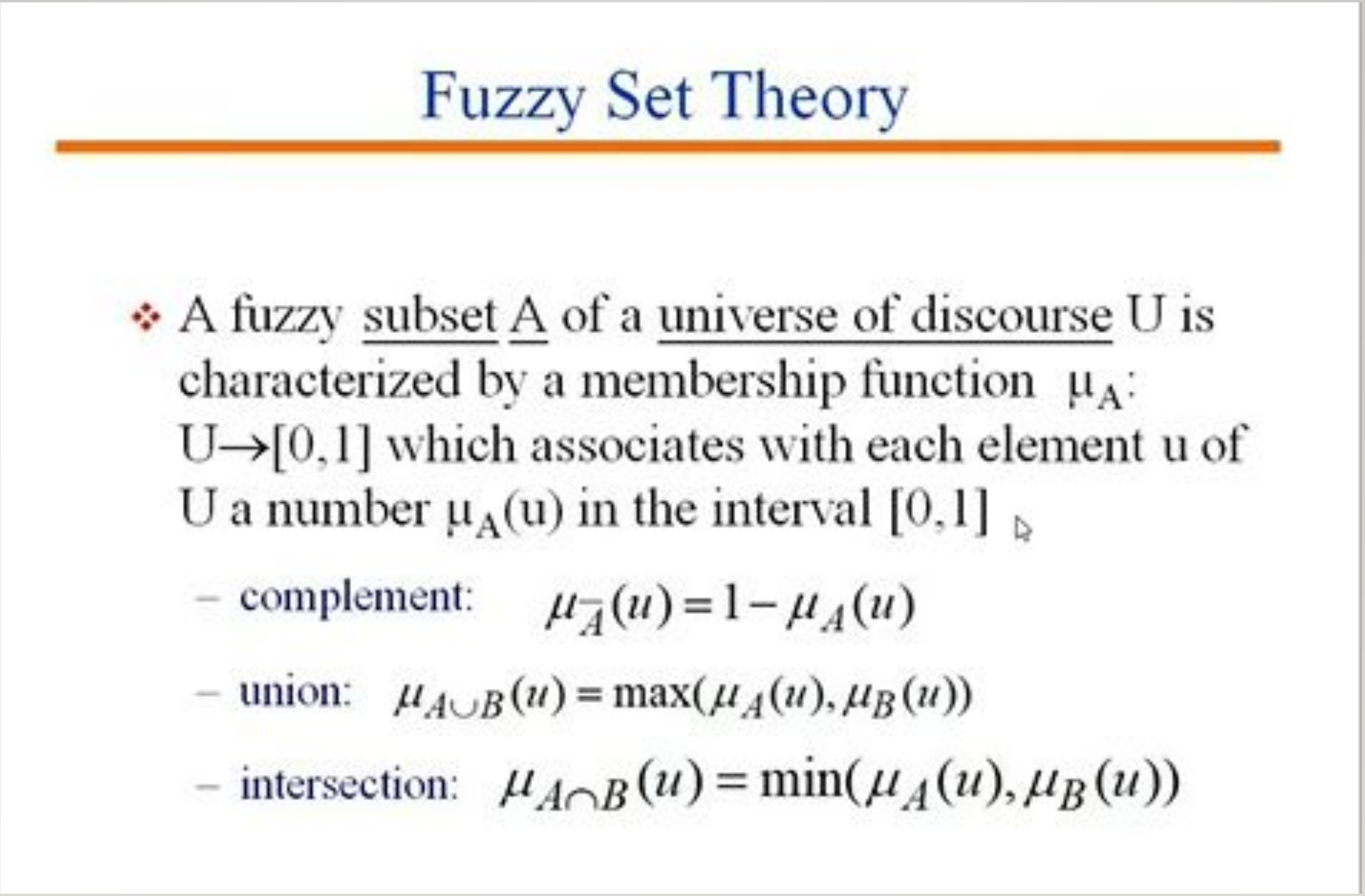
- Framework for representing classes whose boundaries are not well defined
- Key idea is to introduce the notion of a degree of membership associated with the elements of a set
- This degree of membership varies from 0 to 1 and allows modeling the notion of marginal membership
- Thus, membership is now a gradual notion, contrary to the crispy notion enforced by classic Boolean logic



Fuzzy Set Theory

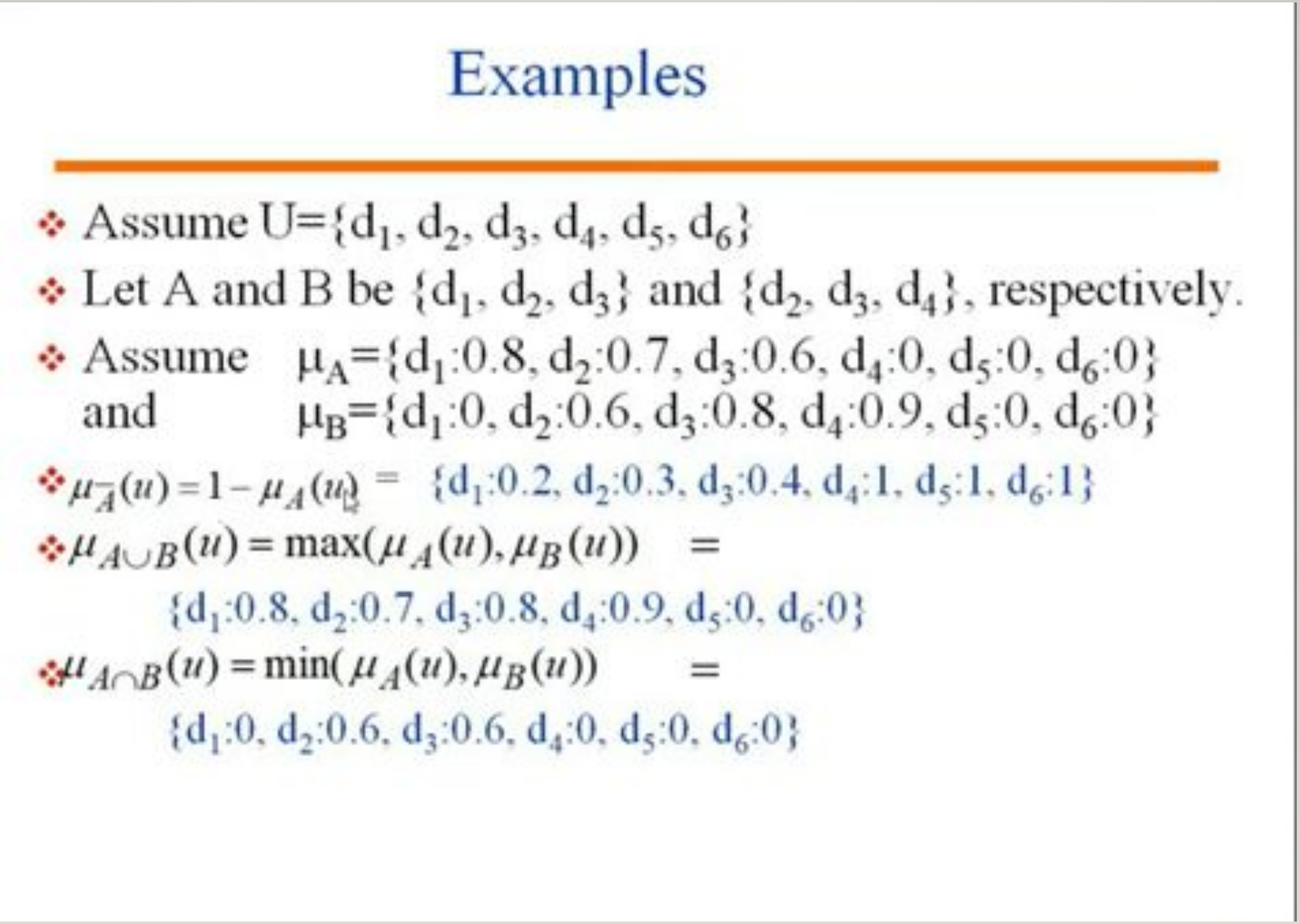
- Model
  - A query term: a fuzzy set
  - A document: degree of membership in this set.
  - Membership function
    - Associate membership function with the elements of the class
    - 0: no membership in the set
    - 1: full membership
    - 0 ~1: marginal elements of the set

documents



Fuzzy Set Theory

- A fuzzy subset A of a universe of discourse U is characterized by a membership function μ_A: U→[0,1] which associates with each element u of U a number $\mu_A(u)$ in the interval [0,1]
  - complement: $\mu_{\overline{A}}(u) = 1 - \mu_A(u)$
  - union: $\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$
  - intersection: $\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$

