Boolean Retrieval.

- The query is expressed as a Boolean expression
  - terms are the operands.
  - standard Boolean operators AND, OR, NOT.

Corpus: collection of plays written by Shakespeare.

Document: an individual play.

~~Query~~ Query: Boolean expression having terms connected by logical operators.

- Which plays of Shakespeare contain words Brutus AND Caesar but NOT Calpurnia

Design of Information Retrieval Systems.

Components/items related to the design of IR systems:

- Documents (doc Id)
- Corpus (collection of documents)
- User has information need
  - ↳ query string
- "Term" ← basic unit of information
- Relevance

algorithmic challenges.

Known world — Documents

Unknown world — query.

relevant?

Brute force:

grep Brutus & Caesar

↓

result → all lines that do not have Calpurnia.

Challenges:

- text volume is too large.

Brutus AND Caesar | NOT Calpurnia.

↓ one time confirmation

↓ more exhaustive

"near"
words Roman / Romans
near countrymen

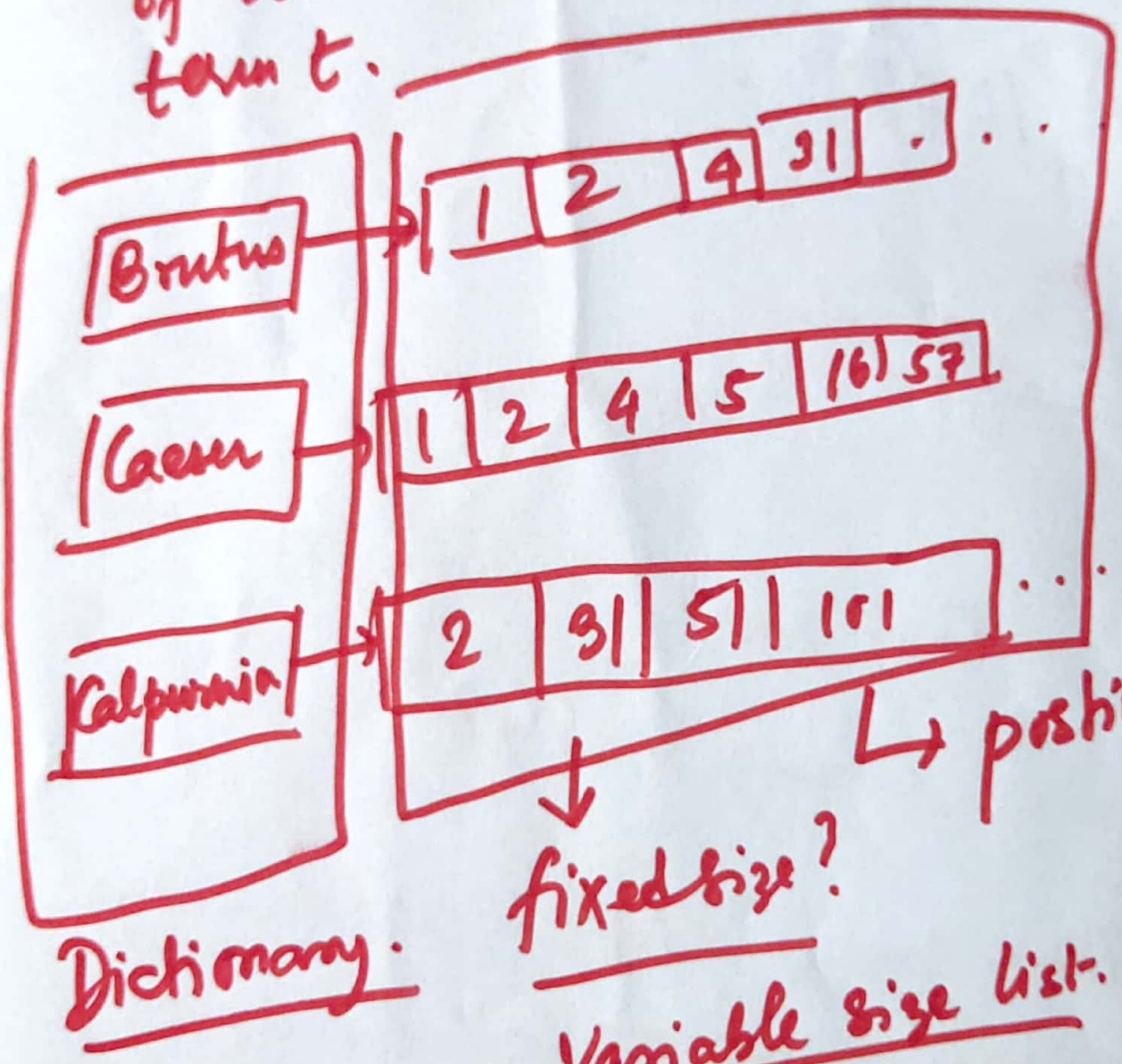
Term-document incidence matrix

	AdC	Je	TTemp	Hamlet	Othello	Mac
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

$1000 \times 1M \leq 1B$.

Inverted Index.

For each term t we store a list of all documents that contain the term t.



Brutus : 110100 AND

Caesar : 110111 AND

Calpurnia: 101111

⇒ 100100. ←

N = 1M docs.

↳ 1000 words.

Each word → 6 bytes. (on avg)

↳ ~ 6GB.

There are M = 500K unique terms across all the docs.

500K × 1M ~ $\frac{1}{2}$ trillion.

Howmany 1's → 1 billion