MANCHESTER 1824
The University of Manchester

COMP34711
Week 3

Indexing Documents: Inverted Index

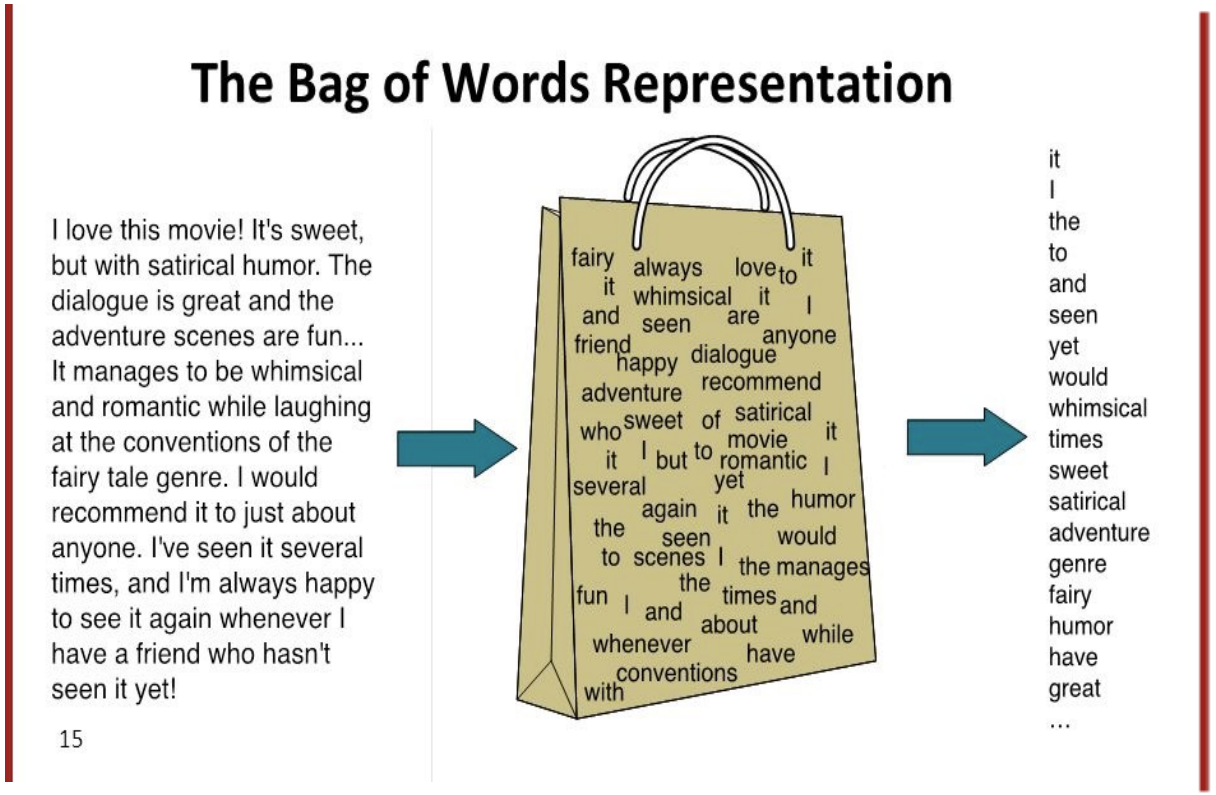
Goran Nenadic

with inputs from J. McNaught and examples from the IIR book

How to find "good" index terms?