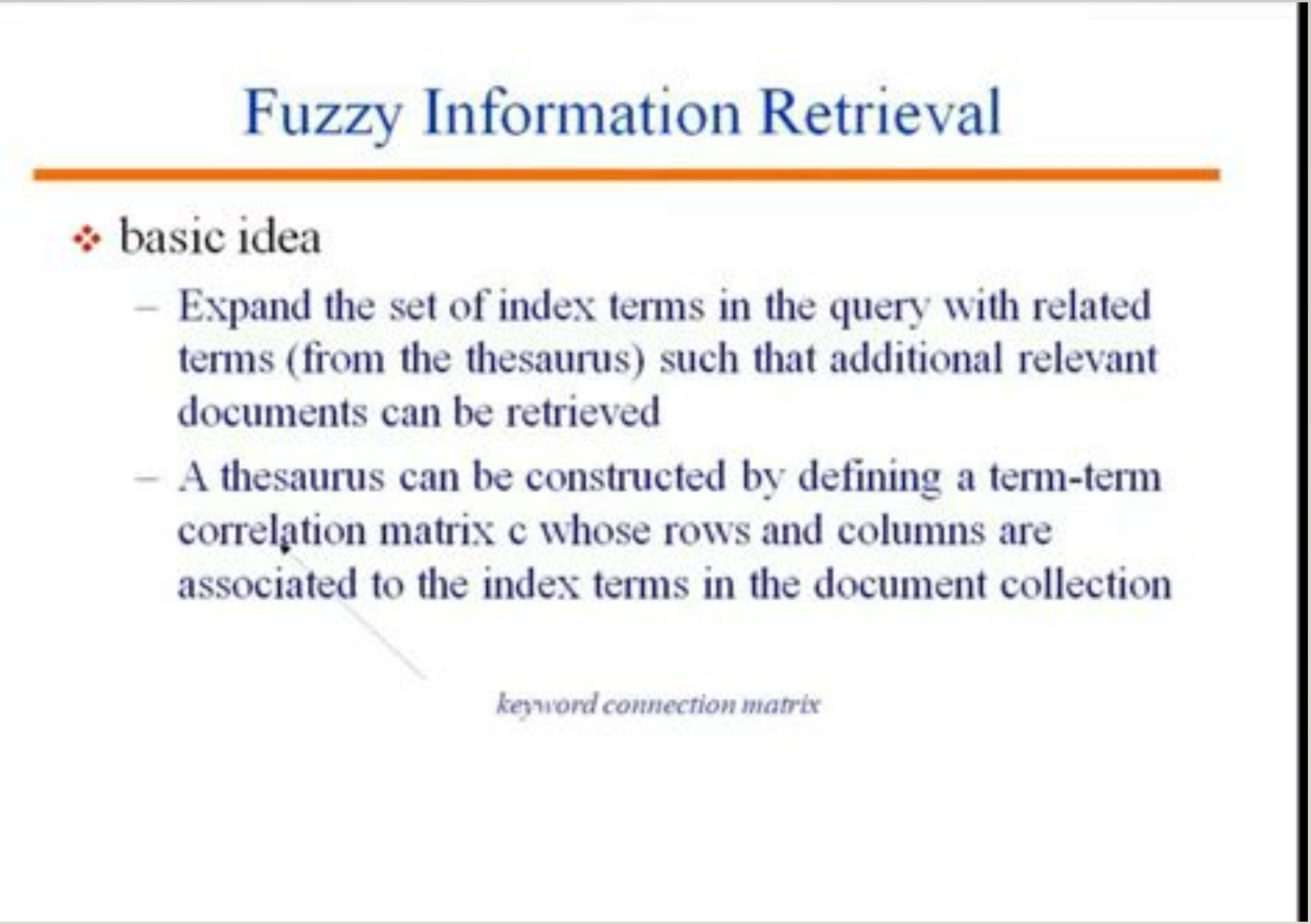


Examples

- Assume U={d_1, d_2, d_3, d_4, d_5, d_6}
- Let A and B be {d_1, d_2, d_3} and {d_2, d_3, d_4}, respectively.
- Assume μ_A={d_1:0.8, d_2:0.7, d_3:0.6, d_4:0, d_5:0, d_6:0}
  and μ_B={d_1:0, d_2:0.6, d_3:0.8, d_4:0.9, d_5:0, d_6:0}
- $\mu_{\overline{A}}(u) = 1 - \mu_A(u)$ = {d_1:0.2, d_2:0.3, d_3:0.4, d_4:1, d_5:1, d_6:1}
- $\mu_{A \cup B}(u) = \max(\mu_A(u), \mu_B(u))$ =

  {d_1:0.8, d_2:0.7, d_3:0.8, d_4:0.9, d_5:0, d_6:0}
- $\mu_{A \cap B}(u) = \min(\mu_A(u), \mu_B(u))$ =

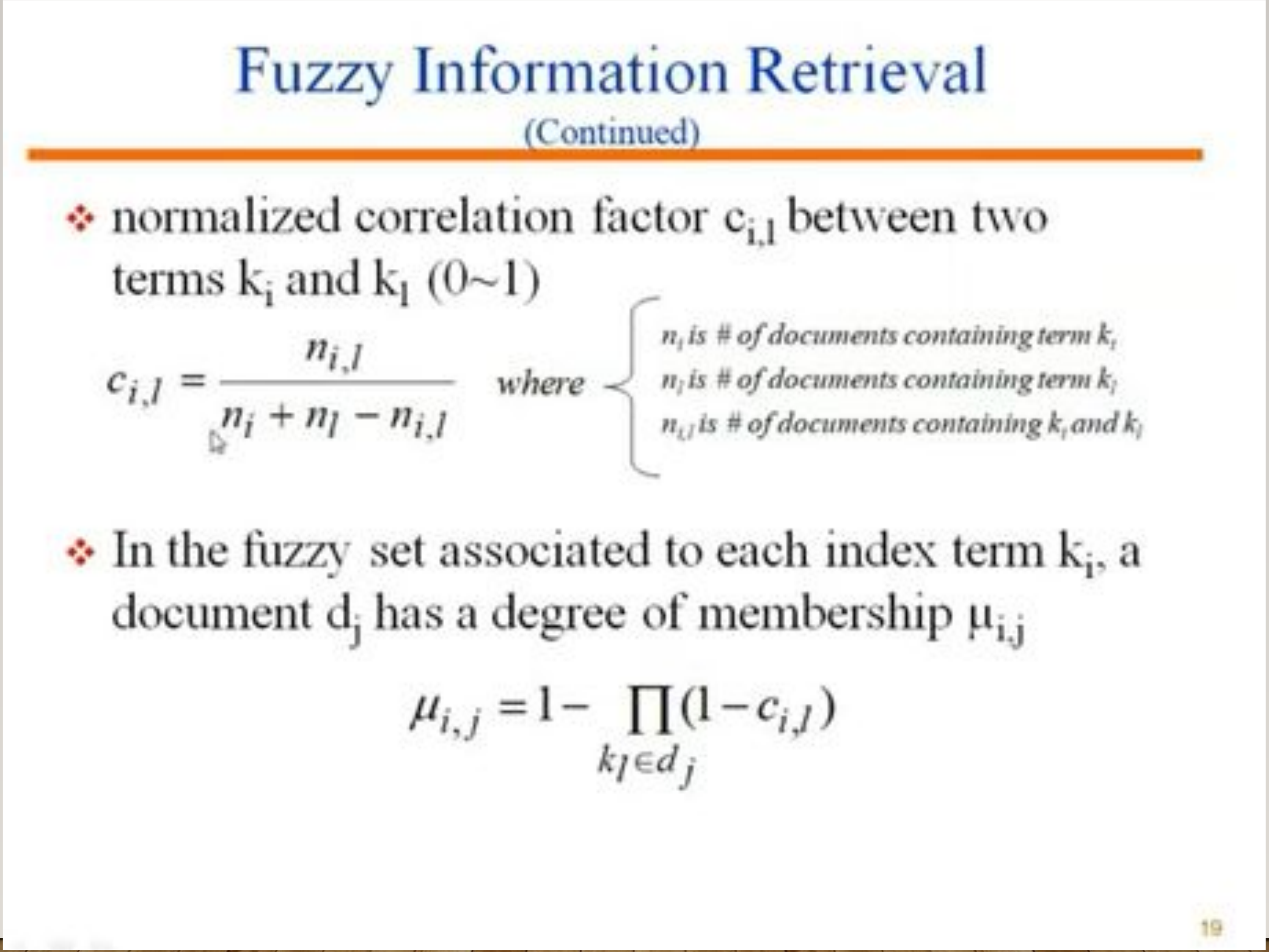
  {d_1:0, d_2:0.6, d_3:0.6, d_4:0, d_5:0, d_6:0}



