키워드 검색에 대한 RDBMS 에 기반을 둔 효율적인 역색인 기법

신윤미*, 전민혁*, 안진현**, 임동혁***
*호서대학교 컴퓨터공학과
**제주대학교 경영정보학과
***호서대학교 컴퓨터정보공학부
e-mail : dhim@hoseo.edu

An Efficient Inverted Index Technique based on RDBMS for Keyword Search

Yoonmi Shin*, Minhyuk Jeon*, Jinhyun Ahn**,Dong-Hyuk Im***
*Department of Computer Engineering, Hoseo University
**Department of Management Information Systems, Jeju National University
***Division of Computer and Information Engineering, Hoseo University

요 약

RDBMS 상에서 문서에 포함된 키워드 검색을 위한 질의 시 병합 조인 방식을 통해 키워드 검색을 시도하게 된다. 그러나 대용량의 문서를 저장하고 있는 RDBMS 내에서 병합 조인을 사용 시 검색 키워드에 대해 불필요한 비교 연산으로 인하여 질의 문에 대한 검색시간이 길어질 수 있다. 본 논문은 행 지향 관계형 역 색인을 이용하여 키워드 검색 질의 시 병합 조인의 단점을 보완한 지그재그 병합 조인 알고리즘을 사용한다. 관계형 데이터베이스인 postgreSQL 에서 프로시저로 불필요한 비교 연산을 최소화한 지그재그 병합 조인 알고리즘을 구현하여 키워드 검색에 대한 질의 속도 향상을 확인하였다.

1. 서론

빅데이터 활용의 증가로 대용량 데이터 처리에 대한 요구가 급증하고 있다. 그에 따른 대용량 데이터에 대한 정보 검색은 주로 관계형 데이터베이스 내에서 복잡한 질의 문의 조합과 다중 검색을 이용해야 한다. 이러한 질의에는 상당한 시간이 소모될 수 있어 검색시스템은 색인을 이용하여 유효성을 높여 사용한다. 보편적으로 사용되는 색인 기반의 키워드 검색은 병합 조인을 사용하여 대다수의 경우에서 유효한 결과를 내고있다. 하지만 역 색인에서 병합 조인 사용시 불필요한 비교 연산 때문에 질의 문에 대한 검색 시간이 증가하게 된다.

따라서 본 논문에서는 관계형 데이터베이스 PostgreSQL 을 이용하여 병합 조인의 단점을 보완한 지그재그 병합 조인 알고리즘[1]을 프로시저로 구현하였다. 기존 merge join 또한 프로시저를 만들어 데이터 사이즈 증가에 따른 쿼리 속도 비교를 하였다.

2. 구현 및 실험

1) 데이터 저장 방식

본 논문에서는 행 지향 관계형 역 색인 테이블을 사용한다. 역 색인은 정보 검색 시 많이 사용되고 검색 작업 속도 향상에 도움이 된다. 또한 대용량 문서를 효율적으로 저장이 가능하다. 역 색인 테이블의 구성은 다음 <표 1>과 같다.

<표 1> 역색인 테이블 구성 요소

Table name	Column name
Main Table	term, docid, df, tf, doclen
Offset Table	term, docid, offset

Main table 은 term 에 해당하는 문서 번호를 나타내는 docid, 해당 단어가 모든 문서에서 나온 횟수를 나타내는 df, 단어가 해당 문서에 나온 횟수를 나타내는 tf, term 이 속한 문서의 길이를 나타내는 doceln 칼럼으로 구성된다. offset table 은 구문 쿼리에서 사용하기 위한 테이블로서 term 이 문서내의 위치를 나타내는 offset 칼럼을 포함하고 있다.

역 색인 테이블의 데이터는 Westbury Lab[2]의 USENET 약 1700 만개의 텍스트로 구성된 문서 중 6 만개의 문서를 사용했다. 각 문서에 대해 자연어처리인 불 용어(stop word) 처리, 어간 추출(stemming) 작업을 통해 역 색인 데이터를 생성하였다.