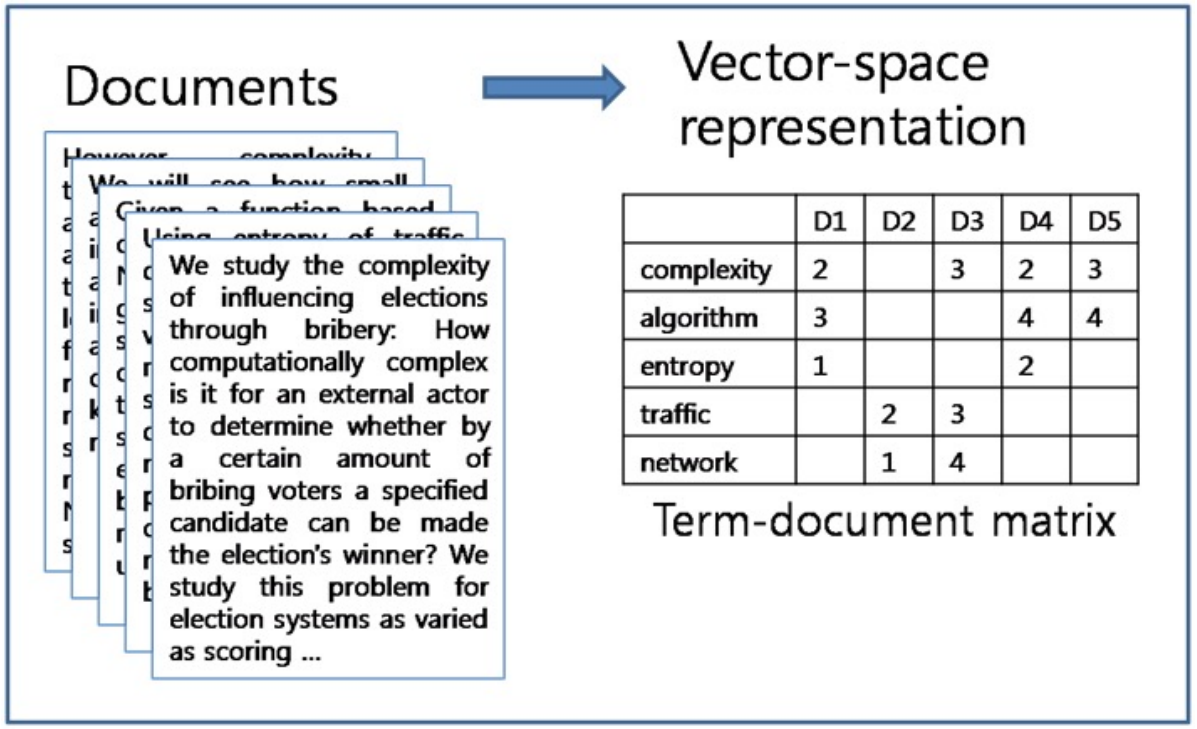
- **Index all words? Or some?**
  - Most frequent ones?
  - How about so called stop-words (the, and, of, it, …)?
- **Do we want to "rank" them?**
  - How do you decide on importance/relevance?
- Last week we looked at
  - some linguistic hypotheses and theories
    - Zipf's law
    - Luhn's hypothesis
  - BoW and vector representation

BoW representation



- BoW is used in many IR systems

Term-document matrix



	D1	D2	D3	D4	D5
complexity	2		3	2	3
algorithm	3			4	4
entropy	1			2	
traffic		2	3		
network		1	4		

Term-document **frequency** matrix

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	121	11	0	0	0	1
Brutus	23	123	0	1	0	0
Caesar	47	407	0	1	1	1
Calpurnia	0	10	0	0	0	0
Cleopatra	17	0	0	0	0	0
mercy	16	0	1	1	1	1
worser	10	0	1	1	1	0

(IIR book)

Term-document **incidence** matrix

	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

1 if document contains word,
0 otherwise

(IIR book)