Fuzzy Information Retrieval

- basic idea
  - Expand the set of index terms in the query with related terms (from the thesaurus) such that additional relevant documents can be retrieved
  - A thesaurus can be constructed by defining a term-term correlation matrix c whose rows and columns are associated to the index terms in the document collection

keyword connection matrix



Fuzzy Information Retrieval

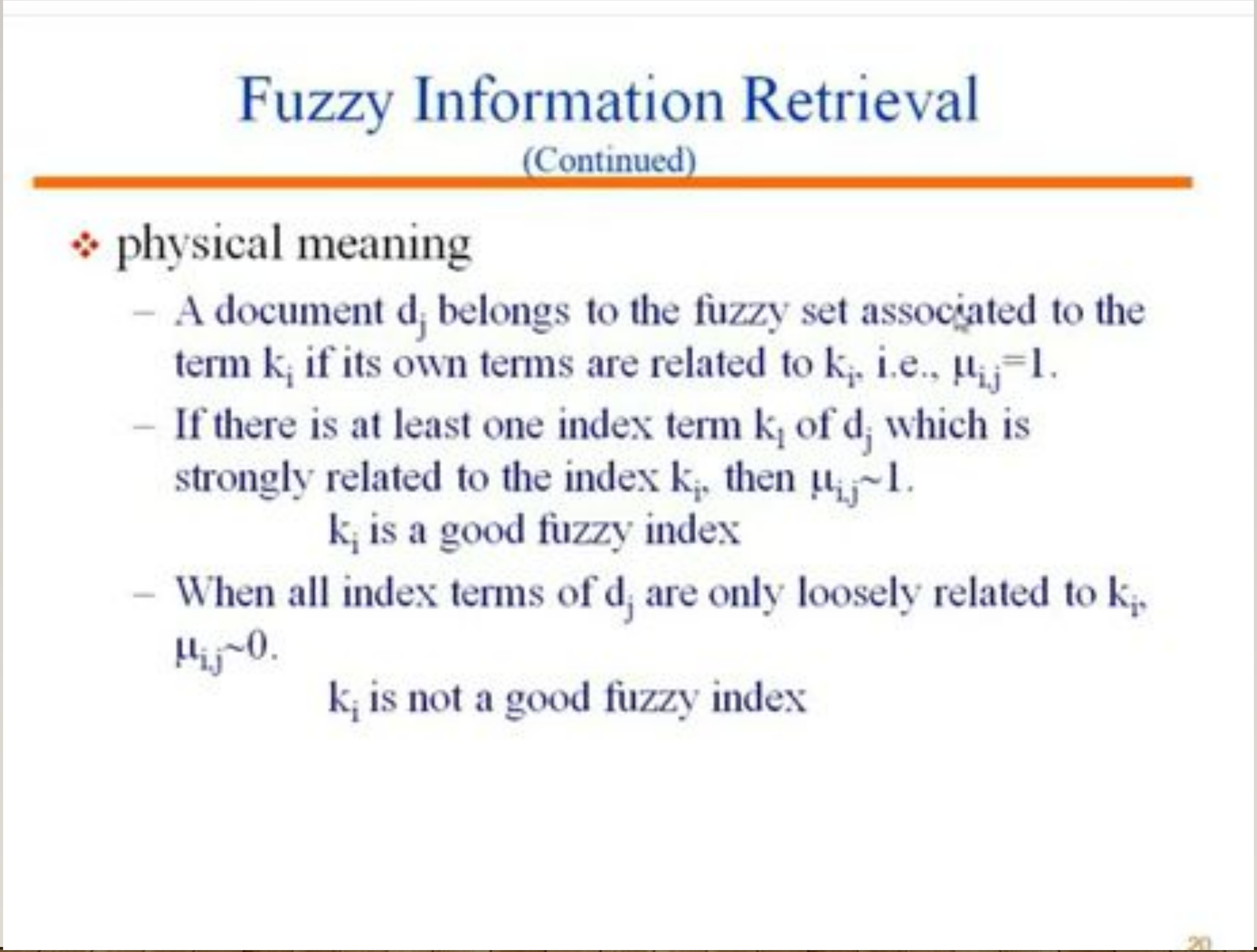
(Continued)

- normalized correlation factor $c_{i,l}$ between two terms k_i and k_l (0~1)

$$c_{i,l} = \frac{n_{i,l}}{n_i + n_l - n_{i,l}} \quad where \begin{cases} n_i \text{ is \# of documents containing term } k_i \\ n_l \text{ is \# of documents containing term } k_l \\ n_{i,l} \text{ is \# of documents containing } k_i \text{ and } k_l \end{cases}$$

- In the fuzzy set associated to each index term k_i, a document d_j has a degree of membership $\mu_{i,j}$

$$\mu_{i,j} = 1 - \prod_{k_l \in d_j} (1 - c_{i,l})$$



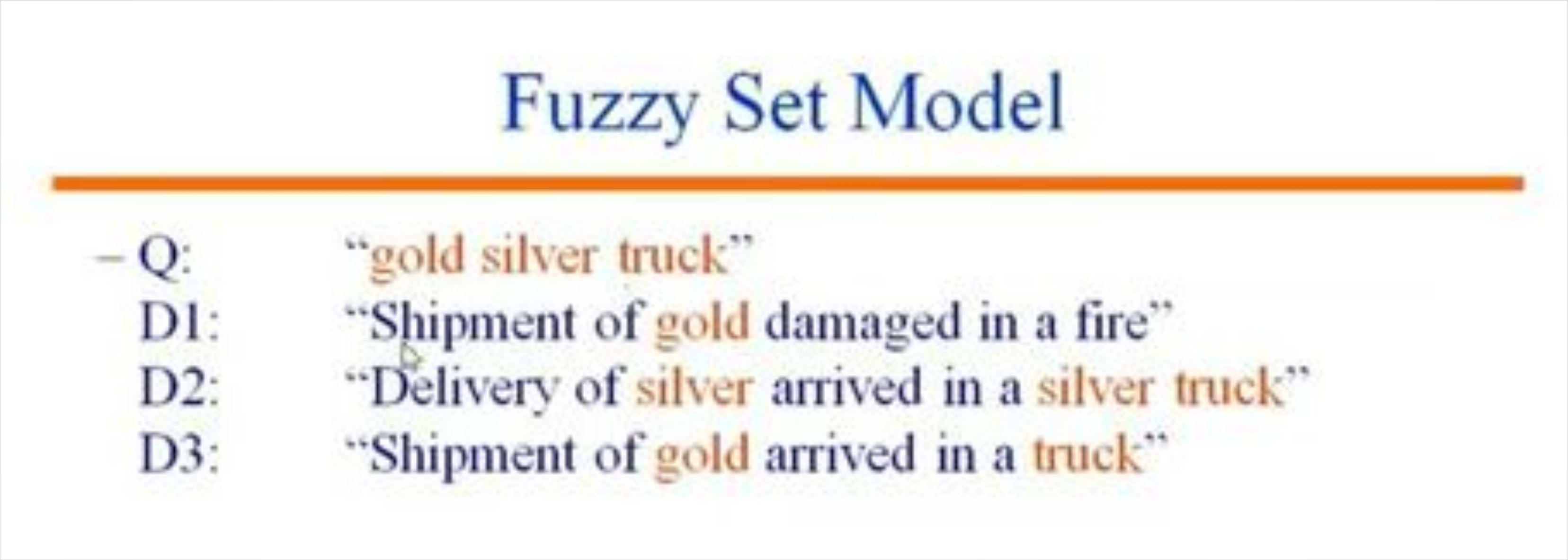
Fuzzy Information Retrieval

(Continued)

- physical meaning
  - A document d_j belongs to the fuzzy set associated to the term k_i if its own terms are related to k_i, i.e., $\mu_{i,j}=1$.
  - If there is at least one index term k_l of d_j which is strongly related to the index k_i, then $\mu_{i,j}\sim1$.

