(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2012년06월21일
(11) 등록번호 10-1157693
(24) 등록일자 2012년06월12일

(51) 국제특허분류(Int. Cl.)
G06F 17/30 (2006.01) *G06F 17/00* (2006.01)
(21) 출원번호 10-2007-7005777
(22) 출원일자(국제) 2005년08월08일
심사청구일자 2010년08월06일
(85) 번역문제출일자 2007년03월13일
(65) 공개번호 10-2007-0049664
(43) 공개일자 2007년05월11일
(86) 국제출원번호 PCT/US2005/028192
(87) 국제공개번호 WO 2006/020595
국제공개일자 2006년02월23일
(30) 우선권주장
10/917,746 2004년08월13일 미국(US)
(56) 선행기술조사문헌
JP2000137730 A
(뒷면에 계속)

(73) 특허권자
구글 인코포레이티드
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이 1600 (우:94043)
(72) 발명자
딘, 제프리, 아드게이트
미국 94303 캘리포니아 팔로 알토 스톡톤 플레이스 3179
하아르, 폴, 지.
미국 94114 캘리포니아 샌프란시스코 22 스트리트 4222
(뒷면에 계속)
(74) 대리인
남상선

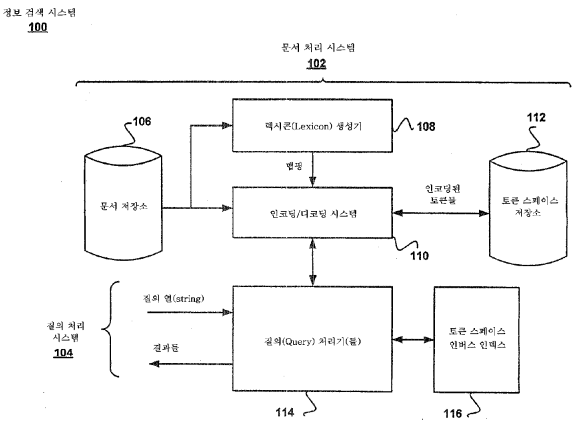
전체 청구항 수 : 총 21 항 심사관 : 석상문

(54) 발명의 명칭 **토큰스페이스 저장소와 함께 사용하기 위한 멀티-스테이지질의 처리 시스템 및 방법**

(57) 요 약

멀티-스테이지 질의 처리 시스템 및 방법은 멀티-계층 맵핑 수단에 의해 촉진되는 증분적인 문서 재구성을 통해, "스니피트(snippet)" 생성을 포함하는 멀티-스테이지 질의 스코어링을 가능하게 한다. 멀티-스테이지 질의 처리 시스템의 하나 이상의 스테이지들에서, 관련도 스코어들의 세트는 사용자에게 정렬된 리스트로서 나타내기 위해 문서들의 서브세트를 선택하는데 사용된다. 관련도 스코어들의 세트는 멀티-스테이지 질의 처리 시스템의 이전 스테이지들에서 결정된 관련도 스코어들의 하나 이상의 세트들로부터 부분적으로 유도될 수 있다. 몇몇 실시예들에서, 멀티-스테이지 질의 처리 시스템은 사용자 질의에 1회 이상의 패스들을 실행할 수 있고, 정렬된 리스트에서 문서들의 관련도를 개선하도록 후속 패스에서 사용하기 위해 사용자 질의를 확장시키도록 각 패스로부터 정보를 이용할 수 있다.

대 표 도



(72) 발명자

서시노글루, 올칸

미국 94040 캘리포니아 마운틴 뷰 웨스트 엘 카미노 릴 2400아파트먼트 716

싱갈, 아미타브, 케이.

미국 94303 캘리포니아 팔로 알토 아지텍 웨이 2435

(56) 선행기술조사문헌

JP2003242170 A

KR1019970076328 A

KR100295354 B1

JP2003242170 A*

KR100295354 B1*

KR1019970076328 A

*는 심사관에 의하여 인용된 문헌

특허청구의 범위

청구항 1

멀티-스테이지 질의 처리 시스템에서 질의(query)를 처리하는 방법으로서,

상기 멀티-스테이지 질의 처리 시스템은 하나 이상의 프로세서들 및 상기 방법을 수행하기 위한 상기 하나 이상의 프로세서들에 의한 실행을 위한 하나 이상의 프로그램들을 저장하는 메모리를 포함하고,

상기 방법은:

질의의 제 1 스테이지 처리를 수행하는 단계 - 상기 질의의 제 1 스테이지 처리를 수행하는 단계는:

하나 이상의 질의 용어(query term)들에 응답하여 인덱스(index)로부터 제 1 세트의 문서 식별자들을 검색(retrieve)하는 단계;

질의 용어들의 존재, 용어 빈도, 및 문서 대중도 중 하나 이상에 기초하여 상기 제 1 세트의 문서 식별자들의 적어도 하나의 서브세트에 대응하는 제 1 세트의 압축된 문서들에 대해 제 1 세트의 관련도 스코어들(relevancy scores)을 생성하여, 상기 메모리에 상기 제 1 세트의 관련도 스코어들을 저장하는 단계

을 포함함 -;

상기 질의의 제 2 스테이지 처리를 수행하는 단계 - 상기 질의의 제 2 스테이지 처리를 수행하는 단계는:

토큰 위치들의 리스트, 상기 문서들에서의 질의 용어들 간의 거리들, 상기 문서들에서의 토큰들의 속성들, 및 상기 제 1 세트의 압축된 문서들 중 한 문서에 사용된 질의 용어 주변에 나타나는 텍스트 중 하나 이상에 기초하여 상기 제 1 세트의 압축된 문서들에서의 문서들에 대한 제 2 세트의 관련도 스코어들을 생성하여, 상기 메모리에 상기 제 2 세트의 관련도 스코어들을 저장하는 단계

를 포함함 -;

상기 메모리로부터 상기 제 1 세트의 관련도 스코어들 및 상기 제 2 세트의 관련도 스코어들을 판독하여, 상기 제 1 세트의 관련도 스코어들 및 상기 제 2 세트의 관련도 스코어들에 기초하여 추가 처리를 위한 문서들의 정렬된 리스트를 생성하는 단계;

상기 문서들의 정렬된 리스트에서의 문서들로부터 부가적인 질의 용어들을 자동으로 생성하는 단계;

상기 부가적인 질의 용어들을 이용하여 새로운 질의를 형성(formulate)하는 단계;

상기 인덱스로부터 제 2 세트의 문서 식별자들을 검색하고 상기 부가적인 질의 용어들에 적어도 부분적으로 기초하여 제 3 세트의 관련도 스코어들을 생성하도록, 상기 새로운 질의를 처리하는 단계; 및

사용자에게 프리젠테이션하기 위한 상위 문서들의 세트를 선택하기 위해 상기 제 3 세트의 관련도 스코어들을 사용하는 단계

를 포함하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 2

제 1 항에 있어서,

제 1 세트의 토큰들을 복원하기 위해 상기 제 1 세트의 압축된 문서들의 적어도 일부분을 압축해제하는 단계 - 상기 제 1 세트의 복원된 토큰들은 상기 제 1 세트의 압축된 문서들에서의 위치들과 연관됨 - ; 및

상기 제 1 세트의 복원된 토큰들을 이용하여 상기 제 1 세트의 압축된 문서들의 하나 이상의 부분들을 재구성함으로써, 상기 제 1 세트의 압축된 문서들에서의 문서들을 위한 스니피트(snippet)들을 생성하는 단계

를 더 포함하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 3

제 2 항에 있어서,

상기 상위 문서들의 세트의 정렬된 리스트로 상기 사용자에게 상기 스니피트들을 나타내는 단계를 더 포함하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 4

제 1 항에 있어서,

상기 제 3 세트의 관련도 스코어들은 상기 제 2 세트의 문서 식별자들에 대응하는 문서들에서의 상기 질의 용어들의 하나 이상의 위치들에 기초하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 5

제 1 항에 있어서,

상기 제 2 세트의 관련도 스코어들은 상기 문서들에서의 토큰들의 속성들에 기초하고, 상기 속성들은 상기 문서들에서의 토큰들의 폰트 속성들을 포함하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 6

제 1 항에 있어서,

상기 제 1 세트의 문서 식별자들을 검색하는 단계는 상기 하나 이상의 질의 용어들에 대한 토큰 위치들의 리스트를 생성하기 위해 상기 인덱스를 사용하는 단계 및 상기 토큰 위치들에 대응하는 문서 식별자들의 세트를 생성하기 위해 맵에 액세스하는 단계를 포함하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

멀티-스테이지 질의 처리 시스템으로서,

하나 이상의 프로세서들; 및

상기 하나 이상의 프로세서들에 의해 실행될 하나 이상의 프로그램들을 저장하는 메모리

를 포함하고,

상기 하나 이상의 프로그램들은:

질의의 제 1 스테이지 처리를 수행하기 위한 명령들 — 상기 질의의 제 1 스테이지 처리를 수행하기 위한 명

령들은:

하나 이상의 질의 용어들에 응답하여 인덱스로부터 제 1 세트의 문서 식별자들을 검색하고;

질의 용어들의 존재, 용어 빈도, 및 문서 대중도 중 하나 이상에 기초하여 상기 제 1 세트의 문서 식별자들의 적어도 하나의 서브세트에 대응하는 제 1 세트의 압축된 문서들에 대해 제 1 세트의 관련도 스코어들을 생성하여, 상기 메모리에 상기 제 1 세트의 관련도 스코어들을 저장하기 위한 명령들

을 포함함 -;

상기 질의의 제 2 스테이지 처리를 수행하기 위한 명령들 - 상기 질의의 제 2 스테이지 처리를 수행하기 위한 명령들은:

토큰 위치들의 리스트, 상기 문서들에서의 질의 용어들 간의 거리들, 상기 문서들에서의 토큰들의 속성들, 및 상기 제 1 세트의 압축된 문서들 중 한 문서에 사용된 질의 용어 주변에 나타나는 텍스트 중 하나 이상에 기초하여 상기 제 1 세트의 압축된 문서들에서의 문서들에 대한 제 2 세트의 관련도 스코어들을 생성하여, 상기 메모리에 상기 제 2 세트의 관련도 스코어들을 저장하기 위한 명령들

을 포함함 -;

상기 메모리로부터 상기 제 1 세트의 관련도 스코어들 및 상기 제 2 세트의 관련도 스코어들을 판독하여, 상기 제 1 세트의 관련도 스코어들 및 상기 제 2 세트의 관련도 스코어들에 기초하여 추가 처리를 위한 문서들의 정렬된 리스트를 생성하기 위한 명령들;

상기 문서들의 정렬된 리스트에서의 문서들로부터 부가적인 질의 용어들을 자동으로 생성하기 위한 명령들;

상기 부가적인 질의 용어들을 이용하여 새로운 질의를 형성하기 위한 명령들;

상기 인덱스로부터 제 2 세트의 문서 식별자들을 검색하고 상기 부가적인 질의 용어들에 적어도 부분적으로 기초하여 제 3 세트의 관련도 스코어들을 생성하도록, 상기 새로운 질의를 처리하기 위한 명령들; 및

사용자에게 프리젠테이션하기 위한 상위 문서들의 세트를 선택하기 위해 상기 제 3 세트의 관련도 스코어들을 사용하기 위한 명령들

을 포함하는,

멀티-스테이지 질의 처리 시스템.

청구항 12

제 11 항에 있어서,

상기 하나 이상의 프로그램들은:

제 1 세트의 토큰들을 복원하기 위해 상기 제 1 세트의 압축된 문서들의 적어도 일부분을 압축해제하고 - 상기 제 1 세트의 복원된 토큰들은 상기 제 1 세트의 압축된 문서들에서의 위치들과 연관됨 - ; 그리고

상기 제 1 세트의 복원된 토큰들을 이용하여 상기 제 1 세트의 압축된 문서들의 하나 이상의 부분들을 재구성함으로써, 상기 제 1 세트의 압축된 문서들에서의 문서들을 위한 스니피트들을 생성하기 위한

명령들을 더 포함하는,

멀티-스테이지 질의 처리 시스템.

청구항 13

제 12 항에 있어서,

상기 하나 이상의 프로그램들은 상기 상위 문서들의 세트의 정렬된 리스트로 상기 사용자에게 상기 스니피트들을 나타내기 위한 명령들을 더 포함하는,

멀티-스테이지 질의 처리 시스템.

청구항 14

제 11 항에 있어서,

상기 제 3 세트의 관련도 스코어들은 상기 제 2 세트의 문서 식별자들에 대응하는 문서들에서의 상기 질의 용어들의 하나 이상의 위치들에 기초하는,

멀티-스테이지 질의 처리 시스템.

청구항 15

제 11 항에 있어서,

상기 제 2 세트의 관련도 스코어들은 상기 문서들에서의 토큰들의 속성들에 기초하고, 상기 속성들은 상기 문서들에서의 토큰들의 폰트 속성들을 포함하는,

멀티-스테이지 질의 처리 시스템.

청구항 16

제 11 항에 있어서,

상기 제 1 세트의 문서 식별자들을 검색하는 것은 상기 하나 이상의 질의 용어들에 대한 토큰 위치들의 리스트를 생성하기 위해 상기 인덱스를 사용하고 상기 토큰 위치들에 대응하는 문서 식별자들의 세트를 생성하기 위해 맵에 액세스하는 것을 포함하는,

멀티-스테이지 질의 처리 시스템.