2) 결합 쿼리

결합 쿼리란 검색한 n 개의 키워드를 포함한 문서를 찾는 쿼리이다. 예를 들어 'hoseo' ,' university' 두개의 키워드를 검색했을 때 둘다 포함한 문서가 있는지 검색한다. 결합 쿼리에서는 RDBMS 의 기존 병합 조인을 변경한 지그재그 병합 조인[1]을 사용함에 있어 <term, docid>를 클러스터한 Btree index 를 이용한다. 기존 병합조인은 문서 id 가 일치하지 않을 때 작은 문서 id 를 가리키는 커서가 앞으로 한 칸씩 이동하여 비교한다. 하지만 지그재그 병합 조인 알고리즘은 비교 값보다 큰 문서 id 또는 같은 문서 id 로 건너뛰어 불필요한 비교 연산을 수행하지 않고, 이러한 건너뛰는 과정에서 갤럽 검색[3]을 사용한다. 갤럽 검색은 커서의 이동이 n*2+1 만큼씩 이동하여 더 큰 값을 검색하게 될 경우 건너뛴 간격 내에서 이진 검색을 통해 값을 찾게 된다.

결합쿼리의 성능 실험은 6 만개 문서와 100 만개의 문서 데이터를 비교하여 진행했다. 우선 문서 내에서 2 개에서 5 개 사이의 검색 키워드를 랜덤으로 뽑아 하나의 셋으로 지정한 뒤, 키워드셋 100 개 각각의 쿼리 결과를 평균 내었다. 그에 따른 (그림 1)과 (그림 2)은 각각 6 만개의 문서, 100 만개의 문서에서 100 개의 키워드셋의 쿼리 평균 실행 시간을 나타낸다.

성능 실험 결과에 따르면 3 개 이상의 키워드가 결합된 키워드셋에서 지그재그 병합조인이 비교적 더 좋은 성능을 보이고, 6 만개 문서에서보다 100 만개 문서에서의 쿼리 속도 향상을 보인다.

(그림 1) 6 만개 문서에서의
병합조인과 지그재그 병합 조인 속도 비교

(그림 2) 100 만개 문서에서의
병합조인과 지그재그 병합 조인 속도 비교

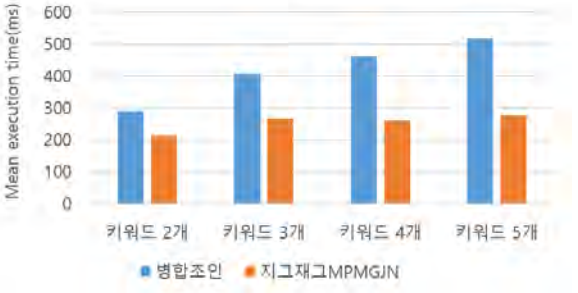
3) 구문 쿼리

구문 쿼리는 결합 쿼리에서 확장된 형태이다. 구문 쿼리란 n 개의 키워드가 연달아 있는 문서를 찾는 쿼리를 의미한다. 문서 id 목록이 일치하면 해당 검색 키워드에 대한 오프셋을 확인하여 검색 키워드가 해당 위치에 있는지 확인이 필요하다. [4]에서 제한된 multi-predicate merge join algorithm(MPMGJN)을 지그재그 병합 조인에 추가하여 사용한다. 지그재그 MPMGJN 에서는 지그재그 병합 조인에서처럼 병합 연산자가 문서 id 간의 불일치가 발생하면 seek() 함수를 통해 다른 문서 id 에서 더 큰 문서 id 를 검색하고 같은 문서 id 에서 지그재그 병합 조인의 논리를 적용하여 오프셋 목록을 병합한다. 지그재그 MPMGJN 은 문서 id 가 교차하지 않을 때 다수의 오프셋 튜플 비교를 건너 뛰어 행지향 관계형 역 색인의 문제점인 <term, docid> 쌍의 비교 반복 횟수를 줄였다.