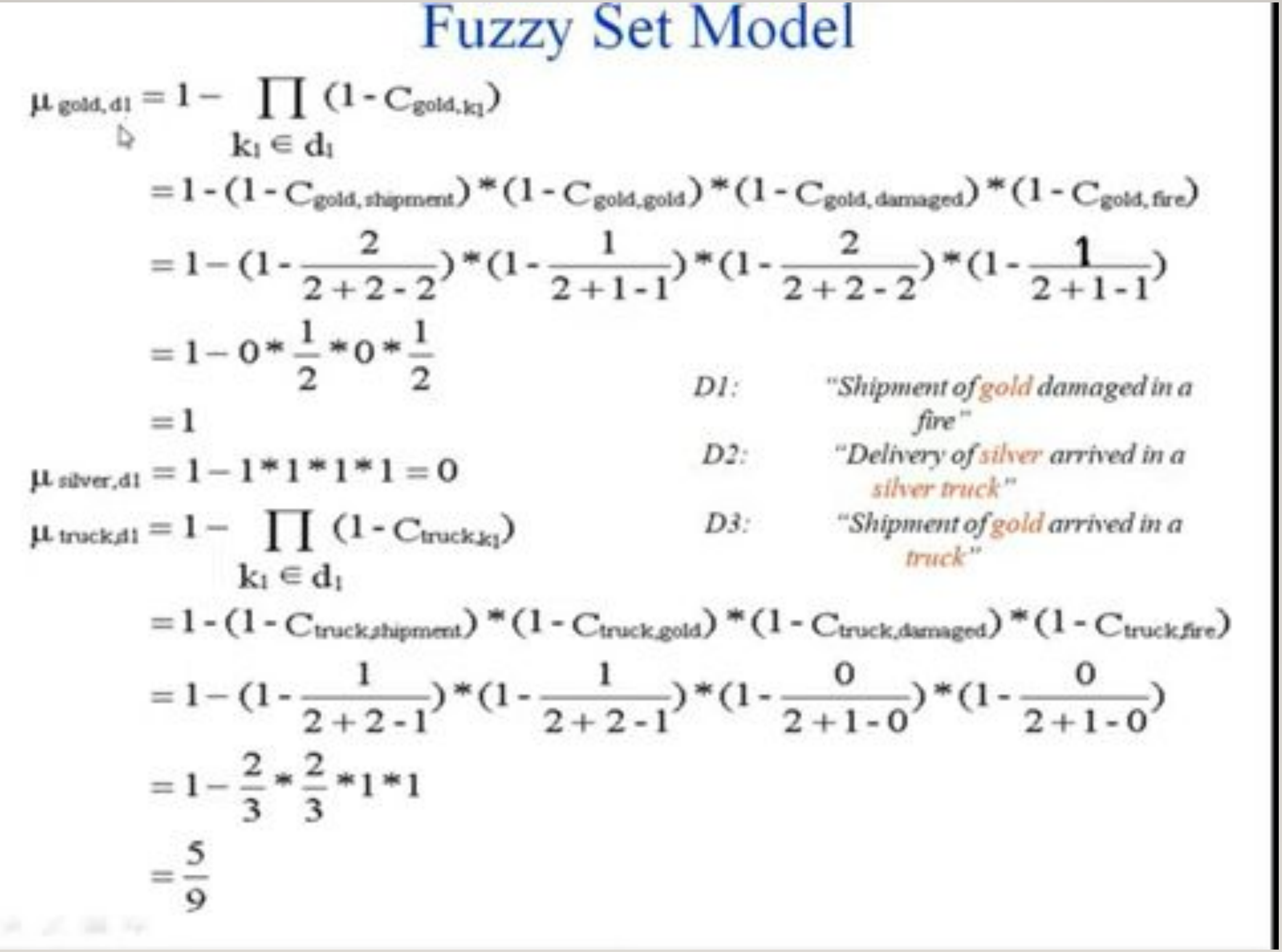
    k_i is a good fuzzy index
  - When all index terms of d_j are only loosely related to k_i, $\mu_{i,j}\sim0$.

    k_i is not a good fuzzy index



Fuzzy Set Model

- Q: “gold silver truck”
  D1: “Shipment of gold damaged in a fire”
  D2: “Delivery of silver arrived in a silver truck”
  D3: “Shipment of gold arrived in a truck”



Fuzzy Set Model

$$\mu_{gold,d1} = 1 - \prod_{k_1 \in d_1} (1 - C_{gold,k1})$$

$$= 1 - (1 - C_{gold,shipment}) * (1 - C_{gold,gold}) * (1 - C_{gold,damaged}) * (1 - C_{gold,fire})$$

$$= 1 - (1 - \frac{2}{2+2-2}) * (1 - \frac{1}{2+1-1}) * (1 - \frac{2}{2+2-2}) * (1 - \frac{1}{2+1-1})$$

$$= 1 - 0 * \frac{1}{2} * 0 * \frac{1}{2}$$

$$= 1$$

$$\mu_{silver,d1} = 1 - 1*1*1*1 = 0$$

$$\mu_{truck,d1} = 1 - \prod_{k_1 \in d_1} (1 - C_{truck,k1})$$

$$= 1 - (1 - C_{truck,shipment}) * (1 - C_{truck,gold}) * (1 - C_{truck,damaged}) * (1 - C_{truck,fire})$$

$$= 1 - (1 - \frac{1}{2+2-1}) * (1 - \frac{1}{2+2-1}) * (1 - \frac{0}{2+1-0}) * (1 - \frac{0}{2+1-0})$$

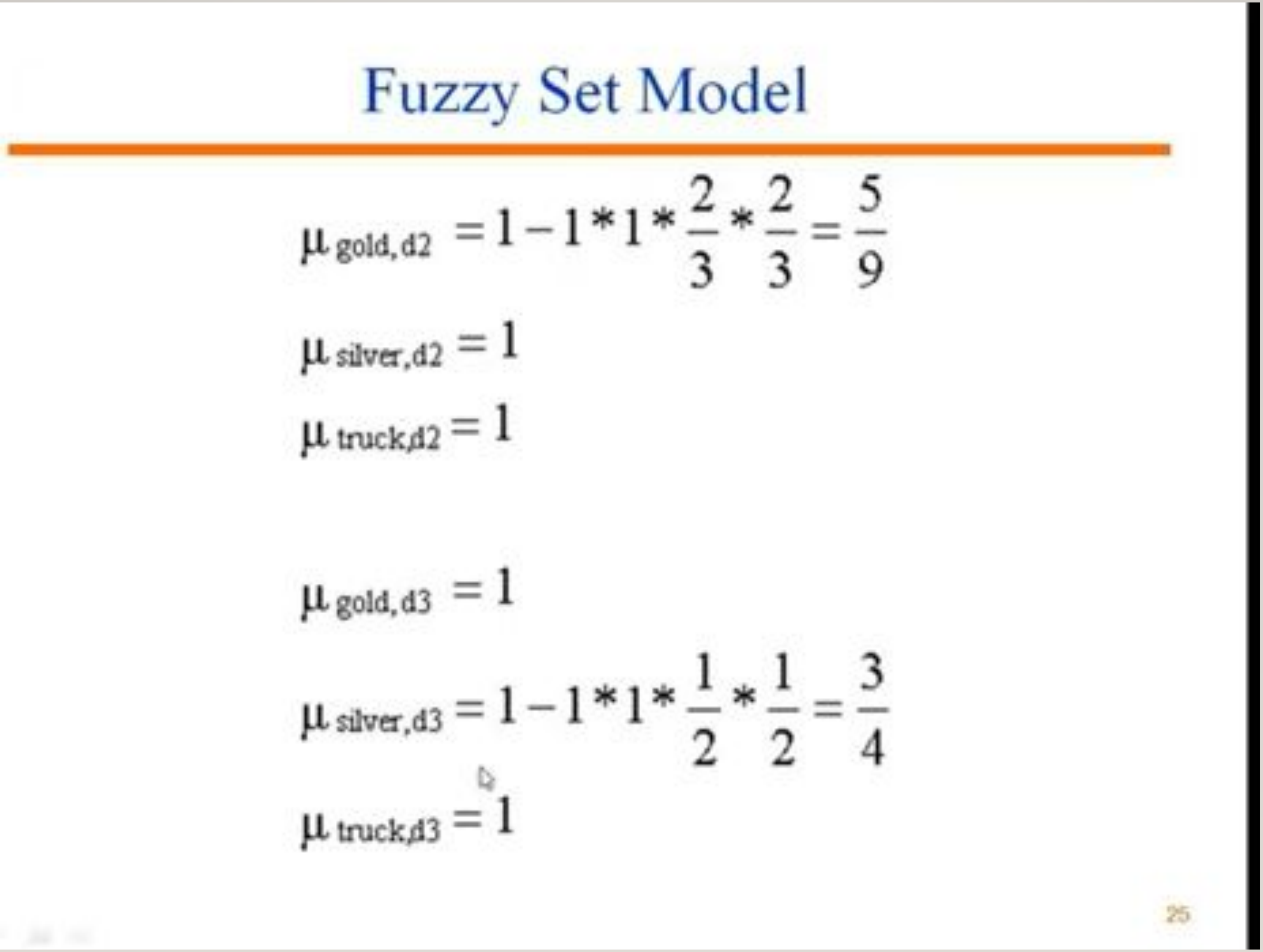
$$= 1 - \frac{2}{3} * \frac{2}{3} * 1 * 1$$