청구항 17

컴퓨터에 의한 실행을 위해 구성된 하나 이상의 프로그램들을 저장하는 컴퓨터 판독가능 저장 매체로서,

상기 하나 이상의 프로그램들은:

질의의 제 1 스테이지 처리를 수행하기 위한 명령들 – 상기 질의의 제 1 스테이지 처리를 수행하기 위한 명령들은:

하나 이상의 질의 용어들에 응답하여 인덱스로부터 제 1 세트의 문서 식별자들을 검색하기 위한 명령들;

질의 용어들의 존재, 용어 빈도, 및 문서 대중도 중 하나 이상에 기초하여 상기 제 1 세트의 문서 식별자들의 적어도 하나의 서브세트에 대응하는 제 1 세트의 압축된 문서들에 대해 제 1 세트의 관련도 스코어들을 생성하여, 메모리에 상기 제 1 세트의 관련도 스코어들을 저장하기 위한 명령들

을 포함함 –;

상기 질의의 제 2 스테이지 처리를 수행하기 위한 명령들 – 상기 질의의 제 2 스테이지 처리를 수행하기 위한 명령들은:

토큰 위치들의 리스트, 상기 문서들에서의 질의 용어들 간의 거리들, 상기 문서들에서의 토큰들의 속성들, 및 상기 제 1 세트의 압축된 문서들 중 한 문서에 사용된 질의 용어 주변에 나타나는 텍스트 중 하나 이상에 기초하여 상기 제 1 세트의 압축된 문서들에서의 문서들에 대한 제 2 세트의 관련도 스코어들을 생성하여, 상기 메모리에 상기 제 2 세트의 관련도 스코어들을 저장하기 위한 명령들

을 포함함 –;

상기 메모리로부터 상기 제 1 세트의 관련도 스코어들 및 상기 제 2 세트의 관련도 스코어들을 판독하여, 상기 제 1 세트의 관련도 스코어들 및 상기 제 2 세트의 관련도 스코어들에 기초하여 추가 처리를 위한 문서들의 정렬된 리스트를 생성하기 위한 명령들;

상기 문서들의 정렬된 리스트에서의 문서들로부터 부가적인 질의 용어들을 자동으로 생성하기 위한 명령들;

상기 부가적인 질의 용어들을 이용하여 새로운 질의를 형성하기 위한 명령들;

상기 인덱스로부터 제 2 세트의 문서 식별자들을 검색하고 상기 부가적인 질의 용어들에 적어도 부분적으로 기초하여 제 3 세트의 관련도 스코어들을 생성하도록, 상기 새로운 질의를 처리하기 위한 명령들; 및

사용자에게 프리젠테이션하기 위한 상위 문서들의 세트를 선택하기 위해 상기 제 3 세트의 관련도 스코어들을 사용하기 위한 명령들

을 포함하는,

컴퓨터 판독가능 저장 매체.

청구항 18

제 17 항에 있어서,

상기 하나 이상의 프로그램들은:

제 1 세트의 토큰들을 복원하기 위해 상기 제 1 세트의 압축된 문서들의 적어도 일부분을 압축해제하고 - 상기 제 1 세트의 복원된 토큰들은 상기 제 1 세트의 압축된 문서들에서의 위치들과 연관됨 - ; 그리고

상기 제 1 세트의 복원된 토큰들을 이용하여 상기 제 1 세트의 압축된 문서들의 하나 이상의 부분들을 재구성함으로써, 상기 제 1 세트의 압축된 문서들에서의 문서들을 위한 스니피트들을 생성하기 위한

명령들을 더 포함하는,

컴퓨터 판독가능 저장 매체.

청구항 19

제 18 항에 있어서,

상기 하나 이상의 프로그램들은 상기 상위 문서들의 세트의 정렬된 리스트로 상기 사용자에게 상기 스니피트들을 나타내기 위한 명령들을 더 포함하는,

컴퓨터 판독가능 저장 매체.

청구항 20

제 17 항에 있어서,

상기 제 3 세트의 관련도 스코어들은 상기 제 2 세트의 문서 식별자들에 대응하는 문서들에서의 상기 질의 용어들의 하나 이상의 위치들에 기초하는,

컴퓨터 판독가능 저장 매체.

청구항 21

제 17 항에 있어서,

상기 제 2 세트의 관련도 스코어들은 상기 문서들에서의 토큰들의 속성들에 기초하고, 상기 속성들은 상기 문서들에서의 토큰들의 폰트 속성들을 포함하는,

컴퓨터 판독가능 저장 매체.

청구항 22

제 17 항에 있어서,

상기 제 1 세트의 문서 식별자들을 검색하기 위한 명령들은 상기 하나 이상의 질의 용어들에 대한 토큰 위치들의 리스트를 생성하기 위해 상기 인덱스를 사용하고 상기 토큰 위치들에 대응하는 문서 식별자들의 세트를 생성하기 위해 맵에 액세스하기 위한 명령들을 포함하는,

컴퓨터 판독가능 저장 매체.

청구항 23

제1항에 있어서,

상기 제 1 세트의 문서 식별자들은 압축된 문서들의 세트를 저장하는 토큰스페이스 저장소(repository)에서의

상기 질의 용어들에 대응하는 토큰들의 위치들에 대응하는,

멀티-스테이지 질의 처리 시스템의 질의 처리 방법.

청구항 24

제11항에 있어서,

상기 제 1 세트의 문서 식별자들은 압축된 문서들의 세트를 저장하는 토큰스페이스 저장소에서의 상기 질의 용어들에 대응하는 토큰들의 위치들에 대응하는,

멀티-스테이지 질의 처리 시스템

청구항 25

제17항에 있어서,

상기 제 1 세트의 문서 식별자들은 압축된 문서들의 세트를 저장하는 토큰스페이스 저장소에서의 상기 질의 용어들에 대응하는 토큰들의 위치들에 대응하는,

컴퓨터 판독가능 저장 매체.

명 세 서

기 술 분 야

[0001] 본 출원은 2004년 8월 13일자로 제출된 미국 특허출원 번호 10/917,745, "System and Method For Encoding And Decoding Variable-Length Data", 및 2004년 8월 13일자로 제출된 미국 특허출원 번호 10/917,739, "Document Compression System and Method For Use With Tokenspace Repository"에 관한 것으로서, 이 출원들은 그 전체가 본 발명에 참조로 포함된다.

[0002] 개시된 실시예들은 일반적으로 데이터 처리 시스템들 및 방법들에 관한 것으로서, 특히 연관 인덱스를 가진 문서들의 집합(이하에서 "토큰스페이스 저장소"로도 지칭됨)과 함께 사용하기 위한 멀티-스테이지 질의 처리 시스템 및 방법에 관한 것이다.

배 경 기 술

[0003] 정보 검색 시스템들(예를 들어, 검색 엔진들)은 문서 코퍼스(corpus)로부터 생성된 문서들의 인덱스(예를 들어, 월드 와이드 웹)에 대해 질의들을 매칭시킨다. 통상적인 인버스 인덱스(inverse index)는 문서들 내에서 이들의 위치들에 대한 포인터들과 함께 각 문서의 워드들을 포함한다. 문서 처리 시스템은 자동 또는 수동 프로세스를 이용하여 문서 코퍼스로부터 검색된 문서들의 내용들, 페이지들 또는 사이트들을 처리함으로써 반전된 인덱스를 마련한다. 문서 처리 시스템은 또한 질의에 응답할 때 질의 처리기(processor)에 의해 사용되기 위해 문서들의 내용들, 또는 내용의 부분들을 저장소(repository)에 저장할 수 있다.

[0004] 질의 결과들이 질의에 적절한 것을 보장하기 위해 보다 지능화된 질의 검색 및 스코어링(scoring) 기술들에 대한 지속적인 요구가 있다. 몇몇 스코어링 기술들은 예를 들어 문서들에서 발견된 질의 용어들 또는 키워드들의 문맥을 결정하기 위해, 후보 문서들의 부분적인 재구성을 요구할 수 있다. 불행히도, 이러한 지능화된 기술들의 도입은 부가적인 처리 및 관련 오버헤드로 인해 검색 성능의 저하를 초래할 수 있다.

발명의 상세한 설명

[0005] 개시된 실시예들은 토큰스페이스(tokenspace) 저장소와 함께 사용하기 위한 멀티-스테이지 질의 처리 시스템 및 방법을 포함한다. 멀티-스테이지 질의 처리 시스템 및 방법은 멀티-계층 맵핑 수단에 의해 용이해지는 점증적 문서 재구성을 통해, "스니피트(snippet: 코드 조각)" 생성을 포함하는 멀티-스테이지 질의 스코어링을 가능하게 한다. 멀티-스테이지 질의 처리 시스템의 하나 이상의 스테이지들에서, 관련도 스코어들의 세트는 사용자에게 정렬된(ordered) 리스트로서 나타내기 위해 문서들의 서브세트를 선택하는데 사용된다. 관련도 스코어들의 세트는 멀티-스테이지 질의 처리 시스템의 스테이지들 이전에 결정된 관련도 스코어들의 하나 이상의 세트들로부터 부분적으로 유도될 수 있다. 몇몇 실시예들에서, 멀티-스테이지 처리 시스템은 사용자 질의에 2개 이상의 패스들(passes)을 실행할 수 있고, 정렬된 리스트에서 문서들의 관련성을 개선하도록 후속

패스에서 사용하기 위해 사용자 질의를 확장하기 위해 각각의 패스로부터 정보를 이용할 수 있다.

실 시 예

[0018] **시스템 개요**

[0019] 도 1은 정보 검색 시스템(100)의 일 실시예의 블럭도이다. 정보 검색 시스템(100)은 문서 처리 시스템(102) 및 질의 처리 시스템(104)을 포함한다. 정보 검색 시스템(100)은, 인터넷(예를 들어, 월드 와이드 웹을 통해) 또는 인트라넷과 같은 하나 이상의 네트워크들 상에서 또는 사용자 컴퓨터상에서 로컬로(예를 들어, 파일들, 이메일, 애플리케이션들 등으로) 명시적인 또는 암시적인 문서 검색들을 수행하기 위한 하나 이상의 컴퓨터 시스템을 포함하는, 질의에 응답하여 정보를 검색할 수 있는 임의의 시스템일 수 있다. "문서들"이란 용어는 문서들, 웹 페이지들, 이메일들, 애플리케이션 특정 문서들, 및 데이터 구조들, 인스턴트 메시징(IM) 메시지들, 오디오 파일들, 비디오 파일들, 및 하나 이상의 컴퓨터 시스템들에 존재할 수 있는 임의의 다른 데이터 또는 애플리케이션들을 의미한다는 것에 유의한다.

[0020] 문서 처리 시스템

[0021] 문서 처리 시스템(102)은 일반적으로 하나 이상의 문서 저장소들(106), 렉시콘(lexicon) 생성기(108), 인코딩/디코딩 시스템(110) 및 토큰스페이스 저장소(112)를 포함한다. 인코딩/디코딩 시스템(110)은 하나 이상의 문서 저장소들(106)로부터 문서들을 검색하고, 문서들을 토큰들로 파싱(parsing)하며, 렉시콘 생성기(108)로부터 맵핑들을 이용하여 압축된 포맷으로 토큰들을 인코딩한 다음, 토큰스페이스 저장소(112)에 인코딩된 토큰들을 저장한다.

[0022] "토큰(token)"은 이에 제한됨이 없이, 용어들(terms), 구문들(phrases), 구두점(punctuation), HTML 태그들 등을 포함하는 문서에서 전형적으로 발견되는 임의의 객체일 수 있다. 파싱 이후, 문서들의 세트는 토큰들의 시퀀스로서 나타낸다. 또한, 토큰들의 시퀀스에서 각각의 토큰은 문서들의 세트에서 토큰의 위치를 나타내는 토큰 위치를 갖는다. 예를 들어, 문서들의 세트에서 제 1 토큰은 0의 위치에 할당되고, 문서들의 세트에서 제 2 토큰은 1의 위치 등으로 할당될 수 있다.

[0023] 일부 구현예들에서, 문서들을 디코딩하기 위해 사용된 컴퓨터들과 완전히 상이한 세트의 컴퓨터들이 문서들을 인코딩하기 위해 사용된다는 것을 유의한다. 예를 들어, 웹 크롤링(crawling) 시스템은 문서들을 인코딩하는 문서 처리 시스템(102)을 포함할 수 있고, 질의 처리 시스템(104)은 인코딩된 문서들의 선택된 부분들을 디코딩할 수 있다. 이러한 구현예들에서, 문서 처리 시스템(102)에 의해 형성된 문서 인버스 인덱스 및 토큰스페이스 저장소(112) 또는 이들의 사본들은 질의 처리 시스템(104)에 의해 사용된다.

[0024] 렉시콘 생성기(108)는 문서들을 파싱함으로써 문서들의 세트를 인코딩하기 위해 사용되는 맵핑들을 생성한다. 렉시콘 생성기(108)에 의해 형성되는 제 1 맵핑은 본 발명에서 글로벌-렉시콘(global-lexicon)으로 지칭되며, 문서들의 세트에서 모든 명시적 토큰들을 식별하고, 글로벌 토큰 식별자를 각각의 고유 토큰에 할당한다. 렉시콘 생성기(108)에 의해 형성되는 제 2 맵핑은 실제적으로 맵핑들의 시퀀스이고, 이들은 각각 본 발명에서 미니-렉시콘(mini-lexicon)으로 지칭된다. 각각의 미니-렉시콘은 문서들의 세트에서 각각의 범위의 위치들을 인코딩 및 디코딩하기 위해서만 사용된다. 글로벌-렉시콘 및 미니-렉시콘들의 생성 및 사용은 이하에서 보다 상세히 설명된다.

[0025] **질의 처리 시스템**

[0026] 질의 처리 시스템(104)은 인코딩/디코딩 시스템(110)에 결합된 하나 이상의 질의 처리기들(114) 및 토큰스페이스 인버스 인덱스(116)를 포함한다. 토큰스페이스 인버스 인덱스(116)는 문서들의 세트의 모든 GTokenID들을 문서들내의 이들의 위치들에 맵핑시킨다. 개념적으로, 인버스 인덱스(116)는 각각의 GTokenID에 대한 토큰 위치들의 리스트를 포함한다. 효율성을 위해, 각각의 GTokenID에 대한 토큰 위치들의 리스트는 인버스 인덱스에 의해 차지되는 공간의 크기를 감소시키기 위해 인코딩된다.

