

An Information Retrieval Example: Boolean Queries

COMP3009J: Information Retrieval

Dr. David Lillis (david.lillis@ucd.ie)

UCD School of Computer Science
Beijing Dublin International College

Example: the Boolean style query

- William Shakespeare was a famous English playwright. Suppose we want to search for characters in his plays.
- Which plays of Shakespeare contain the words **Brutus** AND **Caesar** but NOT **Calpurnia**?
- As a first attempt, we could read the plays line-by-line to find those that contain **Brutus** and **Caesar** and then remove those that contain **Calpurnia**.



Example: the Boolean style query

- “... we could read the plays line-by-line to find those that contain **Brutus** and **Caesar** and then remove those that contain **Calpurnia**.”

A **corpus** is a collection of documents. The plural is **corpora**.

- Why is that not the answer?
 - Slow (for large corpora)
 - Other operations (e.g., find the word **Romans** near **countrymen**) not feasible.
 - Ranked retrieval not possible (best documents shown first)
 - Later lectures!



Example: The Boolean style query

Term-document incidence matrix

- Instead, we store the information we need in some sort of data structure. Here is a **term-document incidence matrix** showing some of the words contained in some of Shakespeare's plays.

		Plays (Documents)					
		Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Words (Terms)	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		
	Cleopatra	1	0	0	0		
	mercy	1	0	1	1		
	worser	1	0	1	1		

1 if the play contains the word, 0 otherwise.

Example: The Boolean style query Incidence vectors

- So we have an **incidence vector** for each word.
- It consists of 1s (for the plays it appears in) and 0s (for those it does not appear in).
- e.g.
 - **Brutus:** 110100
 - **Caesar:** 110111
 - **Calpurnia:** 010000

	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

Example: The Boolean style query Operators

- Our query is: **Brutus AND Caesar NOT Calpurnia**
- To get “**NOT Calpurnia**”, we get the **complement** of the incidence vector for **Calpurnia** (using the bitwise NOT operator, which changes all 1s to 0s and all 0s to 1s).
 - $001000 \rightarrow 110111$
- Now we can use the bitwise AND operator to combine our three vectors:
 - $110100 \text{ AND } 110111 \text{ AND } 101111 = 100100$
- Referring back to the term-document incidence matrix, we see that the answer is: **Antony and Cleopatra, Hamlet**

Bigger collections

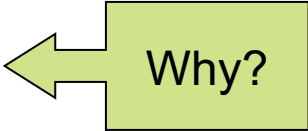
- This approach can be effective, but how well does it **scale** to larger collections?
 - Consider $N = 1,000,000$ documents, each with about 1,000 words.
 - Average 6 bytes per word including spaces and punctuation.
 - 6GB of data in the documents.
 - Say there are $M = 500,000$ *distinct* terms among these.

Can't build the matrix

- 500,000 x 1,000,000 matrix has **half-a-trillion** 0s and 1s.

- But it has no more than one **billion** 1's.

- matrix is extremely **sparse**.



Why?

- What's a better representation?

- We only record the 1 positions, and not the 0s.

- We will look at this in a later lecture.

Questions

- Please post questions in the **Brightspace discussion forum** for COMP3009J.