$$= \frac{5}{9}$$

D1: *"Shipment of gold damaged in a fire"*

D2: *"Delivery of silver arrived in a silver truck"*

D3: *"Shipment of gold arrived in a truck"*



Fuzzy Set Model

$$\mu_{\text{gold,d2}} = 1 - 1 * 1 * \frac{2}{3} * \frac{2}{3} = \frac{5}{9}$$

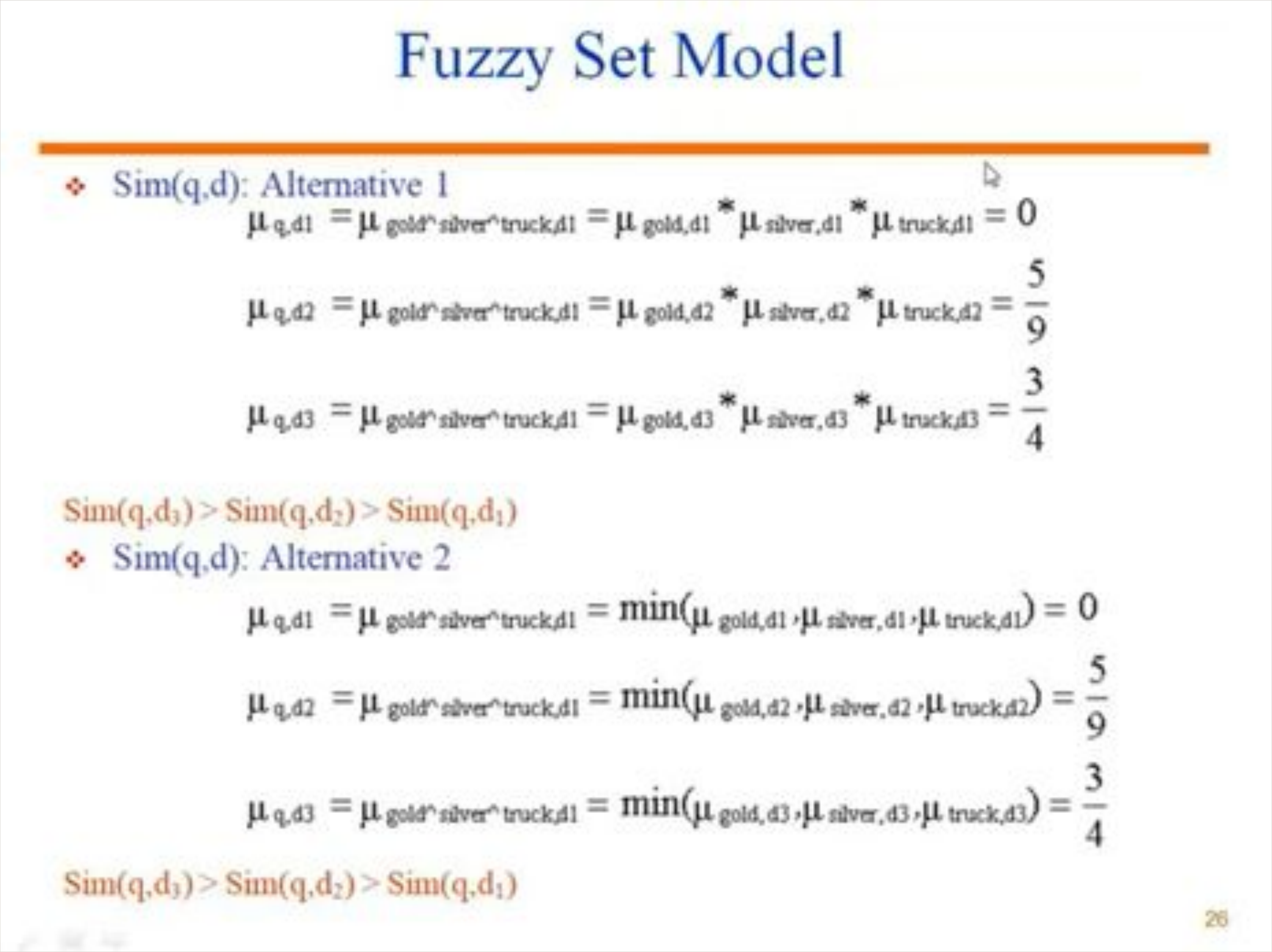
$$\mu_{\text{silver,d2}} = 1$$

$$\mu_{\text{truck,d2}} = 1$$

$$\mu_{\text{gold,d3}} = 1$$

$$\mu_{\text{silver,d3}} = 1 - 1 * 1 * \frac{1}{2} * \frac{1}{2} = \frac{3}{4}$$

$$\mu_{\text{truck,d3}} = 1$$



Fuzzy Set Model

- Sim(q,d): Alternative 1

$$\mu_{q,d1} = \mu_{gold\wedge silver\wedge truck,d1} = \mu_{gold,d1} * \mu_{silver,d1} * \mu_{truck,d1} = 0$$

$$\mu_{q,d2} = \mu_{gold\wedge silver\wedge truck,d1} = \mu_{gold,d2} * \mu_{silver,d2} * \mu_{truck,d2} = \frac{5}{9}$$

$$\mu_{q,d3} = \mu_{gold\wedge silver\wedge truck,d1} = \mu_{gold,d3} * \mu_{silver,d3} * \mu_{truck,d3} = \frac{3}{4}$$

$\mathrm{Sim}(q,d_3) > \mathrm{Sim}(q,d_2) > \mathrm{Sim}(q,d_1)$

- Sim(q,d): Alternative 2

$$\mu_{q,d1} = \mu_{gold\wedge silver\wedge truck,d1} = \min(\mu_{gold,d1}, \mu_{silver,d1}, \mu_{truck,d1}) = 0$$

$$\mu_{q,d2} = \mu_{gold\wedge silver\wedge truck,d1} = \min(\mu_{gold,d2}, \mu_{silver,d2}, \mu_{truck,d2}) = \frac{5}{9}$$

$$\mu_{q,d3} = \mu_{gold\wedge silver\wedge truck,d1} = \min(\mu_{gold,d3}, \mu_{silver,d3}, \mu_{truck,d3}) = \frac{3}{4}$$

$\mathrm{Sim}(q,d_3) > \mathrm{Sim}(q,d_2) > \mathrm{Sim}(q,d_1)$

EXTENDED BOOLEAN RETRIEVAL MODEL

A REALISTIC EXAMPLE

- Consider *N* = 1 million documents.
- Number of distinct terms, T=500,000
- Suppose we create term document incidence matrix

Total number of cells in matrix M = 500,000*10,00,000

=0.5 * 10^12 = approx 500GB

It will require lot of space in memory for execution which is infeasible

BOOLEAN RETRIEVAL MODEL ISSUE: CAN'T BUILD THE MATRIX

- In addition matrix M will have half-a-trillion 0's and 1's.
- Matrix will be extremely sparse.

What's a better representation?