[0027] 몇몇 실시예들에서, 하나 이상의 질의 처리기(들)(114)는 하나 이상의 질의 처리기들(114)에 의해 질의 표현(예를 들어, 부울(Boolean) 트리 표현)으로 변환되는 다수의 질의 용어들로 질의를 파싱한다. 질의 용어들은 도 4에 대해 보다 완전히 기술되는 것처럼, 토큰 위치들을 검색하기 위해 토큰스페이스 인버스 인덱스(116)를 인덱싱하는데 사용된다. 몇몇 실시예들에서, 토큰 위치들은 도 5에 대해 기술되는 것처럼, 질의에 관련된 문서들을 스코어링하기 위해 멀티-스테이지 질의 처리 시스템에 사용된다. 질의 용어들에 응답하여, 질의 처리기들(114)은 하나 이상의 통신 모드들(예를 들어, 디스플레이 장치, 오디오 등)을 통해 사용자에게 나타나는

정렬된 문서들의 리스트를 생성한다.

[0028] **렉시콘 생성기**

[0029] 도 2는 도 1의 렉시콘 생성기(108)의 일 실시예의 개념적 블럭도이다. 렉시콘 생성기(108)는 글로벌-렉시콘 빌더(202)와 미니-렉시콘 빌더(204)를 포함한다.

[0030] **글로벌-렉시콘 빌더**

[0031] 글로벌-렉시콘 빌더(202)는 문서 저장소(106)로부터 문서들을 검색하고 고유 글로벌 토큰 식별자들(GTokenID들)을 문서들 내에 포함된 각각의 고유 토큰에 할당함으로써 글로벌-렉시콘(206)을 생성한다. 몇몇 실시예들에서, 문서 저장소(106)는 다수의 부분들(종종 파티션들로 불림)로 논리적으로 또는 물리적으로 분할되고, 개별 글로벌-렉시콘(206)은 각각의 파티션에 대해 생성된다. 일 실시예에서, 수십억개의 문서들의 세트가 수천개의 파티션들로 분할되고, 그 각각은 글로벌-렉시콘(206)을 생성하도록 처리된다. 전형적인 글로벌-렉시콘(206)은 수백만개의 고유 토큰들을 포함할 수 있다.

[0032] 몇몇 실시예들에서, 인코딩될 문서들의 세트(예를 들어, 하나의 파티션의 문서들)는 토큰들로의 문서들의 파싱 및 토큰들의 처리 이전에 하나 이상의 기준에 따라 분류된다. 문서들의 이러한 분류는 유사한 세트들의 워드들을 사용하는 문서들이 문서들의 세트에서 서로 근접하게 위치되기 때문에, 토큰화된 문서들의 효율적인 인코딩을 용이하게 할 수 있다. 결과적으로, 각각의 미니-렉시콘(이하에서 기술됨)은 평균적으로 그렇치 않은 경우보다 문서들의 세트의 더 큰 부분을 커버할 것이고, 보다 일반적으로는 문서들의 인코딩은 더 적은 공간을 차지할 것이다. 일 실시예에서, 문서들의 세트는 먼저 언어로 분류된 다음, 역으로 정렬되는 URL의 호스트 네임 부분의 필드들과 함께, 각 언어에 대한 문서들이 URL로 분류된다. 예를 들어, 언어에 의한 분류 이후, 모든 불어 문서들이 함께 그룹화되고, 그 다음 불어 문서들이 URL에 의해 분류될 것이다. URL에 의해 분류될 때, 각각의 URL은 초기에 h1.h2...hy.hz/n1/n2...의 패턴을 포함하고, 여기서 h1.h2...hy.hz는 URL의 호스트 네임 부분을 포함하며, /n1/n2는 URL의 나머지 부분을 나타낸다. URL은 URL로의 분류 이전에 hz.hy...h2.h1/n1/n2...의 패턴으로 다시 맵핑된다. 예를 들어, "www.google.com/about.html"의 URL은 "com.google.www/about.html"로 다시 맵핑된다. URL에 의한 분류 이전에 URL들의 호스트 네임 필드들을 반전시킴으로써, 문서들은 서로에 대해 논리적 근사화에 따라 분류된다. 따라서, 유사한 타입의 문서들(특정 언어에 대한 문서들의 그룹내에서)이 함께 그룹화되고; 각 문서 타입에 대한 문서들의 그룹내에서, 각각의 웹 사이트의 문서들이 함께 그룹화되며; 각 웹 사이트에 대한 문서들내에서, 웹 사이트의 다양한 브랜치들에 대한 문서들이 함께 그룹화되고; 이런 식으로 계속된다.

[0033] 몇몇 실시예들에서, 문서들은 하나 이상의 클러스터링 기술들을 이용하여 정렬된다. 문서들내에 포함된 용어들, 단어들 또는 구절들은 다양한 개념들에 관련되는 클러스터들로 문서들을 조직화하는데 사용될 수 있다. 예를 들어, 문서들에 대한 일반적인 정보(예를 들어, 식별된 문서들과 연관되거나 연관되지 않게 내장되는 메타-데이터), 식별된 문서들로부터의 샘플링된 내용, 및/또는 문서들에 대한 카테고리 정보가 문서들을 정렬하는데 사용될 수 있다.

[0034] 몇몇 실시예들에서, 문서들을 파싱하는 동안, 글로벌 렉시콘 빌더(202)는 문서들의 세트에서 각각의 고유 토큰의 발생 회수, 및 고유 토큰과 연관된 언어(있으면)와 같은, 각각의 식별된 고유 토큰에 대한 정보(도 2에 미도시됨)를 저장한다. 고유 토큰과 연관된 언어는 토큰이 발견되는 문서(들)와 연관되는 언어에 기초하여 결정될 수 있다. 특정 토큰이 언어 이상과 연관되는 문서들에서 발견되면, 토큰과 연관된 언어는 임의의 적절한 방법론을 이용하여 결정될 수 있다. 하나의 적절한 방법론은 고유 토큰들을 식별하도록 문서들의 세트를 파싱하는 동안 사용되는 통계적 방법론이다. 각각의 토큰은 발견되는 첫 번째 문서의 언어에 초기에 할당된 다음, 토큰에 할당된 현재 언어 이외의 언어의 문서에서 발생하는 토큰의 각각의 후속적인 발생동안, 0과 1 사이에서 랜덤하게(또는 의사-랜덤하게) 선택된 수가 1/N 미만인 경우에만, 토큰이 다른 언어에 재할당되며, 여기서 N은 토큰 발생의 현재 카운트이다. 다른 실시예들에서, 임의의 유사하거나 유사하지 않은 적절한 언어 할당 메커니즘이 각각의 고유 토큰과 언어를 연관시키는데 사용될 수 있다. 몇몇 실시예들에서, 언어는 구두점 심볼들을 나타내는 고유 토큰들과 연관되지 않는다. 또 다른 실시예에서, 언어가 모든 고유 토큰에 할당되는 동안, 가장 빈번하게 발생하는 N 토큰들(예를 들어, 256)을 처리할 때 언어 연관성은 무시된다. 결과적으로, 구두점 토큰들과 연관된 언어는 효과적으로 무시된다.

[0035] 몇몇 실시예들에서, 고유 토큰들, 및 연관되는 빈도(frequency)와 언어 정보의 리스트는 고유 토큰들의 발생의 빈도를 기초로 분류된다. 그 다음, 선택적으로, 엔트리들이 문서들의 세트의 공간 효율적인 인코딩을 촉진시키기 위해 추가로 분류될 수 있다. 예를 들어, 일 실시예에서, 모든 고유 토큰들은 발생 빈도에 의해 먼

저 분류된다. 그 다음, 고유 토큰들의 결과적인 분류된 리스트는 대역들(bands)로 나누어진다. 예를 들면, 상위 대역, Band 0은 상위 255 또는 256 토큰들(즉, 최고 빈도 카운트들을 갖는 토큰들)을 포함할 수 있다. 제 2 대역, Band 1은 Band 0의 토큰들을 제외하고, 상위 2^{14} (즉, 65,536) 토큰들을 포함할 수 있다. 제 3 대역, Band 2는 고유 토큰들의 분류된 리스트에서 다음 2^{14} (즉, 65,536) 토큰들을 포함할 수 있다. 물론, 각각의 대역에서 다른 실시예들에서 상이할 수 있다. 그 다음, 각 대역의 토큰들은 제 2 세트의 기준들에 따라 분류된다. 예를 들어, 일 실시예에서, 제 1 대역의 토큰들은 수치 또는 알파벳 값에 의해 알파벳으로 분류된다. 각각의 다른 대역들은 언어로 먼저 분류된 다음, 알파벳으로 분류된다. 결과적으로, Band 0 이외의 각 대역의 분류된 토큰들은 언어로 그룹화되고, 각각의 언어 그룹내에서 토큰들이 알파벳으로 분류된다. 다른 실시예들에서, 다른 분류 기준은 각각의 대역들에서 고유 토큰들을 분류하기 위해 사용될 수 있다.

[0036] 분류 프로세스는 고유 토큰들의 분류된 리스트를 생성하고, 고유 토큰들은 각각 리스트에서 각각의 위치를 갖는다. 그 다음, 각각의 분류된 고유 토큰은 고유 글로벌 토큰 식별자(이하에서 "GTokenID"로도 지칭됨)로 할당된다. GTokenID들은 문서 처리 시스템(102)(예를 들어, 32비트 무부호(unsigned) 정수들)을 구현하는데 사용되는 플랫폼에 따른 임의의 적절한 데이터 타입과 폭을 포함할 수 있다. 몇몇 실시예들에서, GTokenID들은 분류된 고유 토큰들에 오름차순으로 할당되어, 높은 빈도의 토큰들에는 작은 값의 GTokenID들이 할당되고 낮은 빈도의 토큰에는 큰 값의 GTokenID들이 할당된다. 보다 구체적으로는, 일 실시예에서, 토큰들의 분류된 리스트의 각 토큰은 고유 토큰들의 분류 리스트에서 수치 위치와 동일한 32비트 글로벌 토큰 식별자로 할당된다. 따라서, 리스트의 제 1 토큰에는 0과 동일한 GTokenID가 할당되고(즉, 16진수 포맷의 00000000), 리스트의 제 2 토큰에는 1과 동일한 GTokenID가 할당되며, 이런 식으로 계속된다. 고유 토큰 값들에 대한 GTokenID들의 맵핑들의 결과 세트는 본 발명에서 글로벌-렉시콘(206)으로 지칭된다. 몇몇 실시예들에서, 글로벌 렉시콘(206)은 실제적으로 2개의 맵핑 구조들을 포함하고, 그 중 하나는 GTokenID들을 토큰들로 맵핑하며, 다른 하나는 토큰들을 GTokenID들로 맵핑한다. GTokenID들로의 토큰들의 맵핑은 인코딩 프로세스 동안 사용되고, 토큰들로의 GTokenID들의 맵핑은 문서들의 부분들을 디코딩하는 동안 사용된다.

[0037] 이하에서 보다 완전히 설명되는 것처럼, 빈도를 기반으로 하는 고유 토큰들의 정렬은 미니-렉시콘들(208)을 저장하기 위해 요구되는 공간의 크기를 감소시키도록 돕는다. 이것은 하위 GTokenID들에 할당된 대역들의 토큰들이 상위 GTokenID들에 할당된 대역들의 토큰들보다 더 높은 발생 빈도를 갖기 때문에, 고유 토큰들의 대역들이 발생 빈도 이외의 기준을 기초로 분류되는 실시예들에서도 가능하다.

[0038] 몇몇 실시예들에서, HTML 태그들 및 구두점과 같이, 평균 토큰보다 더 빈번하게 발생하는 "특별" 토큰들에는 글로벌-렉시콘(206)에서 GTokenID들의 프리픽스(prefix)(205) 부분을 차지하는 GTokenID들이 할당된다(예를 들어, $GTokenID_0$-$GTokenID_{N-1}$). 모든 다른 GTokenID들은 프리픽스(205)에 할당된 마지막 특별 GTokenID에 의해 오프셋될 수 있다.

[0039] 상기한 논의에서, GTokenID들은 32비트 무부호 정수 값들과 같은 고정된 길이 값들로서 기술된다. 그러나, 이러한 동일한 GTokenID들은 GTokenID들이 저장을 위해 인코딩될 때, 제로와 같은 최상위 바이트들(또는 비트들)이 인코딩 동안 버려지거나 감춰질 수 있다. 예를 들어, 몇몇 실시예들에서, 2^8 미만의 값을 갖는 모든 GTokenID들은 단일 바이트 값으로서 인코딩되고, 2^{16} 미만의 값을 갖는 모든 GTokenID들은 2-바이트 값으로서 인코딩되며, 2^{24} 미만의 값을 갖는 모든 GTokenID들은 3-바이트 값으로서 인코딩된다. 이러한 방식으로, 문서들의 세트에서 가장 높은 발생 빈도들을 갖는 토큰들은 더 낮은 발생 빈도를 갖는 토큰들보다 더 짧은 길이의 GTokenID들에 의해 나타낸다.

[0040] 이하에서 기술되는 실시예들에서, 토큰스페이스 저장소는 가변-길이 GTokenID들이 아닌 고정 길이 LTokenID들로 채워진다. 그러나, 토큰스페이스 저장소의 LTokenID들을 원래 토큰들(물론, 가변 길이임)로 다시 맵핑하는 것은 다수의 "미니-렉시콘들"의 저장을 요구하고, 미니-렉시콘들의 내용은 GTokenID들을 포함한다. 미니-렉시콘들을 효율적으로 저장하기 위해, 각각의 미니-렉시콘의 GTokenID들은 가변 길이 값들로서 처리될 수 있다. 선택적으로, 각각의 미니-렉시콘의 GTokenID들은 먼저 델타 인코딩된 리스트로서 처리된 다음, 결과적인 델타 값들이 가변 길이 인코딩 수단을 이용하여 인코딩된다.

