Boolean Retrieval

$1^{st}StutiChaturvedi$
202161006

$2^{nd}HaripriyaGoswami$
202161003

$3^{rd}YashitaVajpayee$
202162012

***Abstract*—To understand Boolean Retrieval and perform weight Zone scoring on a dataset**

1. Introduction

Boolean retrieval is a long established model of information retrieval in which queries are framed in the form of a Boolean expression of terms, in which operators NOT,AND,OR are combined with terms.The model looks each document as set of words. Boolean retrieval is the exact match model in which we get the set of exact documents corresponding to the query..
Basic Assumption of Boolean Model
1. An index term is either present(1) or absent(0) in the document
2. All index terms provide equal evidence with respect to information needs.
3. Queries are defined as Boolean combinations of index terms.
-A AND B: depict documents that contains both A and B
-A OR B: depict documents that contains either A or B
-NOT A: depict the documents that do not contain A

	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	
...							

Query:- $(Antony) \wedge (Brutus) \wedge (Caesar)$
Q:$\{D1, D2, D6\} \cap$ {D1,D2,D4} $\cap$ {D1,D2,D4,D5,D6}
:{D1,D2}
hence document1 and document2 is retrieved

```
INTERSECT(p1, p2)
 1  answer ← ⟨ ⟩
 2  while p1 ≠ NIL and p2 ≠ NIL
 3  do if docID(p1) = docID(p2)
 4        then ADD(answer, docID(p1))
 5             p1 ← next(p1)
 6             p2 ← next(p2)
 7        else if docID(p1) < docID(p2)
 8                then p1 ← next(p1)
 9                else p2 ← next(p2)
10  return answer
```

1.1. Mean Average Precision

The mean average precision just referred to as Arithmetic precision is used to measure the performance of models doing document/information retrival and object detection tasks.

$$\text{MAP} = \frac{\sum_{q=1}^{Q} \text{AveP(q)}}{Q}$$

1.2. Precision and Recall

Precision
To calculate precision take the ratio of the number of relevant and retrieved documents

$$Precision = \frac{relevant\ item\ retrieved}{retrieved\ item}$$

Recall
To calculate Recall take ratio of the number of retrieved and relevant documents.

$$Recall = \frac{relevant\ item\ retrieved}{relevant\ item}$$

1.3. Weighted Zone Scoring

Boolean query q and document d is given , a weighted zone Scoring assigns scores at intervals [0,1] to pairs (q1, q2) by calculating a linear combination of zone evaluations,and boolean value is assigned to each zone in the document.

```
ZONESCORE(q1, q2)
 1  float scores[N] = [0]
 2  constant g[ℓ]
 3  p1 ← postings(q1)
 4  p2 ← postings(q2)
 5  // scores[] is an array with a score entry for each document, initialized to zero.
 6  // p1 and p2 are initialized to point to the beginning of their respective postings.
 7  // Assume g[] is initialized to the respective zone weights.
 8  while p1 ≠ NIL and p2 ≠ NIL
 9  do if docID(p1) = docID(p2)
10        then scores[docID(p1)] ← WEIGHTEDZONE(p1, p2, g)
11             p1 ← next(p1)
12             p2 ← next(p2)
13        else if docID(p1) < docID(p2)
14                then p1 ← next(p1)
15                else p2 ← next(p2)
16  return scores
```

2. Approach and Implementation

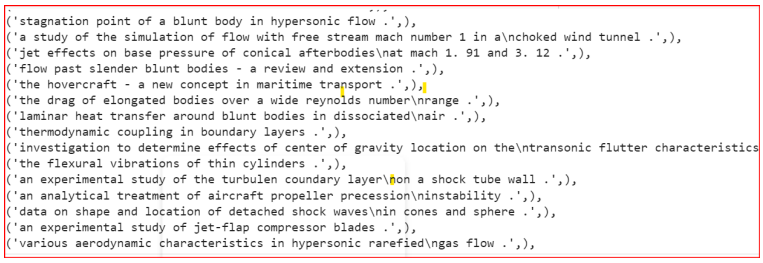
A Term-Document Incidence matrix is created. In which terms are indexed and if a document contains a specific term then it's entry is 1 else it's entry is 0.

2.1. Processing Boolean Queries

we have taken a example in introduction part the steps of processing the query is given below
1.Locate Antony in the Dictionary
2.Retrieve its postings
3.Locate Brutus in the Dictionary
4.Retrieve its postings
5.Locate Caesar in the Dictionary
6.Retrieve its postings
In code we have used merged file created through BSBI and we created the term incidence matrix of it.Then we perform Boolean Retrieval on a term incidence matrix

3. Output

Printing documents after performing boolean retrieval



4. Conclusion

Boolean retrieval is the most exact-match model.Many users, particularly experts, would rather choose Boolean query models. Boolean queries are precise: A document either matches with query or not. This provides the user's control and transparency over what is retrieved.it is easy to implement. A Boolean model keep mark on term presence or absence, but often we would like to group evidence, giving more weightage to documents that have a term several times as opposed to ones that appear it only once. For performing this we need term frequency (the count of term occurs in a document) in postings lists. Boolean queries retrieve a set of matching documents ,but we require effective method to order the returned results. This requires having a procedure for finding a document score which encapsulates how fine a document matches for a query Further reference codes can be found at https://github.com/ir4h/LAB-CS653-2021.

References

[1] Christopher D Manning ,Hinrich Schutze and Prabhakar Raghavan *The Introduction to Information Retrieval* .