- Solution:
  - We only record the 1 positions.
  - This idea is central to the first major concept in information retrieval, **the inverted index.**

INVERTED INDEX

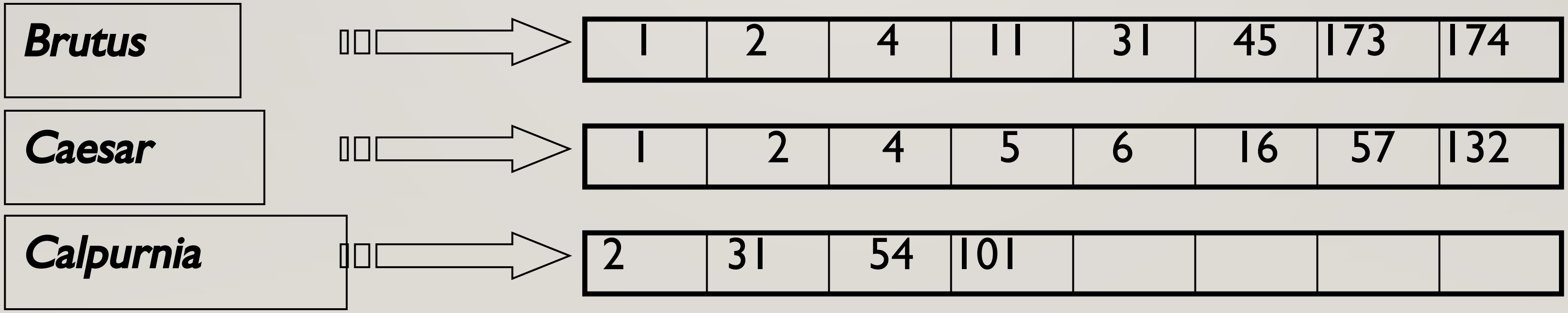
- Here we maintain a dictionary of each term (also known as lexicon)
- For each term *t*, we store a list of all documents that contain *t* known as postings.
- Each document is identified by its document id.
- Each term has its own posting list.

INVERTED INDEX EXAMPLE

Documents

Terms

	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



Brutus	1	2	4	11	31	45	173	174
Caesar	1	2	4	5	6	16	57	132
Calpurnia	2	31	54	101				

INVERTED INDEX EXAMPLE

How to maintain these posting list in memory

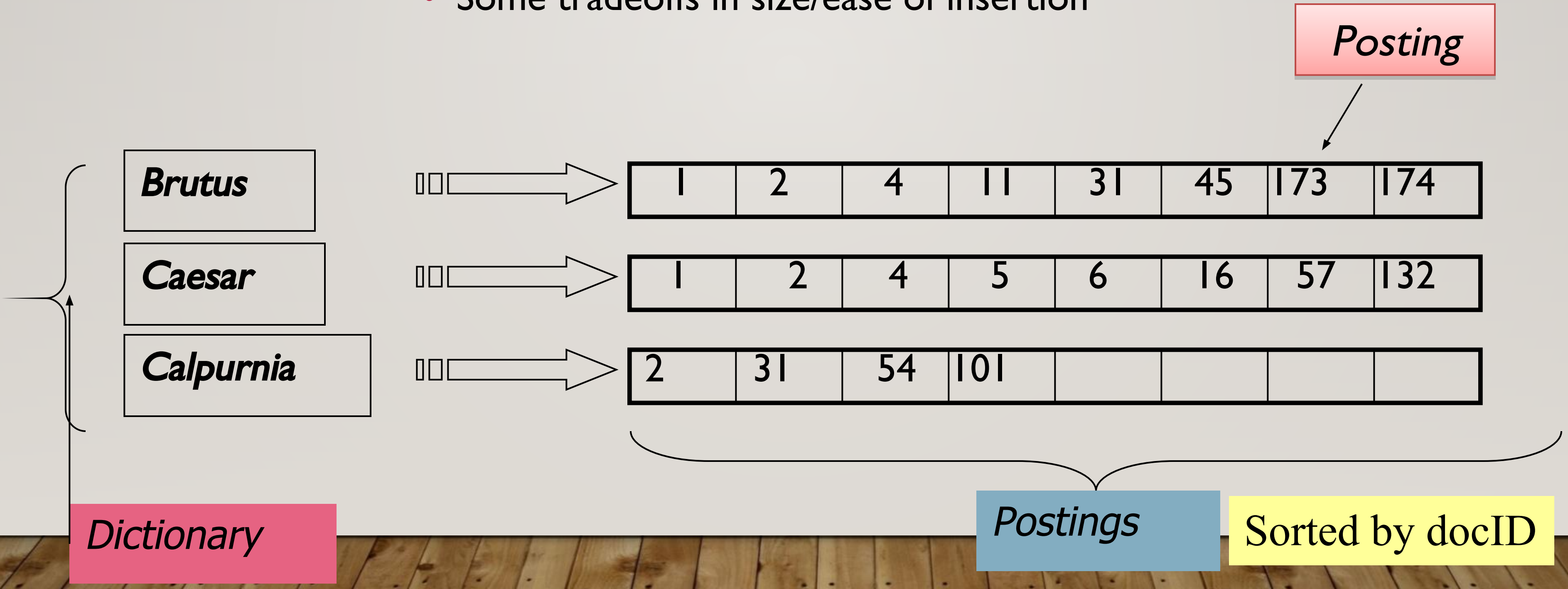
- **Fixed size array** : waste the storage space if unfilled.
- **Linked List**: require additional pointers
- **Variable size array:** insertion is difficult.

Linked list are preferred in case of dynamic insertions.

To search fast variable size arrays are preferred.

INVERTED INDEX

- We need variable-size postings lists
  - In memory, can use linked lists or variable length arrays
    - Some tradeoffs in size/ease of insertion



Posting

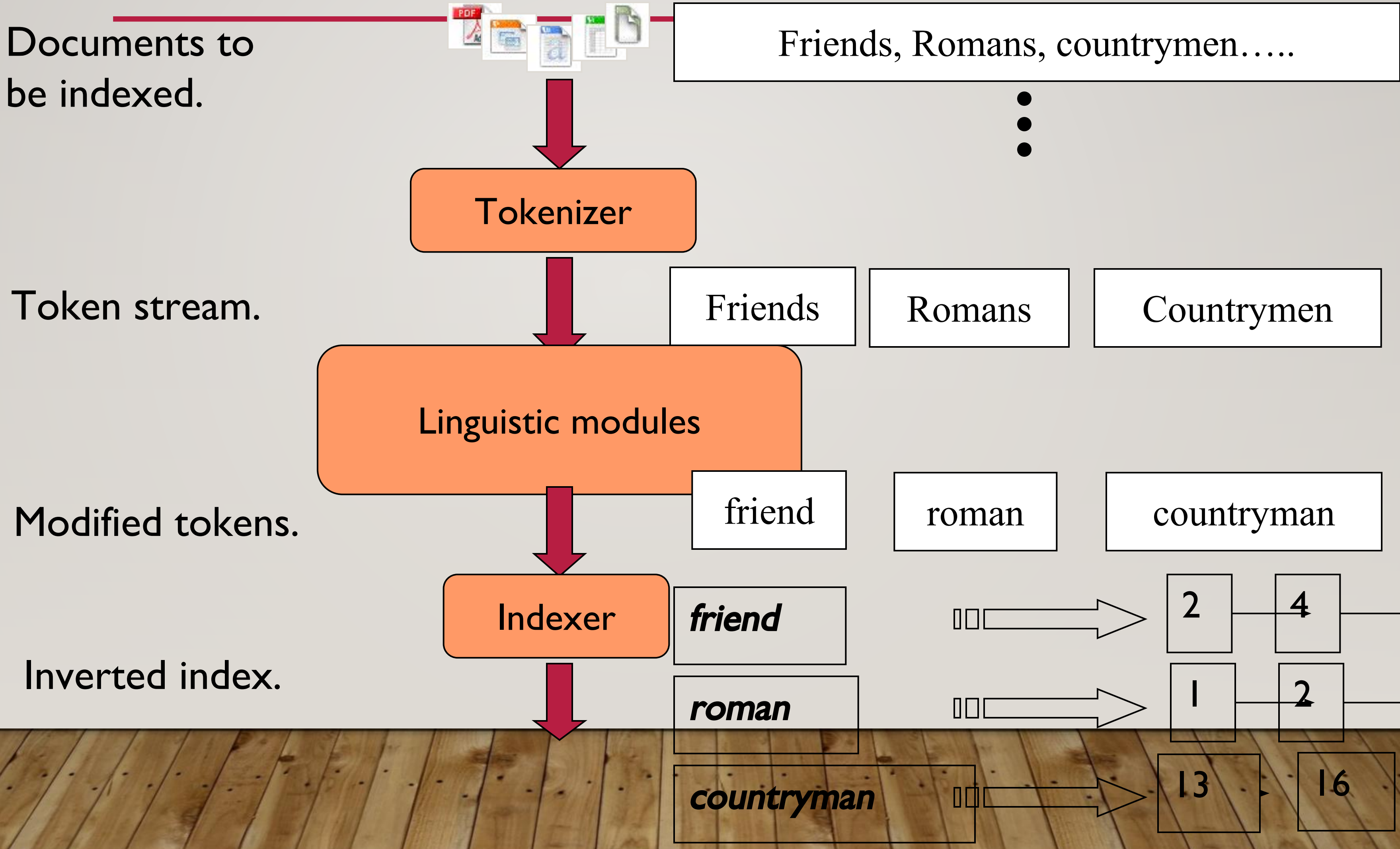
Brutus	1	2	4	11	31	45	173	174
Caesar	1	2	4	5	6	16	57	132
Calpurnia	2	31	54	101				

Dictionary

Postings

Sorted by docID

INVERTED INDEX CONSTRUCTION



Documents to be indexed.