[0041] **미니-렉시콘 빌더**

[0042] 글로벌-렉시콘(206)이 생성된 이후, 인코딩/디코딩 시스템(110)에 의해 사용되기 위해 미니-렉시콘들(208)의 세트가 미니-렉시콘 빌더(204)에 의해 생성된다. 미니-렉시콘(208)의 각 엔트리는 GTokenID 및 대응하는 로

컬 토큰 식별자(LTokenID)를 포함한다. 각각의 엔트리에 대한 LTokenID는 미니-렉시콘(208)의 엔트리의 위치에 의해 내포되어 있기 때문에, 명시적으로 저장될 필요는 없다. 각각의 미니-렉시콘(208)은 토큰화된 문서들에서 다른 각각의 특정 범위의 토큰 위치들을 인코딩 및 디코딩하기 위해서만 사용되므로, 동일 세트의 LTokenID들이 각각의 미니-렉시콘(208)에 의해 사용될 수 있다. 예를 들어, P(예를 들어, 256)개의 엔트리들을 갖는 제 1 미니-렉시콘(208)(예를 들어, 미니-렉시콘 A)은 문서들을 통해 파싱됨에 따라 미니-렉시콘 빌더(204)에 의해 인카운터(encounter)되는 제 1 P 고유 토큰들에 대해 생성된다. 제 1 P개의 고유 토큰들이 인카운터되면, 제 1 미니-렉시콘(208)이 유효한 토큰 위치들의 범위에 대해, 개시(starting) 토큰 위치, $Start_Pos_A$를 포함하는 "유효 범위 맵"(210)의 제 1 엔트리가 형성된다. 제 1 미니-렉시콘(208)의 각각의 P LTokenID들은 고유 GTokenID에 할당된다. 모든 LTokenID들이 GTokenID들로 할당되면, 제 2 미니-렉시콘(208)(예를 들어, 미니-렉시콘 B)은 미니-렉시콘 빌더(204)에 의해 인카운터되는 다음 P개의 고유 토큰들에 대해 생성되고, 제 2 미니-렉시콘(208)이 유효한 위치들의 범위의 개시 토큰 위치, $Start_Pos_B$를 포함하는 유효 범위 맵(210)에서 제 2 엔트리가 형성된다. 따라서, $Start\_Pos_B$ 내지 $Start\_Pos_C$ -1 범위내에 속하는 토큰화된 문서들의 위치를 갖는 토큰은 도 2에 도시된 것처럼, 미니-렉시콘 B를 이용하여 디코딩될 수 있다.

[0043] 구체적인 예를 제공하기 위해, 일 실시예에서 각각의 미니-렉시콘의 LTokenID들은 0 내지 255의 값들을 갖고, 그 각각은 8비트 무부호 정수로 나타내는 반면, GTokenID들은 32비트 무부호 정수들이다. 제 1 미니-렉시콘은 미리 규정된 수 P(예를 들어, 256)의 상이한 토큰들이 식별될 때까지, 토큰 위치 0에서 개시하는 문서들의 세트를 스캐닝함으로써 생성된다. 상이한 P개의 토큰들에 대한 GTokenID들은 리스트에서 어셈블링된다. 몇몇 실시예들에서, 리스트의 GTokenID들은 상위 리스트에서 가장 작은 GTokenID들을 갖는 수치 값으로 분류된다. 그 다음, LTokenID들은 리스트의 GTokenID들의 위치들에 따라, 리스트의 GTokenID들에 할당된다. 예를 들어, 리스트의 제 1 GTokenID에는 0의 LTokenID가 할당되고, 리스트의 다음 GTokenID에는 1의 LTokenID가 할당되며, 이런 식으로 계속된다. GTokenID들로의 LTokenID들의 결과적인 맵핑은 소위 미니-렉시콘(208)으로 지칭된다. $Start\_Pos_A$ 내지 $Start_Pos_B$의 토큰 위치들의 범위는 미니-렉시콘과 연관된다. 제 2 미니-렉시콘은 제 1 미니-렉시콘과 연관되는 마지막 위치 직후의 위치 $Start_Pos_B$에서 개시하는 문서들의 세트를 스캐닝함으로써 생성된다. 스캐닝은 미리 규정된 P개의 상이한 토큰들이 식별될 때까지 계속되고, 그 지점에서 제 2 미니-렉시콘은 전술한 것과 동일한 프로세스를 이용하여 생성된다. 미니-렉시콘 빌더(204)는 문서들의 모든 토큰들이 미니-렉시콘들(208)로 맵핑될 때까지 문서들의 세트에서 토큰 위치들의 순차적인 범위들에 대해 미니-렉시콘들(208)의 시퀀스를 지속적으로 생성한다.

[0044] 선택적 실시예에서, 각각의 미니-렉시콘(208)의 제 1 F LTokenID들은 문서들의 세트에서 가장 빈번한(popular) F 토큰들에 대해 예약된다. 이러한 F LTokenID들에 대해, LTokenID는 항상 GTokenID와 동일하다. 이러한 할당 수단은 문서들의 신속한 디코딩을 촉진시킨다. F-1 또는 그 이하의 값을 갖는 LTokenID(토큰스페이스 저장소에서)가 디코딩될 때마다, LTokenID를 해당 GTokenID로 먼저 맵핑할 필요 없이 글로벌-렉시콘에 따라 직접적으로 토큰에 맵핑될 수 있다.

[0045] 동일 세트의 LTokenID들(예를 들어, 0 내지 255)은 각각의 미니-렉시콘(208)에 사용된다. 문서들의 압축을 용이하게 하기 위해, LTokenID들은 GTokenID들(예를 들어, 4 바이트) 보다 더 작은 폭(예를 들어, 1 바이트)을 갖는다. 이러한 폭들의 차이(예를 들어, 3 바이트)는 토큰스페이스 저장소(112)에 토큰화된 문서들을 저장하는데 사용되는 토큰 당 바이트 수의 감소를 나타낸다. 각각의 LTokenID가 1 바이트를 차지하는 실시예에서, 10억개의 토큰들을 갖는 문서들의 세트는 토큰스페이스 저장소(112)에서 10억 바이트(1GB)를 차지하고, 다른 지지 데이터 구조들에 의해 차지된 공간은 무시된다(본 명세서에서 이후에 기술됨).

[0046] 미니-렉시콘들(208)을 생성하는 프로세스가 종료되면, 토큰화된 문서들에서 모든 토큰은 토큰화된 문서들의 위치를 기반으로 미니-렉시콘(208)과 연관된다. 토큰화된 문서들에서 각각의 고유 토큰은 토큰이 하나 이상의 위치 범위에서 발생하면 하나 이상의 미니-렉시콘(208)과 연관될 수 있다. 일 실시예에서, 평균 문서는 대략적으로 1100개의 토큰들을 갖고, 평균 미니-렉시콘(208)은 약 1000개의 토큰들에 이른다.

[0047] 각각의 미니-렉시콘(208)이 생성된 이후, 문서들의 세트의 해당 부분에서 토큰들은 인코딩/디코딩 시스템(110)에 의해 LTokenID들로 맵핑되고 후속 검색을 위해 토큰스페이스 저장소(112)에 저장된다. 이러한 맵핑과 함께, 문서 저장소(106)의 모든 토큰은 토큰스페이스 저장소(112)에서 고정된 길이(예를 들어, 1 바이트)의 LTokenID로 맵핑된다. 따라서, 디코딩/압축해제 동안, 디코딩 프로세스를 감속(slow down)시킬 수 있는 스킵 테이블들 또는 이와 동일한 데이터 구조들을 필요로 함이 없이 토큰스페이스 저장소(112)의 하나의 토큰

위치에서 다른 토큰 위치로 점프할 수 있다.

[0048] 몇몇 실시예들에서, 미니-렉시콘들(208)은 압축된 포맷으로 인코딩되고 문서 재구성이 필요할 때까지 저장된다. 일 실시예에서, 각각의 미니-렉시콘(208)에서 GTokenID들의 분류된 리스트는 델타 인코딩된 다음, 델타 값들의 결과 리스트는 압축 포맷, 바람직하게는 미니-렉시콘의 신속하고 효율적인 디코딩 및 재구성을 용이하게 하는 포맷으로 인코딩된다. 적절한 데이터 구조 및 인코딩/디코딩 방법은 공동 계류중인 2004년 8월 13일자로 제출된 미국 특허출원 번호 10/917,745, "System and Method For Encoding And Decoding Variable-Length Data"에 기술된다.

[0049] 특정 문서를 압축해제하기 위해, 그 문서에 대한 토큰 위치들의 범위와 연관된 미니-렉시콘들(208)은 LTokenID들을 이들의 해당 GTokenID들로 변환하는 미니-렉시콘들(208)의 엔트리들로부터 형성되는 변환 테이블들 또는 맵핑들로 압축해제된다. 따라서, 토큰스페이스 저장소(112)에서 토큰화된 문서의 디코딩은 문서에 대해 토큰스페이스 저장소(112)에 저장된 고정 길이의 LTokenID들을 판독하고, LTokenID들을 해당 GTokenID들로 변환하기 위해 문서의 각각의 토큰 위치에 대해 미니-렉시콘을 액세스함으로써 달성된다. 그 다음, GTokenID들은 글로벌-렉시콘(206)을 이용하여 해당 토큰들(예를 들어, 텍스트 및 구두점)로 맵핑됨으로써, 문서의 모든 부분 또는 일부를 재구성할 수 있다.

[0050] **인코딩 시스템**

[0051] 도 3A는 토큰스페이스 저장소의 인코딩 문서들을 위한 인코딩 시스템(300)의 일 실시예의 블럭도이다. 인코딩 시스템(300)은 선택적인 전처리기(302), 선택적인 델타 인코더(304) 및 가변-길이 데이터 인코더(306)를 포함한다. 가변-길이 데이터는 이에 제한됨이 없이, 예를 들어 정수들, 문자열들, 부동 소수점, 고정 소수점 등과 같은 다양한 데이터 타입들을 포함할 수 있다. 가변-길이 데이터는 이에 제한됨이 없이, 텍스트, 이미지들, 그래픽들, 오디오 샘플들 등을 포함한다.

[0052] 몇몇 실시예들에서, 정보의 리스트는 효율적인 인코딩을 위해 정보를 정렬하는 전처리기(302)에 의해 수신된다. 전처리기(302)는 하나 이상의 분류 알고리즘들을 이용하여 단조로운(monotonic) 시퀀스로 데이터를 정렬할 수 있다. 예를 들어, 정수들의 세트가 값으로 분류되면, 인접한 정수들은 근접한 크기를 가질 것이므로, 델타 인코더(304)가 인코딩을 위한 작은 값의 정수들인 델타 값들을 생성할 수 있게 한다. 정렬된 데이터는 델타 인코더(304)에 의해 수신되며, 상기 델타 인코더(304)는 작은 값의 정수들을 얻기 위해 정렬된 데이터의 인접 쌍들 사이의 차이들을 계산한다. 작은 값의 정수들은 가변-길이 데이터 인코더(306)에 의해 수신되어, 효율적으로 디코딩될 수 있는 압축 포맷으로 데이터를 인코딩한다. 적절한 가변-길이 데이터 인코더(306)의 일 예는 공동 계류 중인 2004년 8월 13일자로 제출된 미국 특허출원 번호 10/917,745, "System and Method For Encoding And Decoding Variable-Length Data"에서 보다 완전히 기술된다.

[0053] 문서 처리 시스템(102)에 의해 생성되는 다양한 정보는 인코딩 시스템(300)의 모든 부분 또는 일부를 이용하여 인코딩될 수 있다. 몇몇 실시예들에서, 각각의 미니-렉시콘(208)의 GTokenID들은 가장 근접한 크기의 정수 값들이 델타 인코딩되는 것을 보장하기 위해 전처리기(302)를 이용하여 분류된다. 그 다음, 정렬된 GTokenID들은 차이 값들 또는 잔류 값들을 제공하도록 델타 인코더(304)에 의해 델타 인코딩된다. 차이 값들은 가변-길이 데이터 인코더(306)를 이용하여 압축 포맷으로 그룹들(예를 들어, 4 값들의 그룹들)로 인코딩된다. 몇몇 실시예들에서, 인버스 인덱스의 토큰 위치들의 리스트들은 도 4에 대해 보다 완전히 기술되는 것처럼, 위치들의 신속하고 효율적인 디코딩을 용이하게 하기 위해 인코딩된다.

[0054] 가변-길이 데이터 인코더(306)가 신속하고 효율적인 디코딩을 용이하게 하는 압축 포맷을 제공하지만, 정보의 리스트를 압축하기 위해 다른 공지된 인코딩 수단들이 문서 처리 시스템(102)에 사용될 수도 있다(예를 들어, CCITT-G4, LZW 등).

[0055] 디코딩 시스템

[0056] 도 3B는 토큰스페이스 저장소의 문서들을 디코딩하기 위한 디코딩 시스템(308)의 일 실시예의 블럭도이다. 디코딩 시스템(308)은 가변-길이 데이터 디코더(310) 및 선택적인 델타 디코더(312)를 포함한다. 몇몇 실시예들에서, 데이터의 인코딩된 그룹들은 하나 이상의 오프셋/마스크 테이블들의 보조를 통해 상기 그룹들을 디코딩하는 가변-길이 데이터 디코더(310)에 의해 수신된다. 디코딩된 데이터는 델타 디코더(312)에 의해 수신되고 실행(running) 합들을 계산하여, 정보의 원래 리스트와 동일한 델타-디코딩된 데이터를 생성한다. 가변-길이 정수 값들로 인코딩된 그룹을 디코딩함에 있어서 오프셋/마스크 테이블들의 사용은 공동 계류 중인 미국 특허출원 번호 10/917,745, "System and Method For Encoding And Decoding Variable-Length Data"에 보다

상세히 기술된다.