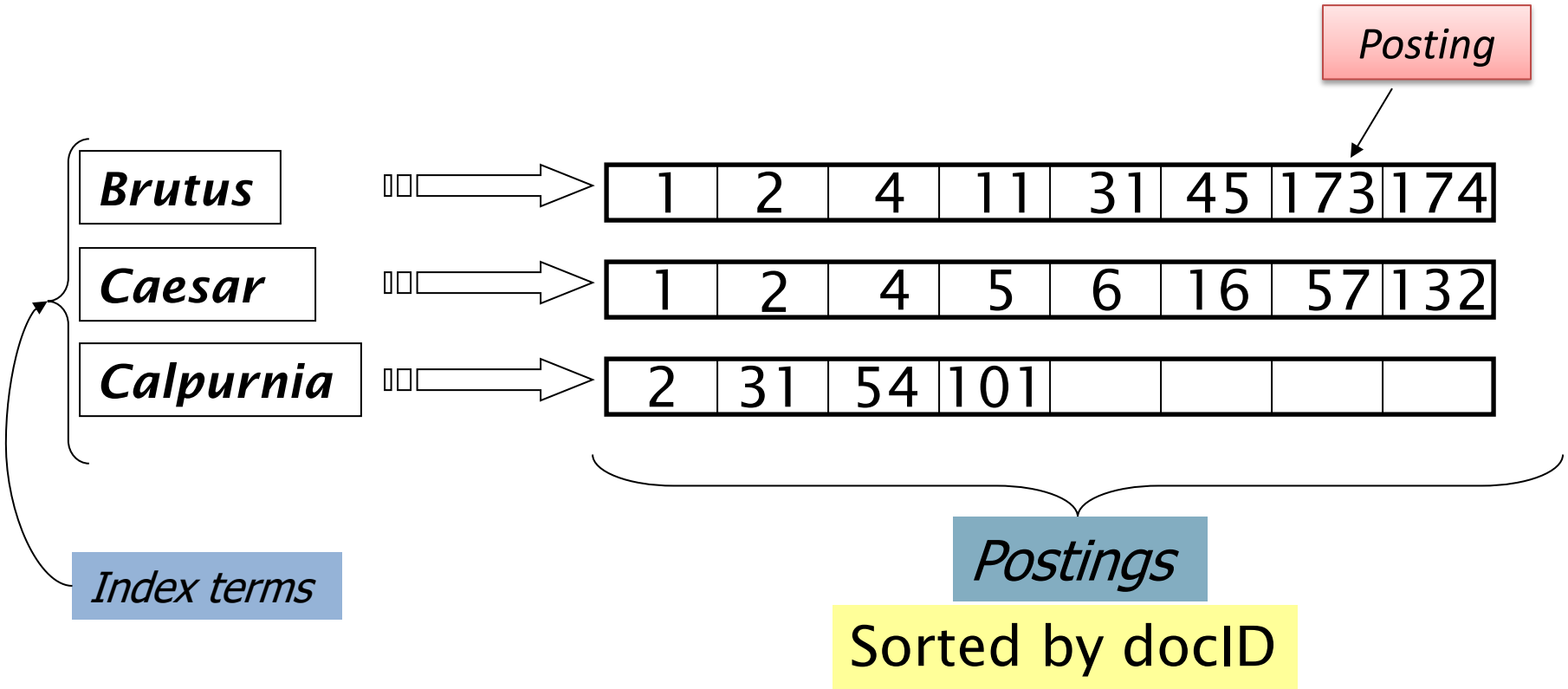
Note: massive matrices

- A small collection has 1 million documents
  - With ~1000 words per document
  - Assume 500,000 unique words in collection
- Matrix: 500,000 * 1,000,000 = 500,000,000,000 (half a trillion)
- But it's a sparse matrix! No more than one billion non-zeros

  → **Just store the 1s**

Inverted index

- For each index term *t*, we store a list of all documents (**docID**) that contain *t*



(IIR book)

Indexer steps: (1) Token sequence

- Sequence of (Token, Document ID) pairs.

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	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

(IIR book)

Indexer steps: (2) Sort

- Sort first by terms
- Then sort by 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

(IIR book)

Indexer steps: (3) Dictionary & Postings

- Multiple term entries in a single document are merged
- Split into Dictionary and Postings
- Document frequency (for each term) is also added

