A REPORT

ON

CS F469 Information Retrieval: Assignment 1

by

Name of the student	ID number
Dev Gala	2021A7PS0182H
Atharva Dashora	2021A7PS0127H
Sricharan Reddy Bollampalli	2021A7PS0379H
Abhishek Kali Madiki	2021AAPS0550H

BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
(Hyderabad Campus)

February, 2024

ACKNOWLEDGEMENT

Abstract

This

CONTENTS

List of Tables

List of Figures

1 Introduction

The ability to effectively extract relevant data from the vast pool of digital information (documents, web pages, images, etc) has gained significant importance in the field of Computer Science. Information Retrieval (IR) systems play a crucial in fetching and organizing vast amounts of data.

With the IT revolution post 1970, the amount of information on the internet has increased many fold (Fig:1)[Dua]. This lead to the rise of many IR systems like Google's Search Engine, Yahoo!, DuckDuckGo etc. , have become an integral part of everyone's daily life. With their ever growing presence and importance, it is important to understand how IR Systems work and what algorithms are employed by them to ensure correctness and relevance of the documents they retrieve.

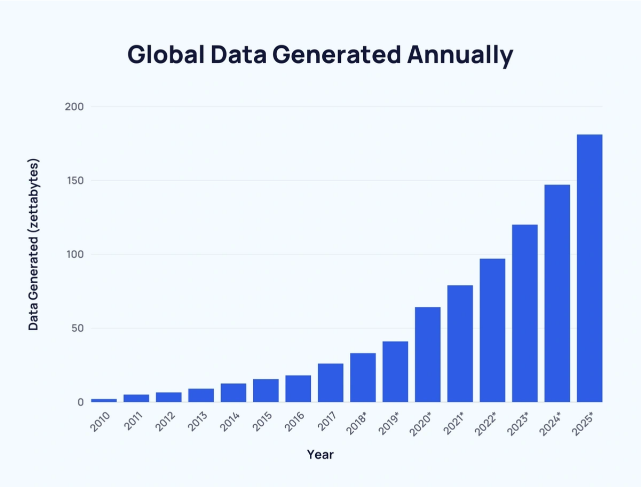


Figure 1: Global Data Generation

This assignment aims to further our understanding of the simplest form of information retrieval called Boolean Retrieval (2.1). This assignment also helps us understand wildcard queries and how to use retrieval algorithms using them. We conduct 5 experiments to understand the implementation of boolean retrieval and wildcard retrieval; note down results and draw conclusions about performance of various retrieval methods based on time taken and memory consumption.

2 Concepts

2.1 Boolean Retrieval

Boolean retrieval is a classic information retrieval (IR) model that focuses on retrieving documents based on the logical relationships between keywords present in a user's query. It relies on Boolean logic and the following basic operators:

- **AND:** Documents must contain all keywords connected by the AND operator for them to be retrieved.
- **OR:** Documents will be retrieved if they contain at least one of the keywords connected by the OR operator.
- **NOT:** Documents containing the keyword following the NOT operator will be excluded from the results.

Boolean retrieval pre-processes the corpus by indexing the words in the corpus. This involves creating an index with words as the key and the document the occur in as the value. From here many data structures such as Inverted Index (2.2) and Tries (2.3)/B-Trees are constructed to facilitate effective retrieval of documents. The query provided by the user is broken down into keywords and documents are retrieved for each of them. Then the corresponding set operations are applied to the each set of documents according to the given boolean operators in the query.(See Figure 2 [Rod])