[0057] **속성 인코딩/디코딩 시스템**

[0058] 도 3C는 문서 속성들을 인코딩/디코딩하기 위한 속성 인코딩/디코딩 시스템(314)의 일 실시예의 블럭도이다. 속성 인코딩/디코딩 시스템(314)은 속성 테이블(316)에 저장을 위해 속성 정보(322)를 속성 기록들(318)로 인코딩하는 인코딩/디코딩 시스템(320)을 포함한다. 문서에 대한 속성들은 토큰 단위로 결정되고, 0 또는 1 비트 값은 주어진 토큰에 대한 각각의 속성의 존재 또는 부재를 나타내는데 사용된다. 예를 들어, 속성 테이블의 속성 기록(318)은 A × K 비트 맵으로서 개념적으로 나타낼 수 있으며, 여기서 A는 인코딩된 속성들의 수이고, K는 그 속성들이 기록(318)에 의해 나타낸 토큰들의 수이다. A가 8이고 K가 32이면, 각각의 속성 기록(318)은 각각의 32개의 토큰들에 대해 8개의 속성들을 저장한다. 각각의 속성 기록(318)은 질의 처리 동안 선택된 속성 기록들의 매우 신속한 디코딩을 가능하게 하면서 속성 테이블에 의해 차지되는 공간의 크기를 압축하기 위해 인코딩될 수 있다. 속성 기록들(318)을 인코딩 및 디코딩하기 위한 한 가지 적절한 방법론은 공동 계류 중인 2004년 8월 13일자로 제출된 미국 특허출원 번호 10/917,745, "System and Method For Encoding And Decoding Variable-Length Data"에 기술된다. 선택적으로, 각각의 속성 기록에서 정보는 런-길이 인코딩될 수 있다.

[0059] 속성 테이블(316)에서 기록되는 속성들의 세트는 하나 이상의 폰트 속성들(예를 들어, 굵게, 밑줄 등), 하나 이상의 문서 위치 속성들(예를 들어, 제목, 헤딩), 메타데이터 및 문서들의 세트에서 토큰들을 구별하는데 사용될 수 있는 임의의 다른 피쳐들 또는 특징들을 포함할 수 있다.

[0060] 몇몇 실시예들에서, 문서들의 세트에서 토큰들의 속성들은 상술한 것처럼, 토큰화된 문서들이 인코딩되고 토큰스페이스 저장소에 저장되는 동시에, 식별되고 인코딩된다. 인코딩된 속성들은 도 5에 대해 보다 상세히 기술되는 것처럼, 관련도 스코어링의 하나 이상의 스테이지들에 사용된다.

[0061] 문서 저장소 인코딩 및 디코딩 시스템 - 제 2 실시예

[0062] 도 8A 및 도 8B는 전술한 실시예와는 다소 상이한 방식으로 문서들의 토큰화된 수집("토큰스페이스 저장소")이 인코딩되는 일 실시예의 블럭도들이다. 전술한 것처럼, 글로벌 렉시콘 빌더(202)는 문서들(106)의 세트를 토큰화하고, 모든 고유 토큰들을 식별하며, 글로벌 토큰 식별자들을 모든 고유 토큰들에 할당한다. 그 결과는 글로벌 렉시콘(206)이다. 그 다음, 문서들의 세트(토큰화됨)는 영역(region) 렉시콘 빌더(804)에 의해 처리된다. 개념적으로, 문서들의 세트는 영역들(820)로 분할되고, 각각의 영역(820)은 블럭들(822)로 분할된다. 영역 렉시콘 빌더(804)는 각 영역에 대해 "렉시콘" 또는 사전(830)을 형성하고, 인코딩 시스템(810)은 각각의 영역에 대해 인코딩된 토큰들(832)의 세트와 각 영역에 대한 블럭 오프셋들(834)의 세트를 생성한다. 영역 렉시콘(830), 인코딩된 토큰들(832) 및 블럭 오프셋들(834)(그 각각은 다음에 보다 상세히 기술될 것임)은 문서들의 세트의 각 영역(820)의 인코딩된 표현을 함께 형성한다.

[0063] 일 실시예에서, 문서들의 세트는 영역들(820)로 분할되고, 그 각각(아마도 마지막 영역을 제외하고)은 8192개의 토큰들(또는 임의의 다른 적절한 크기)과 같은 미리 결정된 고정 크기를 갖는다. 영역(820)의 각각의 블럭(822)은 64 토큰들(또는 임의의 다른 적절한 크기)과 같이 미리 규정된 고정 크기를 갖는다.

[0064] 일 실시예에서, 각각의 영역(820)에 대한 "렉시콘"(830)은 가장 높은 반복율들, 또는 임의의 유사한 구조를 갖는 토큰들의 가장 긴 시퀀스들의 정렬된 리스트이다. 렉시콘(830)은 영역에서 후보 토큰 열들의 테이블을 생성함으로써 형성될 수 있고, 렉시콘(830)은 영역내에서 이들의 반복 카운트들을 결정한 다음, 최대 렉시콘 크기에 도달될 때까지 가장 최선의 후보들을 선택한다. 예시적인 실시예에서, 최대 렉시콘 크기는 64개의 토큰들이지만, 임의의 다른 적절한 크기 제한값이 다른 실시예들에서 사용될 수 있다. 다음에서 기술될 것처럼, 렉시콘(830)은 각 영역(820)의 각각의 블럭들(822)을 인코딩하기 위한 콘텍스트(context)로서 사용되고, 영역의 고 압축 표현을 가능하게 한다. 몇몇 실시예들에서, 하나 이상의 영역 렉시콘들(830)은 예를 들어, 본 명세서에서 이전에 참조된 2004년 8월 13일자로 제출된 미국 특허출원 번호 10/917,745, "System and Method For Encoding And Decoding Variable-Length Data"에 기술된 인코딩 방법을 이용하여 압축 포맷으로 인코딩될 수 있다.

[0065] 도 9A 및 도 9B를 참조하면, 일 실시예에서, 인코딩 시스템(810)은 다음과 같이 토큰들의 각각의 블럭(822)을 인코딩한다. 해당 영역에 대한 렉시콘(830)은 블럭의 토큰들 직전의 토큰들의 세트로서 처리된다. 시퀀스에서, 블럭의 토큰들은 첫 번째 토큰부터 마지막 토큰까지 처리되고, 각각의 토큰을 매칭시키며, 렉시콘(830)을 포함하는 토큰들의 선행하는 시퀀스에서 매칭되는 가장 긴 토큰 시퀀스와 가능한 많은 후속 토큰들을 매칭시

킨다. 매칭되는 선행 시퀀스가 발견되면, "복사 코드"가 생성된다. 그렇치 않으면, "문자(literal) 코드"는 토큰을 나타내기위해 생성된다. 그 다음, 현재 코드에 의해 커버되는 모든 토큰들은 블럭에서 다음 토큰(있다면)의 후속 처리를 위한 선행 토큰들로서 처리된다. 도 9B에 나타낸 것처럼, 블럭에서 토큰들의 세트를 나타내는 각각의 "코드"는 타입 필드(902)를 포함할 수 있다. 코드가 "문자 코드"이면, 코드의 제 2 부분(904)은 글로벌 토큰 식별자를 나타낸다. 몇몇 실시예들에서, 이러한 타입 필드(902)는 글로벌 토큰 식별자를 나타내기 위해 요구되는 비트들의 수를 의미한다. 예를 들어, 일 실시예에서, 타입 코드(902)는 7개까지의 상이한 문자 코드들을 표시할 수 있고, 그 각각은 해당 글로벌 토큰 식별자 길이를 갖는다. 다른 실시예들에서, 상이한 타입 코드들의 수는 8 이상 또는 8 미만일 수 있다(예를 들어, 하나는 복사 코드를 나타내고, 나머지는 문자 코드들을 나타냄). 문자 코드가 "복사 코드"이면, 코드의 제 2 부분(906)은 포인터(908)와 길이(910)를 포함할 수 있고, 여기서 포인터(908)는 선행하는 텍스트의 어디에서 개시되는지를 나타내며, 길이(910)는 매칭되는 시퀀스의 길이를 나타낸다(즉, 디코딩 동안 복사될 토큰들의 수). 따라서, 만약 4개의 토큰들의 매칭 시퀀스가 인코딩 시스템(810)에 의해 발견되면, 현재 위치에 선행하는 31개 토큰들 위치에서 시작하며, 이러한 시퀀스에 대한 코드는 다음과 같다:

[0066] <type=copy, ptr=31, length=4>.

[0067] 복사 코드(비트로 측정되는 바와 같이)의 길이는 영역 렉시콘(830)의 최대 토큰 길이 및 블럭의 최대 토큰 길이, 매칭 시퀀스의 최대 허용 길이, 및 상이한 코드들의 수에 따를 것이다. 일 예에서, 총 12비트에 대해, 타입 필드(902)는 3비트(8 타입 코드들을 허용함)이고, 포인터 필드(908)는 7비트이며, 길이 필드(910)는 2비트이다. 복사 코드의 각 필드에 대한 다른 비트 길이들은 다른 실시예들에서 사용될 수 있다. 각각의 문자 코드의 길이(비트들로 측정되는 바와 같이)는 문자 코드의 타입에 의해 특정된다.

[0068] 도 8B를 다시 참조하면, 인코딩 시스템(810)이 영역의 블럭들을 인코딩함에 따라, 인코딩 시스템(810)은 영역의 각 블럭에 대해 인코딩된 토큰들의 위치들을 나타내는 블럭 오프셋들(834)의 세트를 생성한다. 일 실시예에서, 영역의 제 1 블럭의 블럭 오프셋은 토큰 저장소로의 포인터이고, 영역에 대한 각각의 다른 블럭 오프셋들은 영역의 제 1 블럭의 개시 위치에 대한 상대적 오프셋이다. 일 실시예에서, 영역 렉시콘들(830) 및 블럭 오프셋들(834)은 고정된 영역 크기로 분할되는 영역들(820)의 개시 위치들에 따라 인덱싱되는 테이블 또는 이와 동등한 데이터 구조에 저장된다. 다른 관점에서, 각각의 영역(820)에는 고정된 영역 크기로 분할되는 개시 위치를 포함하는 영역 넘버(Region Number)가 할당되고, 영역 렉시콘들(830)과 블럭 오프셋들(834)이 저장된 데이터 구조(들)는 영역 넘버로 인덱싱된다.

[0069] 영역(820)의 블럭(822)을 디코딩하는 것은 해당 영역의 영역 렉시콘(830)을 위치시키고, 영역의 블럭 오프셋들(834)을 이용하여 인코딩된 블럭을 위치시킨 다음, 글로벌 토큰 식별자들의 시퀀스를 형성하기 위해 블럭에 대한 코드들의 세트를 디코딩함으로써 달성된다. 그 다음, 글로벌 토큰 식별자들, 또는 임의의 그 서브세트의 결과 시퀀스는 글로벌 렉시콘(206)을 이용하여 심볼들 또는 용어들의 해당 세트로 변환될 수 있다.

[0070] **질의 처리 시스템**

[0071] 도 4는 토큰스페이스 저장소와 함께 사용하기 위한 질의 처리 시스템(104)의 제 1 스테이지의 일 실시예의 블럭도이다. 질의 처리 시스템(104)은 글로벌-렉시콘(402), 토큰스페이스 인버스 인덱스(408), 제 1 스테이지 룩업 테이블(406) 및 제 2 스테이지 룩업 테이블(410)을 포함한다. 질의 용어들 또는 열(string)들은 변환 테이블 또는 글로벌 렉시콘(402)의 엔트리들로부터 형성된 맵핑을 이용하여 질의 용어들을 GTokenID들로 변환하는 글로벌-렉시콘(402)에 의해 수신된다. GTokenID들은 GTokenID들을 인버스 인덱스(408)에 저장된 인덱스 기록들(412)로 맵핑하기 위한 맵(404)을 포함하는 인버스 인덱스(408)에 의해 수신된다. 맵(404)을 이용하여 식별된 각각의 인덱스 기록(412)은 토큰스페이스 저장소(112)의 토큰 위치들에 직접적으로 해당하는 토큰 위치들의 리스트를 포함한다. 몇몇 실시예들에서, 인버스 인덱스(408)는 글로벌-렉시콘이 생성된 이후 생성되고, 미니-렉시콘들을 생성하는데 사용되는 문서들을 동일하게 통과하는 동안 생성될 수 있다.

[0072] 몇몇 실시예들에서, 인버스 인덱스(408)는 제 1 스테이지 룩업 테이블(406)로의 인덱스로서 사용될 수 있는 위치들의 리스트를 제공한다. 질의가 다수의 용어들을 포함하면, 위치들의 다수의 리스트들은 인버스 인덱스(408)에 의해 형성된다. 위치들의 리스트(들)에서 각각의 위치에 해당하는 엔트리에 대해 전체 DocID 맵(410)을 검색해야 하는 것을 방지하기 위해, 제 1 스테이지 룩업 테이블(406)은 토큰스페이스 저장소의 위치들의 각 블럭에 대해 하나의 엔트리를 갖는다. 예를 들어, 각각의 블럭은 32,768개의 위치들의 크기를 가질 수 있고, 각각의 엔트리는 위치들의 해당 블럭에 대해 DocID 룩업 테이블(410)의 제 1 엔트리에 대한 포인터를 가질 수 있다. 따라서, 제 1 스테이지 룩업 테이블(406)은 종종 DocID 테이블(410)로 지칭되는 제 2 스테

이지 룩업 테이블(410)에서 문서 식별자(DocID) 엔트리들(412)에 대한 개시점 위치들로 위치들의 리스트(들)를 변환한다. 대안적으로, 테이블들(406, 410)은 공통적으로 DocID 룩업 테이블로 지칭될 수 있다. 제 2 스테이지 룩업 테이블(410)의 각 엔트리(412)는 DocID(문서 식별자) 및 해당 문서에 대한 개시 저장소 위치를 포함한다. 임의의 문서의 마지막 토큰은 제 2 스테이지 룩업 테이블의 다음 엔트리(412)에 의해 식별되는 개시 위치 직전의 위치이다. DocID들에 대한 개시점 위치들 $Start_Pos_{A-Z}$는 개시점 위치들을 각각의 질의 용어들에 대한 DocID들의 리스트로 변환하는 제 2 룩업 테이블(410)에 의해 수신된다.