Friends, Romans, countrymen…..

Tokenizer

Token stream.

Friends | Romans | Countrymen

Linguistic modules

Modified tokens.

friend | roman | countryman

Indexer

Inverted index.

friend → 2 → 4

roman → 1 → 2

countryman → 13 → 16

INDEXER STEPS: TOKEN SEQUENCE

- Sequence of (Modified token, Document ID) pairs.

Doc 1

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

Doc 2

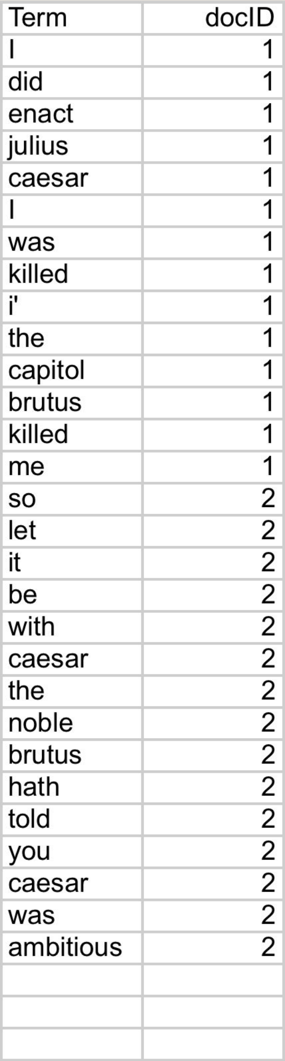
So let it be with
Caesar. The noble
Brutus hath told you
Caesar was ambitious

Term	docID
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

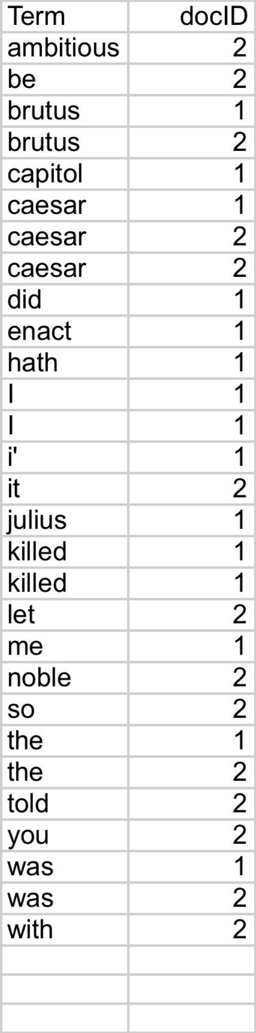
INDEXER STEPS: SORT

- Sort by terms
  - And then docID

Core indexing step



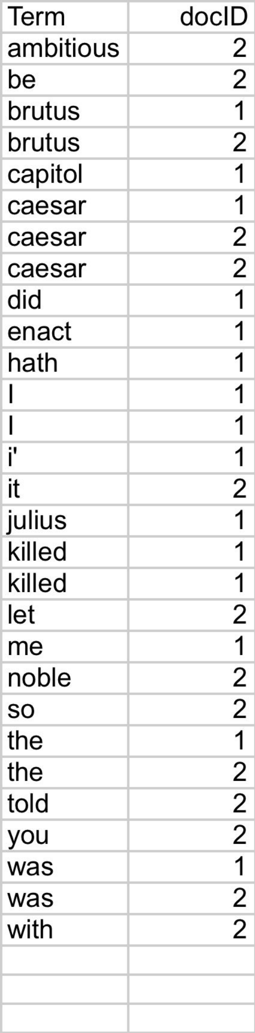
Term	docID
I	1
did	1
enact	1
julius	1
caesar	1
I	1
was	1
killed	1
i'	1
the	1
capitol	1
brutus	1
killed	1
me	1
so	2
let	2
it	2
be	2
with	2
caesar	2
the	2
noble	2
brutus	2
hath	2
told	2
you	2
caesar	2
was	2
ambitious	2



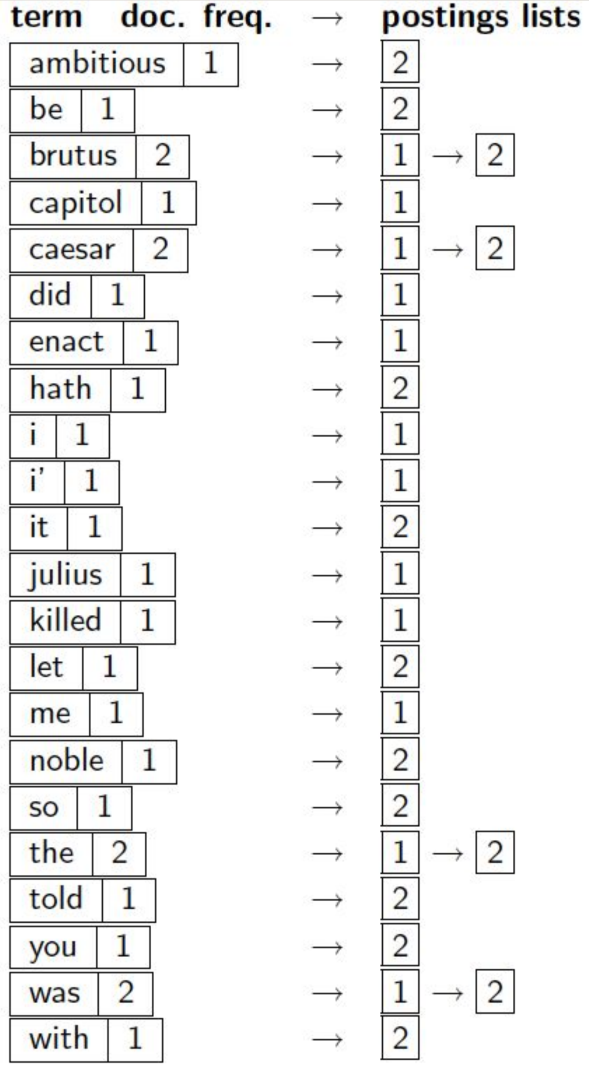
Term	docID
ambitious	2
be	2
brutus	1
brutus	2
capitol	1
caesar	1
caesar	2
caesar	2
did	1
enact	1
hath	1
I	1
I	1
i'	1
it	2
julius	1
killed	1
killed	1
let	2
me	1
noble	2
so	2
the	1
the	2
told	2
you	2
was	1
was	2
with	2

INDEXER STEPS: DICTIONARY & POSTINGS

- Multiple term entries in a single document are merged.
- Split into Dictionary and Postings
- Doc. frequency information is added.

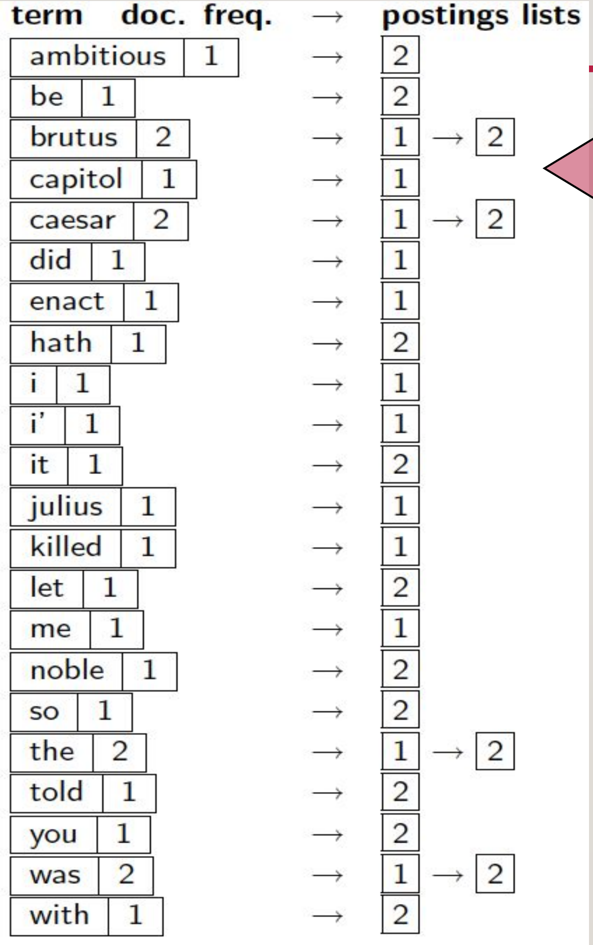


Term	docID
ambitious	2
be	2
brutus	1
brutus	2
capitol	1
caesar	1
caesar	2
caesar	2
did	1
enact	1
hath	1
I	1
I	1
i'	1
it	2
julius	1
killed	1
killed	1
let	2
me	1
noble	2
so	2
the	1
the	2
told	2
you	2
was	1
was	2
with	2



term	doc. freq.	→	postings lists
ambitious	1	→	2
be	1	→	2
brutus	2	→	1 → 2
capitol	1	→	1
caesar	2	→	1 → 2
did	1	→	1
enact	1	→	1
hath	1	→	2
i	1	→	1
i'	1	→	1
it	1	→	2
julius	1	→	1
killed	1	→	1
let	1	→	2
me	1	→	1
noble	1	→	2
so	1	→	2
the	2	→	1 → 2
told	1	→	2
you	1	→	2
was	2	→	1 → 2
with	1	→	2