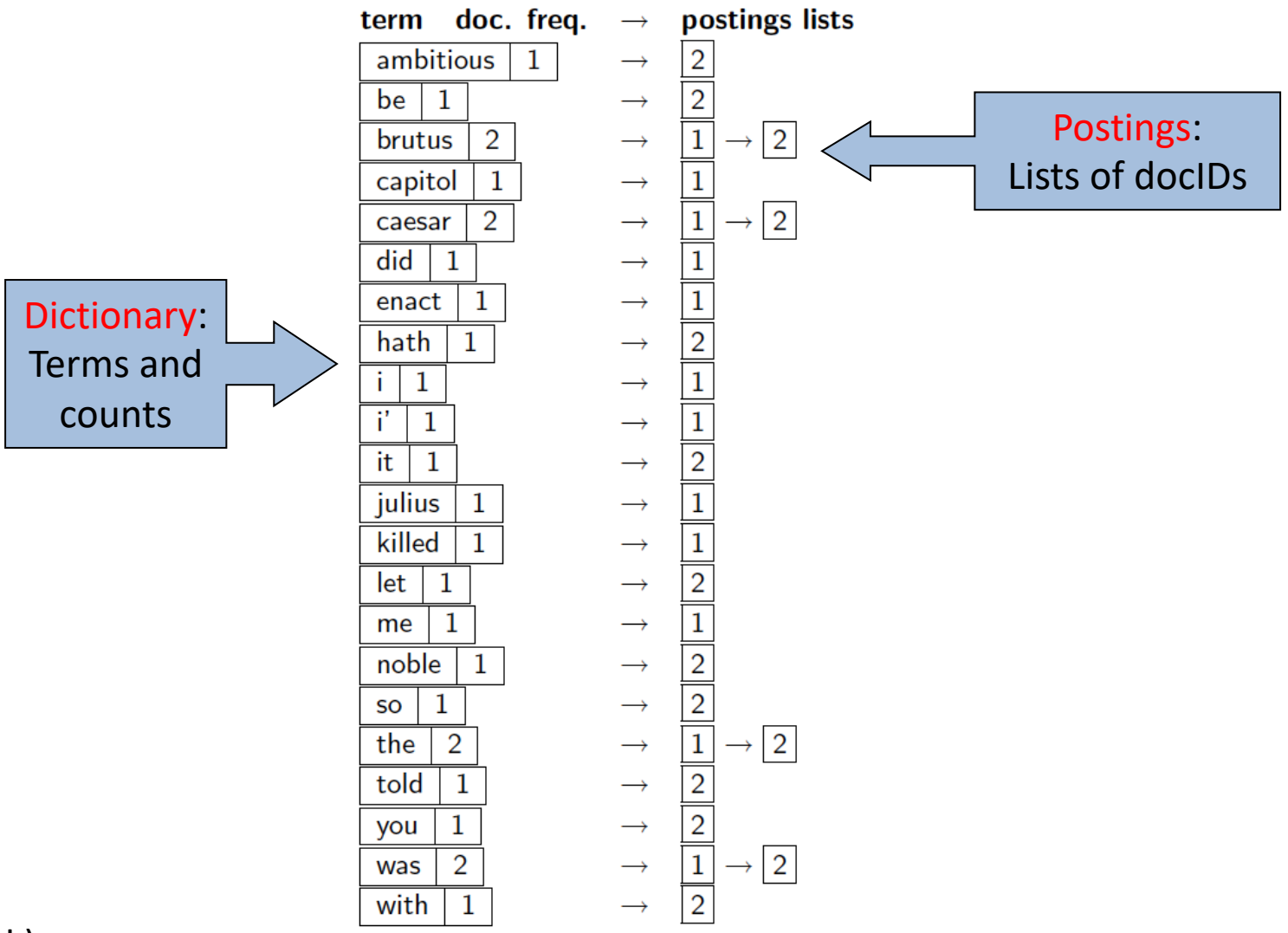
Why doc. frequency?
Will discuss later.

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

(IIR book)

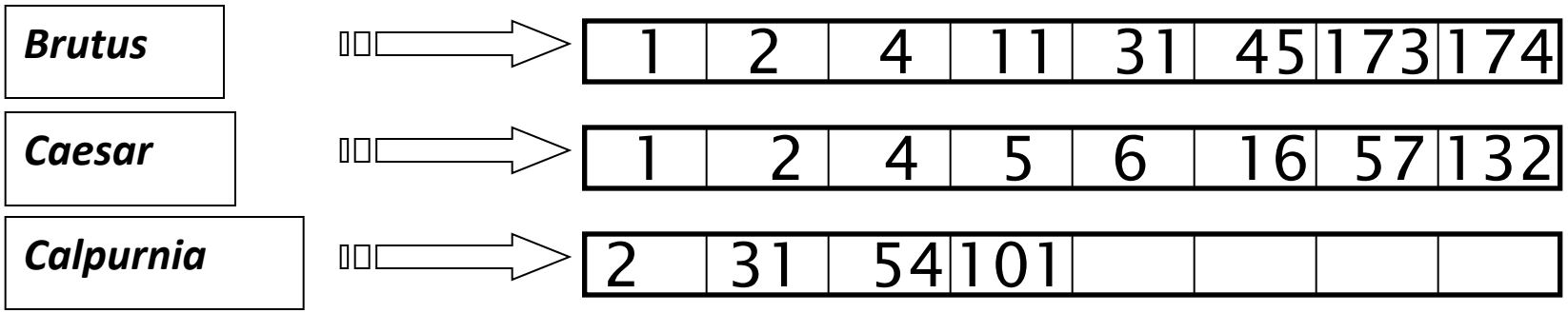
Indexer steps: (3) Dictionary & Postings



(IIR book)

Inverted index

- We need variable-size postings lists
  - On disk, a continuous run of postings is normal and best
  - In memory, can use linked lists or variable length arrays
    - Some tradeoffs in size/ease of insertion



(IIR book)

Inverted index construction

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

terms

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

sort

X 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

dict.

posting

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

(IIR book)