[0073] 몇몇 실시예들에서, 제 1 스테이지 질의 처리기는 결과 세트를 형성하기 위한 로직(416)을 포함한다. DocID들의 리스트들은 DocID들의 결과 세트를 형성하기 위해, 질의 또는 질의 트리에 의해 특정되는 부울 로직에 따라, 로직(416)에 의해 병합된다. 또한, 로직(416)은 결과 세트의 DocID들에 해당하는 문서들 내에 위치되지 않는 토큰 위치들을 제거하도록 토큰 위치들의 리스트들을 선택적으로 필터링할 수 있다. 더욱이, 스코어링 함수는 스코어(종종 질의 스코어로 지칭됨)를 결과 세트의 각 DocID와 연관시키기 위해 DocID들에 의해 식별되는 각 문서내의 토큰 위치들과 DocID들을 이용하여, 결과 세트에 적용될 수 있다.

[0074] **멀티-스테이지 질의 처리**

[0075] 도 5는 토큰스페이스 저장소(524)와 함께 사용하기 위한 멀티-스테이지 질의 처리 시스템(500)의 일 실시예의 블럭도이다. 몇몇 실시예들에서, 질의 처리 시스템(500)은 제 1 스테이지 질의 처리기(510), 제 2 스테이지 질의 처리기(514), 제 3 스테이지 질의 처리기(518) 및 제 4 스테이지 질의 처리기(520)를 포함하는, 질의 처리 및 관련도 스코어 생성의 4 스테이지들을 포함한다. 애플리케이션에 따라, 더 많은 또는 더 적은 질의 처리 스테이지들이 시스템(500)에 사용될 수 있다는 점에 유의하라. 각각의 스테이지는 사용자에게 리턴 및/또는 애플리케이션에 따라 이전 스테이지들에서 생성되는 관련도 스코어들과 조합될 수 있는 하나 이상의 세트들의 관련도 스코어들을 계산한다.

[0076] **질의 처리 - 스테이지 I**

[0077] 제 1 스테이지 질의 처리기(510)는 일반적으로 도 4와 관련하여 기술된다. 질의 열(502)은 토큰화되고 질의 파서(query parser)(504)에 의해 파싱된다(즉, 질의에서 각각의 개별 용어가 토큰으로서 처리됨). 토큰화된 질의 용어들은 도 2 및 도 4와 관련하여 이전에 기술된 것처럼, 변환 테이블 또는 맵핑을 이용하여, 글로벌-렉시콘(508)에 의해 해당 GTokenID들로 변환된다. 사용자들은 부울, 인접 또는 근접 연산자들을 포함하는, 이들의 질의 열의 특별 연산자들을 사용할 수 있기 때문에, 시스템(500)은 질의 용어들과 연산자들로 질의를 파싱한다. 이러한 연산자들은 예약된 구두점(예를 들어, 인용 부호) 또는 특별한 포맷(예를 들어, AND, OR)의 예약된 용어들의 형태로 생성될 수 있다. 자연어 처리(NLP) 시스템의 경우, 연산자들은 연산자들이 표현될 수 있는 방법(예를 들어, 전치사들, 접속사들, 정렬 등)에 무관하게 사용되는 언어로 절대적으로 인식될 수 있다. 다른 질의 처리는 삭제되는 불용어(stop words)(예를 들어, "a", "the" 등) 및 용어 스테밍(stemming)(즉, 단어 접미사들의 제거)와 같이, 제 1 스테이지 질의 처리기(510)에 포함될 수도 있다.

[0078] 그 다음, GTokenID들의 리스트는 질의 열에 사용되는 임의의 연산자들을 고려하는 다른 질의 표현(예를 들어, 부울 표현) 또는 질의 트리를 생성하는 질의 확장기(506)에 의해 처리된다. 선택적으로, 질의 확장기(506)는 다양한 방법들로 질의를 확장시킬 수도 있다. 예를 들어, 질의 용어는 용어, 하나 이상의 동의어들 또는 질의 용어와 관련된 다른 용어들을 포함하는 서브트리로 변환될 수 있고, 서브트리의 용어들은 OR 연산자 또는 모(parent) 노드에 의해 서로 관련된다.

[0079] 이하에서 보다 상세히 기술되는 것처럼, 몇몇 실시예들에서, 질의는 도 5에 도시된 질의 처리 스테이지들의 시퀀스에 의해 1회 이상 처리된다. 각각의 패스(마지막을 제외한)에서, 부가적인 질의 확장 용어들이 생성된 다음(이하에서 설명되는 바와 같이), 이러한 부가적인 용어들은 질의 트리에 부가된다. 질의 트리는 스코어링 트리로서 사용될 수도 있고, 가중치들은 질의 트리의 용어들과 연관된다. 확장된 질의 트리는 또한 질의에 응답하는 문서들에 나타내는데 요구되지 않지만 질의에 응답하여 문서들의 관련도를 스코어링하는데 사용되는 용어들의 서브트리들 및 보충적인 용어들을 포함할 수 있다. 하나 이상의 질의 용어가 있다면, 제 1 패스 동안 가중치들은 검색 결과들을 개선시키도록 질의 용어들에 대해 계산될 수 있다.

[0080] 몇몇 실시예들에서, 시스템(500)을 통과하는 제 1 패스는 문서 코퍼스로부터 문서들의 랜덤한 샘플을 처리한다. 랜덤 샘플의 크기는 문서 코퍼스에 대해 질의를 매칭시키는 다수의 문서들을 추정하기 위해 시스템(500)에 의해 사용될 수 있는 하나 이상의 더 작은 랜덤 샘플들을 기초로 선택될 수 있다. 다른 실시예들에서, 제 1 문서 코퍼스(예를 들어, 질의 세션들의 세트)는 시스템(500)을 통과하는 제 1 패스에 사용되고, 상이한 제 2 코퍼스는 시스템(500)을 통과하는 제 2 또는 후속 패스에 사용된다. 질의 세션들의 이전의 세트들을 이

용하면 시스템(500)이 유사 질의들에서 공통으로 상호 발생하는 다른 관련 용어들을 결정할 수 있다. 이러한 관련 용어들은 후속적인 패스들에 대해 질의를 확장시키도록 질의 확장기(506)에 의해 사용될 수 있다.

[0081] 제 1 스테이지 질의 처리기(510)는 토큰스페이스 인버스 인덱스(512)에 대해 검색하고 질의에 매칭되는 문서들을 식별하기 위해 질의 용어들을 사용한다. 제 1 스테이지 질의 처리기(510)는 질의 트리의 용어들에 대한 토큰 위치들의 리스트(토큰스페이스 저장소 위치들로도 지칭됨)를 생성하도록 인버스 인덱스(512)를 액세스하고 토큰 위치들에 해당하는 문서들에 대한 DocID들의 세트를 생성하도록 DocID 맵(516)을 액세스한다. 또한, 제 1 스테이지 질의 처리기(510)는 질의에 응답하는 DocID들의 세트를 생성하기 위해 질의 또는 질의 트리에 의해 특정되는 부울 로직을 수행한다. 몇몇 실시예들에서, 제 1 스테이지 질의 처리기(510)는 또한 하나 이상의 스코어링 알고리즘들을 기초로 질의와 각 문서 사이의 관련도 스코어들(S_1)의 제 1 세트를 계산한다. 일반적으로, 스코어링 알고리즘들은 이에 제한됨이 없이, 질의 용어(들)의 존재 또는 부재, 용어 빈도, 부울 로직 수행, 질의 용어 가중치들, 다수의 문서들의 대중도(예를 들어, 문서의 중요도 또는 대중도 또는 상호연관성의 질의 독립적 스코어), 서로에 대한 질의 용어들의 근접도, 문맥, 속성들 등을 포함하는 하나 이상의 질의 특징들을 기반으로 각각의 매칭 문서에 대한 관련도 순위들을 제공한다. 일 실시예에서, 관련도 스코어들(S_1)의 제 1 세트는 질의 용어들의 존재, 용어 빈도, 및 문서 대중도를 포함하는 인자들의 세트를 기반으로 한다.

[0082] 몇몇 실시예들에서, 관련도 스코어들(S_1)의 제 1 세트는 단순히 클릭하고 내부 포인터들을 선택된 문서로 전달할 수 있는 사용자에게 정렬된 리스트로서 나타내기 위해 문서들을 선택하는데 사용될 수 있다. 다른 실시예들에서, DocID들 및 해당 위치들과 함께, 관련도 스코어들(S_1)의 제 1 세트는 추가적인 처리를 위해 제 2 스테이지 질의 처리기(514)로 제공된다.

[0083] **질의 처리 - 스테이지 Ⅱ**

[0084] 제 2 스테이지 질의 처리기(514)는 DocID들의 세트, 해당 문서들에 대한 토큰스페이스 저장소 위치들의 리스트, 및 제 1 스테이지 질의 처리기(510)로부터의 제 1 세트의 관련도 스코어들(S_1)을 수신한다. 제 2 스테이지 질의 처리기(514)는 문서들에서 발견되는 질의 용어들의 관련 위치들 또는 근접도를 기반으로 관련도 스코어들(S_2)의 제 2 세트를 생성하기 위해 위치들의 리스트를 사용한다. 질의의 용어들이 문서내에서 서로에 대해 근접하게 발생할 때, 문서는 용어들이 더 큰 거리에서 발생하는 경우보다 더 질의에 대해 관련성이 있다고 보여진다. 따라서, 관련도 스코어들(S_2)의 제 2 세트는 용어들이 일정 거리에서 발생하는 문서들과 비교하여, 질이 용어들이 서로 인접하거나 긴밀하게 근접하여 발생하는 경우, 문서들에 더 높은 순위를 매기는데 사용된다. 몇몇 실시예들에서, 관련도 스코어들(S_2)의 제 2 세트는 간단히 클릭하여 내부 포인터들을 선택된 문서로 전달할 수 있는 사용자에게 정렬된 리스트로서 나타내기 위해 상위 X 문서들을 선택하는데 사용될 수 있다. 몇몇 실시예들에서, 관련도 스코어들(S_2)의 제 2 세트는 사용자에게 나타내기 위해 및/또는 제 3 스테이지 질의 처리기(518)에 의한 추가적인 처리를 위해 문서들의 정렬된 리스트(관련도 스코어들 S_2의 제 2 세트에 따라 정렬됨)를 생성하기 위해 (예를 들어, 제 2 스테이지 질의 처리기(514)에 의해 사용되는 부가적인 스코어링 인자들에 따라 S_1 스코어들을 조절함으로써) 관련도 스코어 S_1의 제 1 세트로부터 부분적으로 유도된다.

[0085] **질의 처리 - 스테이지 Ⅲ**

[0086] 몇몇 실시예들에서, 제 2 스테이지 질의 처리기(514)는 도 3C에 대해 이전에 기술된 것처럼, 속성 테이블(522)에 인코딩된 용어 속성들(예를 들어, 폰트 속성들, 제목, 헤딩들, 메타데이터 등)을 처리하기 위해 제 3 스테이지 질의 처리기(518)에 결합된다. 제 3 스테이지 질의 처리기(518)는 DocID들의 세트, 해당 문서들에 대한 토큰스페이스 저장소 위치들의 리스트, 및 제 2 스테이지 질의 처리기(514)로부터의 관련도 스코어들 S_2의 제 2 세트를 수신한다. 대안적으로, 제 3 스테이지 질의 처리기는 제 2 세트의 관련도 스코어들 S_2 뿐만 아니라 제 1 세트의 관련도 스코어들 S_1을 수신한다.

[0087] 몇몇 연구들은 문서의 용어의 위치가 그 중요도를 문서에 나타낸다는 것을 보여준다. 예를 들어, 질의 용어를 매칭시키는 문서의 제목에서 발생하는 용어들은 문서의 바디에서 발생하는 질의 용어들 보다 더 크게 가중될 수 있다. 유사하게, 섹션 헤딩들 또는 문서의 제 1 문단에서 발생하는 질의 용어들은 문서내에서 덜 중요한 위치들에서 발생하는 용어들 보다 질의에 대한 문서의 관련도를 보다 더 나타낼 수 있는 것으로 보인다.

관련도의 지시자들로서 사용될 수 있는 다른 속성들은 굵은 텍스트, 밑줄 텍스트 및 폰트 크기를 포함한다. 따라서, 제 3 세트의 스코어들(S_3)은 질의 용어들을 매칭시키는 문서들의 토큰들의 속성들을 이용하여 결정된다. 도 3C를 참조하면, 문서의 질의 용어들에 대한 속성들(매칭되는 토큰들의 속성들, 또는 질의 용어들에 관련된 속성들)을 액세스하기 위해, 문서의 질의 용어들의 토큰 위치들은 속성 테이블(316)로의 인덱스로 사용된다(도 5의 522). 보다 구체적으로는, 그 속성들이 각각의 속성 기록(318)에 의해 인코딩되는 토큰들의 수가 K이면, K로 분할되는 토큰 위치들은 속성 테이블(316)로의 인덱스로 사용된다. 몇몇 실시예들에서, 식별된 속성 기록 또는 기록들(318)은 인코딩된 압축 형태로 저장되므로, 각각의 질의 용어들과 연관된 속성들을 결정하기 위해 디코딩되어야 한다.