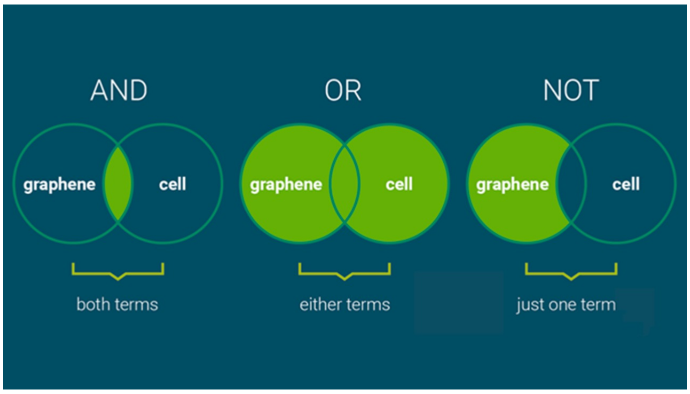
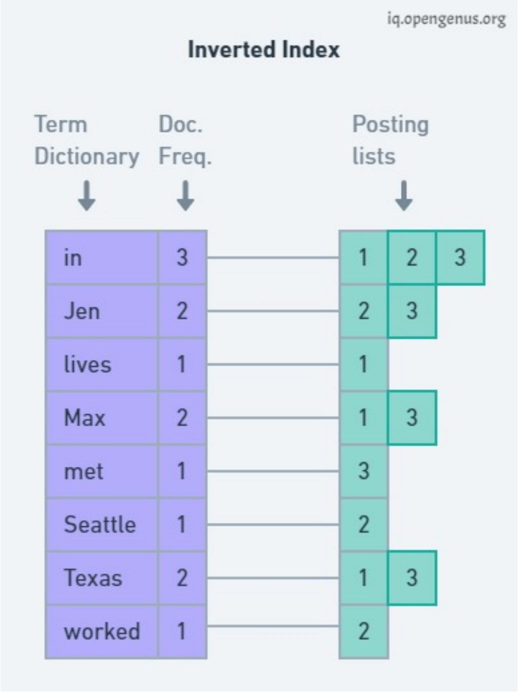


Figure 2: Boolean Query

Boolean queries, because of their simplistic nature require very less pre-processing on the query side of the system, it has its limitations. One of them being that the queries need to be very structured i.e they need be in a proper boolean expression form in order to get correct documents. This is useful for certain use cases like legal information retrieval (systems like WestLaw) but boolean retrieval might not be useful for layman as it cannot accept natural language.

2.2 Inverted Index

Information retrieval systems can obtain documents or web pages containing a certain phrase or combination of terms more quickly by using an Inverted Index data structure. An inverted index lists papers or web pages that contain each term, which is arranged in the index according to terms. Search engines, database systems, and other applications that demand efficient text search frequently use inverse indexes. They are especially helpful in cases of big document collections, where it would be impractically slow to search through every document.
Inverted index is constructed by using each word (called term) as the key and a list of all the documents that contain said term (See Figure 3 [Sen]). This index can be implemented using HashTables, Tries 2.3, etc depending on the requirements on the system designer.



Term Dictionary	Doc. Freq.	Posting lists
in	3	1 2 3
Jen	2	2 3
lives	1	1
Max	2	1 3
met	1	3
Seattle	1	2
Texas	2	1 3
worked	1	2

Figure 3: Inverted Index

2.3 Tries

3 Apparatus

4 Methodology

4.1 Experiment 1

4.2 Experiment 2

4.3 Experiment 3

4.4 Experiment 4

4.5 Experiment 5

5 Results and Observations

5.1 Experiment 1

5.2 Experiment 2

5.3 Experiment 3

5.4 Experiment 4

5.5 Experiment 5

6 Conclusion

7 References

References

[Dua] Fabio Duarte. *Amount of Data Created Daily (2024)*. URL: https://explodingtopics.com/blog/data-generated-per-day.

[Rod] Stanley Rodnik. *AThe Power of Boolean Search: Unleashing the Potential of Precise Information Retrieval*. URL: https://www.linkedin.com/pulse/power-boolean-search-unleashing-potential-precise-retrieval-rodnik/.

[Sen] Nilanjan Sengupta. *Indexing in Natural Language Processing for Information Retrieval*. URL: https://www.analyticsvidhya.com/blog/2021/07/indexing-in-natural-language-processing-for-information-retrieval/.

8 Glossary

Document	A unit of information that can be stored and retrieved
Corpus	Collection of Documents in an IR system
Query	A string defining the information the user would like to retrieve
Intent	Message that is passed between components such as activities, content providers, broadcast receivers, services etc.
WebView	Android View to render web pages withing an android activity.
XML	eXtensible Markup Language.
U.I.	User Interface
SnackBar	Snackbars provide brief messages about app processes at the bottom of the screen.
HTTP	HyperText Transfer Protocol
Asynchronous Process	Process that runs in parallel with main process
Cursor	Middleware between SQLite database connection and SQL query
Context	Allows access to application-specific resources and classes, as well as up-calls for application-level operations such as launching activities, broadcasting and receiving intents, etc.
No-SQL databases	Databases that do not follow relational database pattern.

Table 1: Glossary