구문 쿼리의 성능 실험은 100 개의 검색 키워드를 랜덤으로 선택한다. 그리고 선택한 키워드 기준으로 뒤에 이어지는 1 개에서 4 개 사이의 키워드와 같이 묶어 하나의 키워드셋을 만들었다. 이렇게 최대 5 개의 키워드로 구성된 키워드셋에 대한 구문 검색 쿼리 실행 결과를 평균 낸 결과가 (그림 3)와 (그림 4)이다. 각각의 그래프는 6 만개 문서와 100 만개 문서에 대한 구문 검색 시 기존의 병합 조인과 지그재그 MPMGJN 의 쿼리 실행 시간을 나타내고 있다.

성능 실험 결과에 따르면 지그재그 MPMGJN 이 키워드가 많아질수록 병합조인 사용시보다 질의 속도가 향상되며 6 만개의 문서를 이용할 때 보다 100 만개의 문서에서 키워드 검색 쿼리에서 비교적 빠른 속도가 나옴을 알 수 있다.

(그림 3) 6 만개 문서에서의
병합조인과 지그재그 MPMGJN 속도 비교



(그림 4) 100 만개 문서에서의
병합조인과 지그재그 MPMGJN 속도 비교

4. 결론 및 향후 계획

본 논문에서는 병합 조인과 지그재그 병합 조인을 이용하여 6 만개 문서와 100 만개 문서에서의 쿼리 속도를 비교하였다. 결합쿼리 및 구문쿼리 실험에서 키워드 수가 증가하고 문서의 양이 많아질수록 기존 병합 조인보다 지그재그 병합 조인의 이용 시 쿼리 속도가 더욱 향상되었다. 따라서 지그재그 병합 조인은 대용량 문서에서 키워드 검색 시 병합 조인보다 좋은 쿼리 성능을 기대할 수 있다.

또한 지그재그 MPMGJN 알고리즘을 이용해 대용량 문서가 저장된 RDBMS 상에서 키워드 검색 시 불필요한 문서 ID 와 오프셋의 비교를 갤럽 검색을 통해 건너뛰었다. 그에따라 행 지향 관계형 역 색인의 문제인 <term, docid>의 반복이 최소화되었고, 키워드 검색 쿼리 속도가 증가한 것을 확인 하였다.

향후 계획은 문서를 문단으로 나누어 저장 할 수 있는 문단 컬럼을 추가해 구문 쿼리 시 불필요한 오프셋 행를 건너뛰어 보다 빠른 속도 향상을 기대할 수 있다. 또한 파티션 기법을 활용해 검색 공간의 절약 방법을 모색 할 것이다.

Acknowledgement

이 논문은 2017 년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구(No.NRF-2017R1C1B1003600)이며 정보통신기술진흥센터의 지원을 받아 수행된 연구임(No.R0113-15-0005, 대규모 트랜잭션 처리와 실시간 복합 분석을 통합한 일체형 데이터 엔지니어링 기술 개발). 또한, 이 논문은 2018 년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. NRF-2018R1D1A1B07048380).

참고문헌

[1] I. Rae, A. Halverson, J. Naughton: In-RDBMS Inverted Indexes revisited. ICDE 2014: 352-363

[2] C. Shaoul and C. Westbury. (2011). A USENET corpus (2005–2010), Edmonton, AB: University of Alberta, [Online].Available:http://www.psych.ualberta.ca/~westburylab/downloads/usenetcorpus.download.html.

[3] J. L. Bentley and A. C.-C. Yao, “An almost optimal algorithm for unbounded searching”, Information processing letters, vol. 5, no. 3,pp. 82–87, 1976

[4] C. Zhang, J. Naughton, D. DeWitt, et al., “On Supporting Containment Queries in Relational Database Management Systems”, in SIGMOD, 2001, pp. 425–436.