[0088] 몇몇 실시예들에서, 제 3 세트의 관련도 스코어들(S_3)은 간단히 클릭하여 내부 포인터들을 선택된 문서로 전달할 수 있는 사용자에게 정렬된 리스트로서 나타내기 위해 상위 Y 문서들을 선택하는데 사용될 수 있다. 몇몇 실시예들에서, 제 3 세트의 관련도 스코어들(S_3)은 사용자에게 나타내기 위해 및/또는 제 4 스테이지 질의 처리기(520)에 의한 추가적인 처리를 위해, 문서들의 정렬된 리스트를 생성하도록 하나 이상의 제 1 및 제 2 세트의 관련도 스코어들(S_1, S_2)로부터 부분적으로 유도된다. 일 실시예에서, S_3 스코어들은 제 3 스테이지 질의 처리기(518)에 의해 생성되는 부가적인 스코어링 인자들에 따라 S_2 스코어들을 조절함으로써 형성된다.

[0089] **질의 처리 - 스테이지 Ⅳ**

[0090] 제 4 스테이지 질의 처리기(520)는 DocID들의 세트, DocID들에 해당하는 문서들의 위치들의 리스트, 및 제 3 스테이지 질의 처리기(518)로부터의 제 3 세트의 관련도 스코어들(S_3)을 수신한다. 제 4 스테이지 질의 처리기(520)는 제 1 및/또는 제 2 세트의 관련도 스코어들(S_1, S_2)도 선택적으로 수신할 수 있다. 제 4 스테이지 질의 처리기(520)는 하나 이상의 미니-렉시콘 맵들(523), 토큰스페이스 저장소(524) 및 하나 이상의 글로벌-렉시콘 맵들(508)에 결합되는 디코딩 시스템(527)에 결합된다. 미니-렉시콘 맵들(523), 토큰스페이스 저장소(524) 및 글로벌 렉시콘 맵들(508)은 모두 도 1 및 도 2와 관련하여 이전에 기술되었다.

[0091] 제 4 스테이지 질의 처리기(520)는 문맥을 기초로 제 4 세트의 관련도 스코어들(S_4)을 생성하고, 결과 세트에 리스트되는 하나 이상의 문서들에 대한 "스니피트(snippet)"를 생성할 수도 있다. 스니피트들은 문서로부터의 텍스트의 작은 부분들이고, 검색되는 키워드들 주변에서 나타나는 텍스트를 전형적으로 포함한다. 일 실시예에서, 결과 세트에 리스트되는 문서에 대한 스니피트를 생성하기 위해, 질의 처리기는 문서에 나타나는 각 질의 용어의 제 1 발생 이전 및 이후에 위치된 미리 규정된 수의 토큰들을 디코딩하고, 이에 따라 문서의 하나 이상의 텍스트 부분들을 재구성한 다음, 스니피트에 포함되도록 텍스트 부분들의 서브세트를 선택한다. 결과 세트의 위치들의 리스트를 이용하여, 디코딩 시스템(527)은 문서의 질의 용어들의 발생들에 선행 및 후속하는 문서의 부분들을 디코딩하는데 필요한 미니-렉시콘들(523)을 선택할 수 있다. 선택된 미니-렉시콘들(523) 및 글로벌-렉시콘(508)은 토큰스페이스 저장소의 LTokenID들을 GTokenID들로 변환한 다음, 도 2와 관련하여 상술된 것처럼, GTokenID들을 토큰들로 변환하는데 사용된다.

[0092] 몇몇 실시예들에서, 제 4 세트의 관련도 스코어들(S_4)은 간단히 클릭하여 내부 포인터들을 선택된 문서로 전달할 수 있는 사용자에게 정렬된 리스트로서 나타내기 위해 상위 Z 문서들을 선택하는데 사용될 수 있다. 몇몇 실시예들에서, 제 4 세트의 관련도 스코어들(S_4)은 사용자에게 나타내기 위해 및/또는 관련 피드백 모듈(517)에 의한 추가적인 처리를 위해, 문서들의 정렬된 리스트를 생성하도록, 하나 이상의 제 1 , 제 2 및 제 3 세트의 관련도 스코어들(S_1, S_2, S_3)로부터 부분적으로 유도된다. 대안적인 실시예에서, 최종 스테이지 질의 처리기는 선행하는 질의 처리 스테이지에 의해 형성되는 관련도 스코어들에서 가장 높은 스코어들을 갖는 문서들에 대한 스니피트들을 생성하지만, 새로운 세트의 관련도 스코어들(S_4)을 생성하지 않는다.

[0093] 몇몇 실시예들에서, 최종 세트의 관련도 스코어들은 마지막 질의 스테이지에 의해 형성되는 결과 세트의 문서들을 기초로 하나 이상의 새로운 질의 확장 용어들을 생성하는 관련 피드백 모듈(517)에 제공된다. 예를 들어, 관련 피드백 모듈(517)은, 전체 문서 접근법을 기반으로 하는 의사-관련도(pseudo-relevance) 피드백 알고리즘들(전체 웹 페이지를 기반으로 하는 의사 관련도 피드백), 문서 객체 모델(DOM) 세그먼트화, 비전-기반 페이지 세그먼트화(VIPS), 개념 격자들(concept lattices)을 이용한 개념적 관련도 피드백 등을 포함하지만 이에 제한되지 않는, 하나 이상의 공지된 관련도 피드백 알고리즘들을 수행할 수 있다. 관련도 피드백 알고리즘들은 이전의 질의 처리 스테이지들로부터 검사된 문서들을 분석하고, 분석 결과들을 기초로 질의 확장 용

어들을 생성할 수 있다. 새로운 질의 확장 용어들은 하나 이상의 질의 처리기들(510, 514, 518, 520)에 의해 처리될 새로운 질의 표현을 생성하는 질의 확장기(506)에 제공된다. 따라서, 멀티-스테이지 질의 처리 시스템(500)은 질의에 대해 2회 이상의 패스들을 실행할 수 있고, 궁극적으로 사용자가 보다 관련 있는 문서들을 수신하는 개선된 질의들을 생성하기 위해, 각각의 패스로부터의 정보를 이용할 수 있다.

[0094] 일 실시예에서, 최종 질의 스테이지 처리기(520)는 예를 들어, 문서에서 질의 용어들의 각각의 발생에 선행 및 후속하는 N(예를 들어, 10 내지 40)개의 토큰들을 포함하는 질의의 선행하는 제 1 패스를 수행할 때 긴 스니피트들을 생성한다. 스니피트는 미리 규정된 길이를 초과하는 경우 종결될 수 있다. 마지막 질의 스테이지(520)에 의해 형성되는 질의 및 긴 스니피트들은, 질의 확장 용어들의 세트, 및 선택적으로 질의 용어 가중치들의 세트를 생성하기 위해, 관련도 스코어들과 함께 관련 피드백 모듈(517)에 제공된다. 확장된 질의의 제 2 패스 처리 동안, 마지막 질의 스테이지(520)는 가장 높은 스코어들 또는 최상의 스코어들을 갖는 결과 세트의 문서들의 리스트로 디스플레이하기 위한 적절한 길이 및 컨텐츠의 짧은 스니피트들을 생성한다.

[0095] 일 실시예에서, 질의 처리 시스템은 L개의 병렬 질의 처리 서브-시스템들을 포함하고, 그 각각은 인버스 인덱스(512) 및 문서들의 집합의 각각의 서브세트를 위한 토큰스페이스 저장소(524)를 포함한다. 예를 들어, 질의 처리 시스템은 천개 이상의 병렬 질의 처리 서브-시스템들을 포함할 수 있다. 관련 피드백 모듈(517)(도 5)은 모든 질의 처리 서브-시스템들에 의해 공유될 수 있다. 질의 처리 시스템을 통과하는 제 1 패스 동안, 질의는 병렬 질의 처리 서브-시스템들의 작은 부분에 의해 처리되고, 제 2 패스 동안, 질의는 전체 질의 처리 시스템에 의해 처리된다. 예를 들어, 질의 처리 시스템은 S개의 서브세트들(예를 들어, 32 서브세트들)로 분할될 수 있고, 각각의 질의는 해쉬(hash) 함수를 정규화된 버전의 질의에 적용한 다음, 해쉬 기능에 의해 형성되는 결과에 모듈로(modulo) 함수를 적용하는 결과에 따라 서브세트들 중 하나에 할당된다. 질의 처리 시스템의 각각의 서브세트는 질의 처리 시스템의 "파티션"으로 지칭될 수 있고, 각각의 질의 처리 서브-시스템은 "서브-파티션"으로 지칭될 수 있다.

[0096] 질의의 제 1 패스 처리의 주 목적은 질의의 제 2 패스 처리에 의해 형성되는 질의 결과들의 품질을 개선하기 위해, 질의 확장 용어들 및 질의 용어 가중치들의 세트를 생성하는 것이다. 질의 처리 시스템의 문서들이 질의 처리 서브-시스템들에 대해 상당히 랜덤하게 분포되는 한, 질의 확장 용어들의 세트를 생성하기 위해 단지 작은 수의 서브-시스템들에 의해 질의가 처리될 필요가 있다. 질의 확장 용어들은 확장된 질의 트리 또는 질의 표현을 생성한 다음 상술한 것처럼 질의 처리 스테이지들(질의의 제 2 패스 처리에서)에 의해 처리되도록, 질의 확장기(506)에 의해 사용된다. 예를 들어, "뉴욕 사진들"의 질의는 "뉴욕(사진들 또는 이미지들 또는 이미지 또는 사진)"으로 확장될 수 있다. 제 2 패스 동안 마지막 질의 스테이지에 의해 생성되는 결과 세트와 스니피트들은 질의가 수신되는 컴퓨터 또는 장치에 의한 디스플레이(또는 보다 일반적으로 프리젠테이션)를 위해 포맷팅될 수 있다.

[0097] 일 실시예에서, 질의의 제 1 패스 처리는 후속하는 패스들과 상이한 데이터베이스에서 수행된다. 제 1 패스를 위한 초기 데이터베이스는 이전 처리된 질의들의 데이터베이스일 수 있고, 후속적인 패스들을 위한 데이터베이스는 데이터베이스의 문서들에 질의 용어들을 맵핑시키기 위한 인버스 인덱스를 갖는 문서들의 세트일 수 있다.

[0098] **문서 처리 서버**

[0099] 도 6은 토큰스페이스 저장소 서버(600)의 일 실시예의 블럭도이다. 서버(600)는 독립형 컴퓨터 시스템 또는 다중 컴퓨터 시스템들을 포함하는 분배 처리 시스템의 일부일 수 있다. 서버(600)는 일반적으로 하나 이상의 처리 유닛들(CPUs)(604), 하나 이상의 네트워크 또는 다른 통신 인터페이스들(608), 메모리(602), 및 이러한 컴포넌트들을 상호접속시키기 위한 하나 이상의 통신 버스들(606)을 포함한다. 서버(600)는 사용자 인터페이스, 예를 들어 디스플레이 및 키보드를 선택적으로 포함할 수 있다. 메모리(602)는 고속 랜덤 액세스 메모리를 포함할 수 있고, 하나 이상의 자기 디스크 스토리지 장치들과 같은 비휘발성 메모리를 포함할 수도 있다. 메모리(602)는 중앙 처리 유닛(들)(604)으로부터 원격으로 위치된 대용량 스토리지를 포함할 수 있다.

[0100] 메모리(602)는 운영 체제(610)(예를 들어, 리눅스 또는 유닉스), 네트워크 통신 모듈(612), 렉시콘 생성기(614)(예를 들어, 렉시콘 생성기(108)), 인코딩 시스템(616)(예를 들어, 인코딩 시스템(300)), 하나 이상의 글로벌-렉시콘들(618)(예를 들어, 글로벌-렉시콘(206)), 하나 이상의 미니-렉시콘들(620)(예를 들어, 미니-렉시콘들(208)), 토큰스페이스 저장소(622)(예를 들어, 토큰스페이스 저장소(112)), 속성 기록들(624)(예를 들어, 속성 기록 테이블(316)), 및 유효 범위 맵(626)(예를 들어, 유효 범위 맵(210))을 저장한다. 각각의 이러한 컴포넌트들의 운영은 도 1 내지 도 5에 대해 이전에 기술되었다.

[0101] **질의 처리 서버**

[0102] 도 7은 질의 처리 서버(700)의 일 실시예의 블럭도이다. 서버(700)는 독립형 컴퓨터 시스템 또는 다중 컴퓨터 시스템들을 포함하는 분배 처리 시스템의 일부일 수 있다. 서버(700)는 일반적으로 하나 이상의 처리 유닛들(CPUs)(704), 하나 이상의 네트워크 또는 다른 통신 인터페이스들(708), 메모리(702), 및 이러한 컴포넌트들을 상호접속시키기 위한 하나 이상의 통신 버스들(706)을 포함한다. 서버(700)는 예를 들어, 디스플레이 및 키보드와 같은 사용자 인터페이스를 선택적으로 포함할 수 있다. 메모리(702)는 고속 랜덤 액세스 메모리를 포함할 수 있고, 하나 이상의 자기 디스크 스토리지 장치들과 같은 비휘발성 메모리를 포함할 수도 있다. 메모리(702)는 중앙 처리 유닛(들)(704)으로부터 원격으로 위치된 대용량 스토리지를 포함할 수 있다.

