The Boolean Retrieval Model & Extended Boolean Models

Boolean queries : Exact match

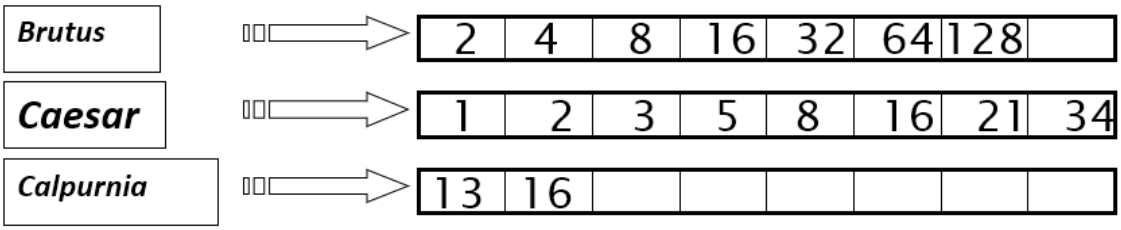
- **boolean retrieval model** → being able to ask a query that is a boolean expression
  - using **AND**, **OR** and **NOT** to join query terms
    - views each documents as a set of words. 즉, 단어는 문서는 단어로 구성됨 → boolean expression이 가능한 이유
    - is precise : document matches condition or not.
  - the simplest model to build an IR system on
  - 현재 잘 사용하는 모델은 아님, 하지만 IR system의 근본 모델. (60대 ~ 90년대)
- ex : WestLaw : 아메리카 합중국의 법조와 법무 담당자를 위한 온라인 법률 조사와 데이터 베이스 열람 서비스

Boolean queries : More general merges

- Exercise : adapt the merge for the queries:
  - Brutus AND NOT Caesar
  - Brutus OR NOT Caesar

Query optimization

- What is the best order for query processing? → process in order of increasing freq. In other words, start with smallest set
  - AND 연산의 경우 document가 적은 단어를 먼저 처리하는 것이 시간 속도 개선에 효율적
  - 미리 dictionary에 document frequency를 저장하는 이유
- Consider a query that is an AND of n terms.
- For each of the n terms, get its postings, then AND them together.
- example : **Brutus** AND **Calpurnia** AND **Caesar**



More general optimization

- example : (**madding** or **crowd**) AND (**ignoble** OR **strife**)

1. get doc. freq.'s for all terms
2. or의 두 피연산자 단어 posting을 합병시킨후, estimate the size of each OR by the sum of its doc. freq.'s (conservative)
3. process in increasing order of OR sizes

출처 : 스탠포드 IR 강의 (https://www.youtube.com/watch?v=QVVvx_Csd2I&list=PLaZQkZp6WhWwoDuD6pQCmgVyDbUWl_ZUi&index=6)