WHERE DO WE PAY IN STORAGE?



term	doc. freq.	→	postings lists
ambitious	1	→	2
be	1	→	2
brutus	2	→	1 → 2
capitol	1	→	1
caesar	2	→	1 → 2
did	1	→	1
enact	1	→	1
hath	1	→	2
i	1	→	1
i'	1	→	1
it	1	→	2
julius	1	→	1
killed	1	→	1
let	1	→	2
me	1	→	1
noble	1	→	2
so	1	→	2
the	2	→	1 → 2
told	1	→	2
you	1	→	2
was	2	→	1 → 2
with	1	→	2

Lists of docIDs

Terms and counts

Pointers

PRACTICE QUESTION

Question:: Consider these documents:

Doc 1 breakthrough drug for schizophrenia
Doc 2 new schizophrenia drug
Doc 3 new approach for treatment of schizophrenia
Doc 4 new hopes for schizophrenia patients

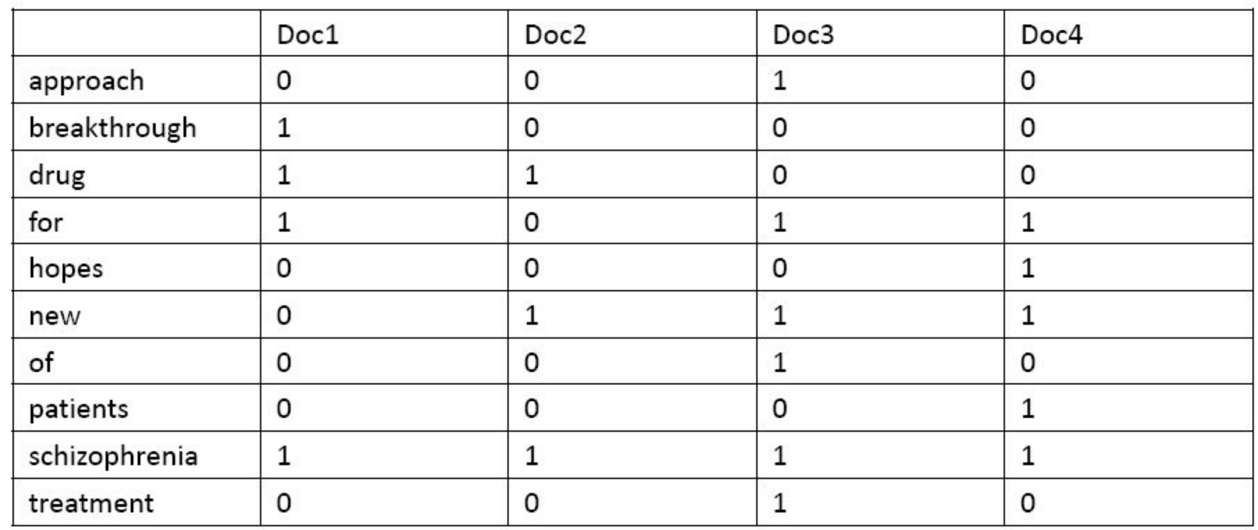
a. Draw the term‑document incidence matrix for this document collection.

b. Draw the inverted index representation for this collection,

c. Answer the query:

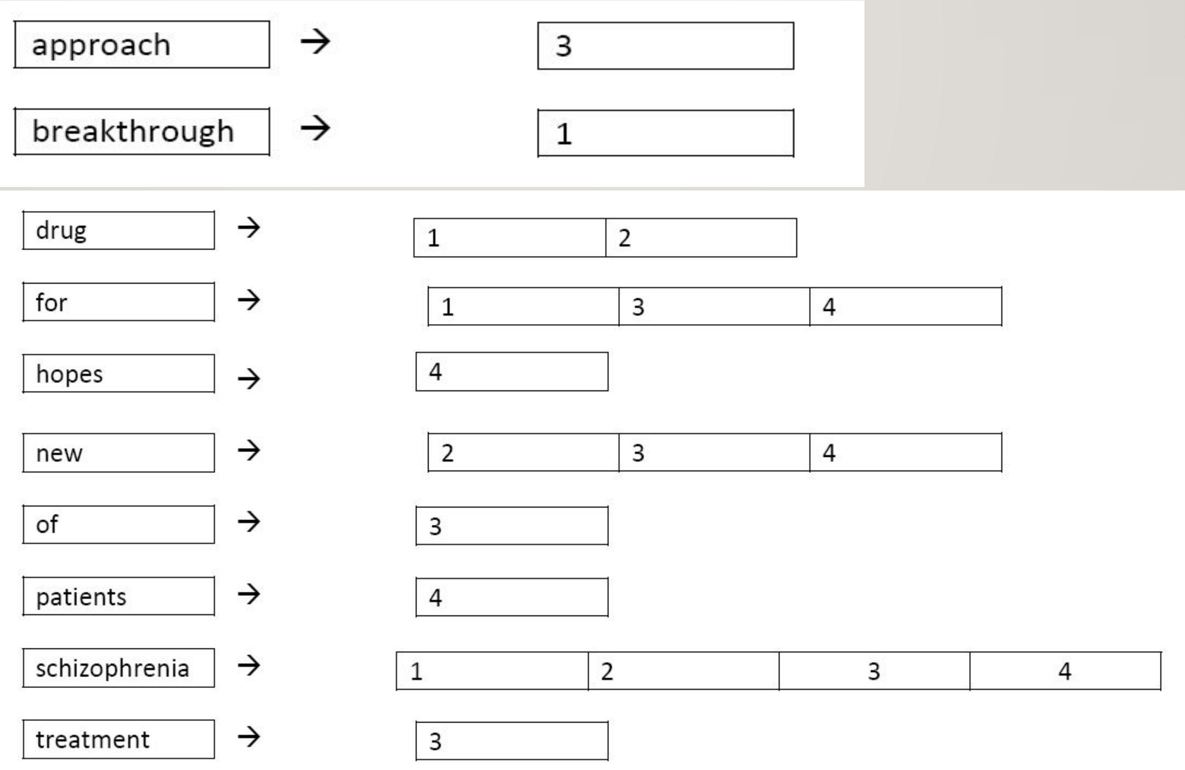
- a. schizophrenia AND drug
- b. for AND NOT(drug OR approach)

TERM DOCUMENT MATRIX



	Doc1	Doc2	Doc3	Doc4
approach	0	0	1	0
breakthrough	1	0	0	0
drug	1	1	0	0
for	1	0	1	1
hopes	0	0	0	1
new	0	1	1	1
of	0	0	1	0
patients	0	0	0	1
schizophrenia	1	1	1	1
treatment	0	0	1	0

INVERTED INDEX



approach	→	3			
breakthrough	→	1			
drug	→	1	2		
for	→	1	3	4	
hopes	→	4			
new	→	2	3	4	
of	→	3			
patients	→	4			
schizophrenia	→	1	2	3	4
treatment	→	3			

C. Query answers

a. schizophrenia AND drug

Doc1 and Doc 2

b. for AND NOT(drug OR approach)

Doc4

PROCESSING THE BOOLEAN QUERIES

- How do we process a query using an inverted index and basic Boolean retrieval model?

REFERENCES

- Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, "An introduction to Information Retrieval", 2013 Cambridge University Press UP.
- Stefan Büttcher, Charles L. A. Clarke and Gordon V. Cormack, Information Retrieval, 2010, MIT Press.