[0103] 메모리(702)는 운영 시스템(710)(예를 들어, 리눅스 또는 유닉스), 네트워크 통신 모듈(712), 토큰스페이스 인버스 인덱스(714)(예를 들어, 토큰스페이스 인버스 인덱스(408)), 디코딩 시스템(716)(예를 들어, 디코딩 시스템(308)), 하나 이상의 렉시콘 변환 테이블들 또는 맵핑들(718)(예를 들어, 글로벌-렉시콘(206) 및 미니-렉시콘들(208)로부터 유도됨), 유효 범위 맵(720)(예를 들어, 유효 범위 맵(210)), DocID 맵(722)(예를 들어, DocID 맵(410)), 질의 파서(724)(예를 들어, 질의 파서(504)), 질의 트리(726), 하나 이상의 질의 처리기들(728)(예를 들어, 질의 처리기들(510, 514, 518, 520)), 속성 기록들(730)(예를 들어, 속성 기록 테이블(316)), 및 토큰스페이스 저장소(732)(예를 들어, 토큰스페이스 저장소(112))를 저장한다. 각각의 이러한 컴포넌트들의 운영은 도 1 내지 도 5에 대해 이전에 기술되었다.

[0104] 전술한 상세한 설명은 설명을 목적으로 특정 실시예들을 참조로 기술되었다. 그러나, 상기한 예시적인 논의들은 개시된 정확한 형태들로 본 발명을 제한하거나 소모하려는 의도가 아니다. 많은 변형들과 변화들은 상기한 기술들의 관점에서 가능할 수 있다. 실시예들은 본 발명의 원리들과 그 실제적인 애플리케이션들을 최선으로 설명하기 위해 선택되고 기술되며, 이에 따라 통상의 당업자가 본 발명 및 고려되는 특정 사용에 적합한 다양한 변형들을 가진 다양한 실시예들에 최상으로 활용할 수 있도록 한다.

도면의 간단한 설명

[0006] 도 1은 정보 검색 시스템의 일 실시예의 블럭도이다.

[0007] 도 2는 도 1의 렉시콘(lexicon) 생성기의 일 실시예의 개념적 블럭도이다.

[0008] 도 3A는 토큰스페이스 저장소의 문서들을 인코딩하기 위한 인코딩 시스템의 일 실시예의 블럭도이다.

[0009] 도 3B는 토큰스페이스 저장소의 문서들을 디코딩하기 위한 디코딩 시스템의 일 실시예의 블럭도이다.

[0010] 도 3C는 인코딩/디코딩 문서 속성들에 대한 속성 인코딩/디코딩 시스템의 일 실시예의 블럭도이다.

[0011] 도 4는 토큰스페이스 저장소와 함께 사용하기 위한 질의 처리 시스템의 일 실시예의 블럭도이다.

[0012] 도 5는 토큰스페이스 저장소와 함께 사용하기 위한 멀티-스테이지 질의 처리 시스템의 일 실시예의 블럭도이다.

[0013] 도 6은 토큰스페이스 저장소 서버의 일 실시예의 블럭도이다.

[0014] 도 7은 질의 처리 서버의 일 실시예의 블럭도이다.

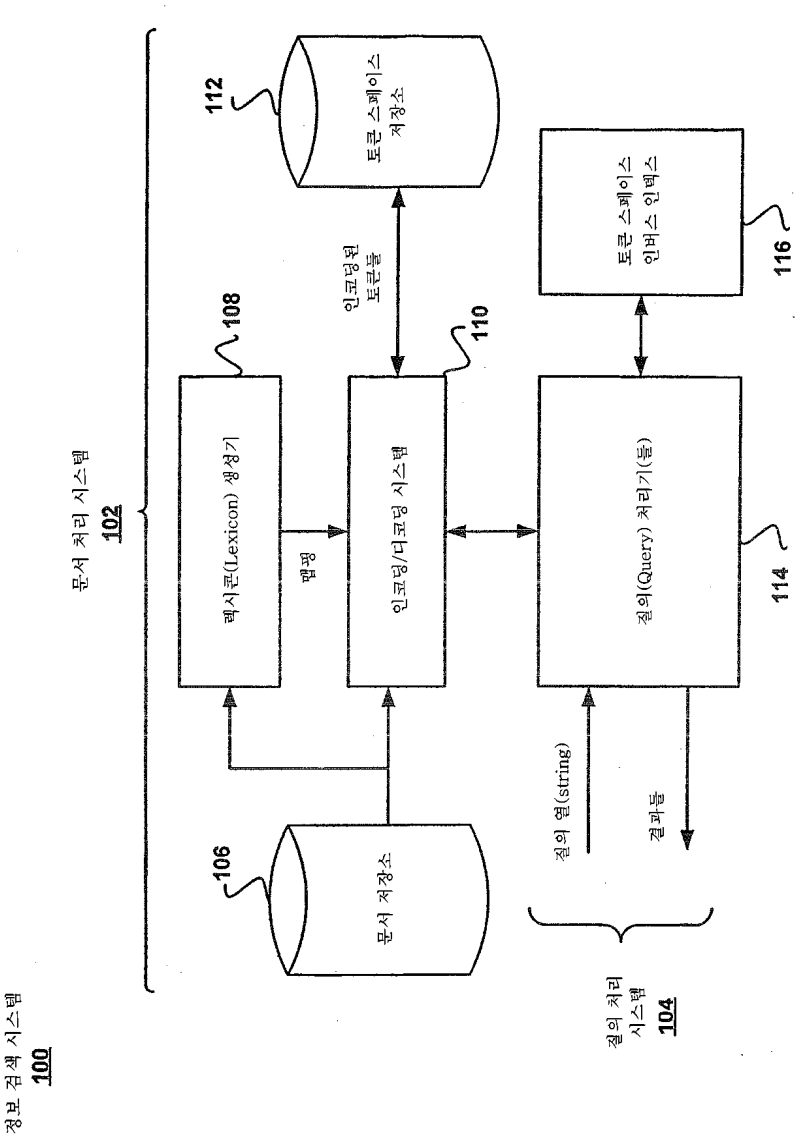
[0015] 도 8A는 토큰화된 문서 저장소의 제 2 실시예의 블럭도이고, 도 8B는 도 1의 렉시콘 생성기의 제 2 실시예의 개념적 블럭도이다.

[0016] 도 9A는 렉시콘 생성기의 실시예에 사용된 인코딩 프로세스의 개념적 블럭도이고, 도 9B는 인코딩된 토큰들을 나타내기 위한 예시적인 데이터 구조들을 도시한다.

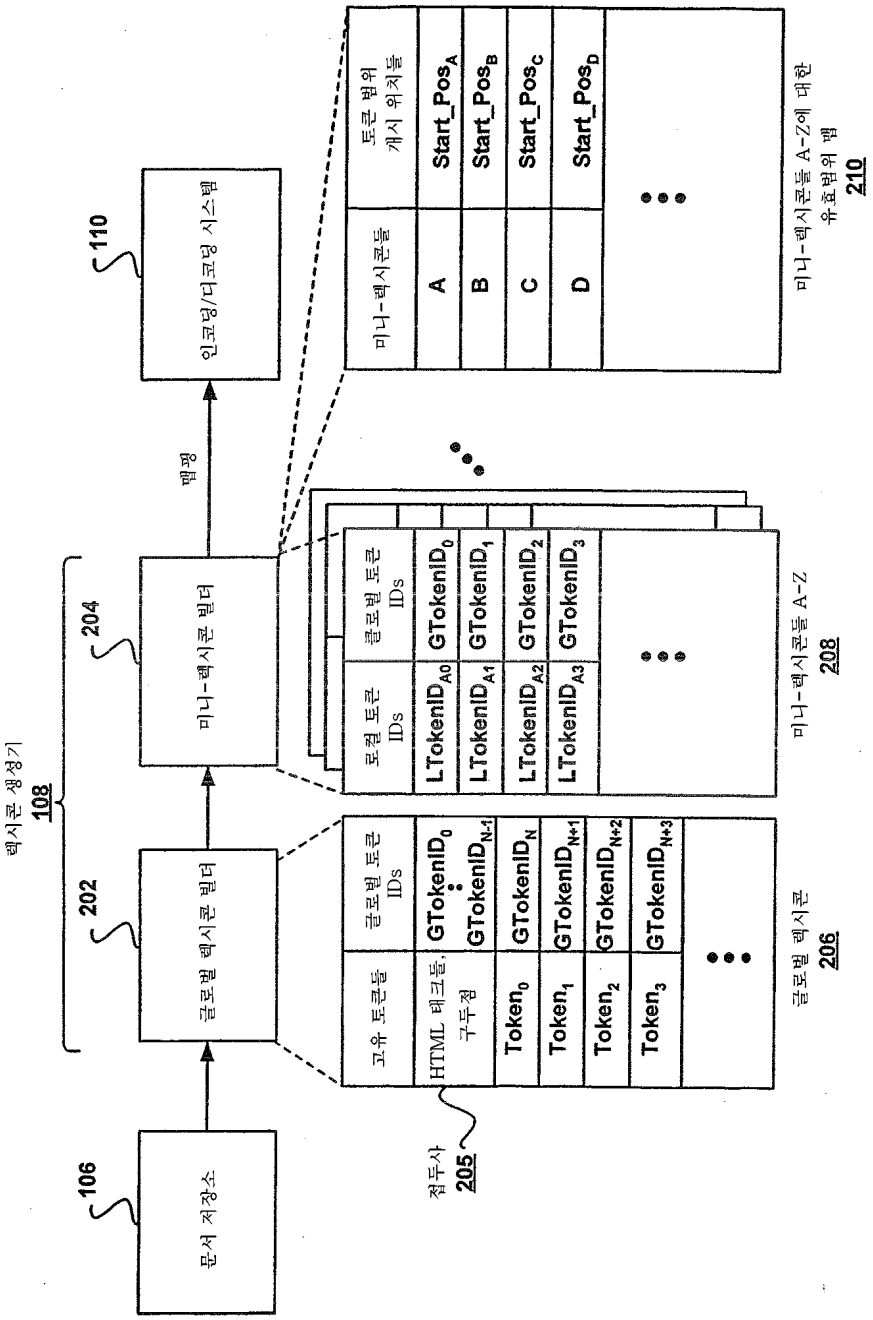
[0017] 동일한 참조 부호들은 몇몇 도면들을 통해 상응하는 부분들을 지칭한다.

도면

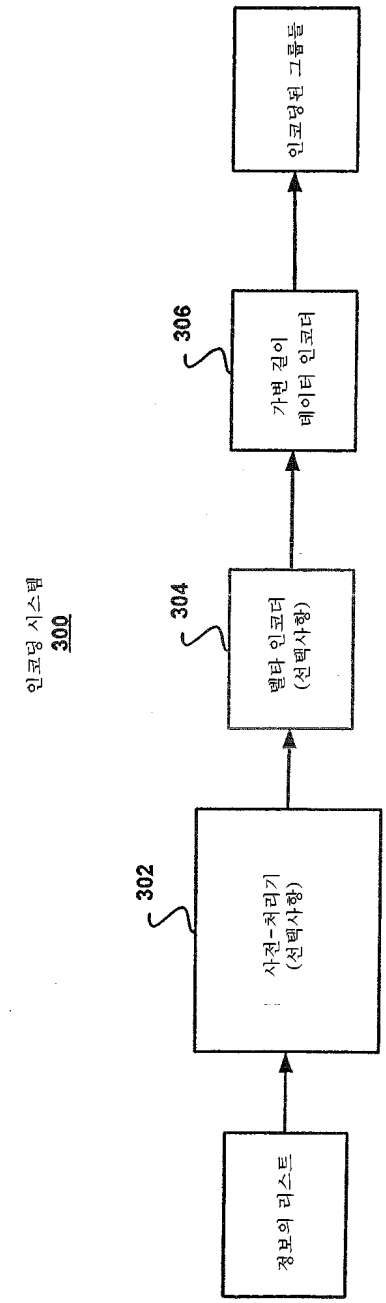
도면1


정보 검색 시스템
100
문서 처리 시스템
102
문서 저장소
106
렉시콘(Lexicon) 생성기
108
맵핑
인코딩/디코딩 시스템
110
인코딩된
토큰들
토큰 스페이스
저장소
112
질의 처리
시스템
104
질의 열(string)
결과들
질의(Query) 처리기(툴)
114
토큰 스페이스
인버스 인덱스
116

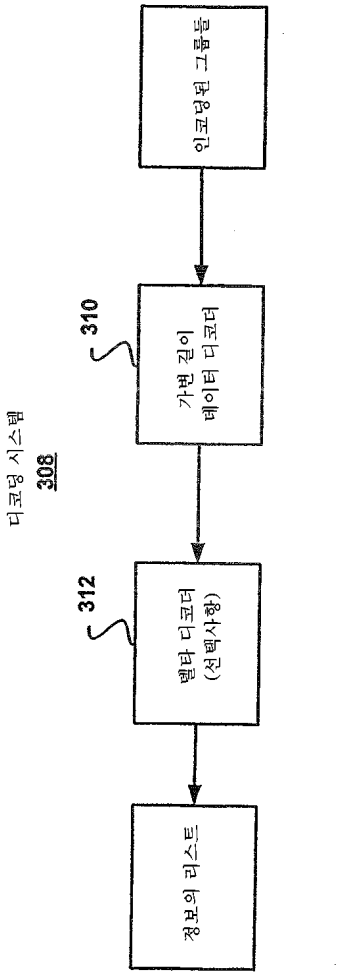
도면2


문서 저장소
106
렉시콘 생성기
108
글로벌 렉시콘 빌더
202
미니-렉시콘 빌더
204
맵핑
인코딩/디코딩 시스템
110
파서
205
글로벌 렉시콘
206
미니-렉시콘들 A-Z
208
미니-렉시콘들 A-Z에 대한 유효범위 맵
210
고유 토큰들
HTML 테크들, 구두점
글로벌 토큰 IDs
로컬 토큰 IDs
미니-렉시콘들
토큰 범위 개시 위치들
Token0
Token1
Token2
Token3
GTokenID0
GTokenID1
GTokenID2
GTokenID3
GTokenIDN-1
GTokenIDN
GTokenIDN+1
GTokenIDN+2
GTokenIDN+3
LTokenIDA0
LTokenIDA1
LTokenIDA2
LTokenIDA3
A
B
C
D
Start_PosA
Start_PosB
Start_PosC
Start_PosD

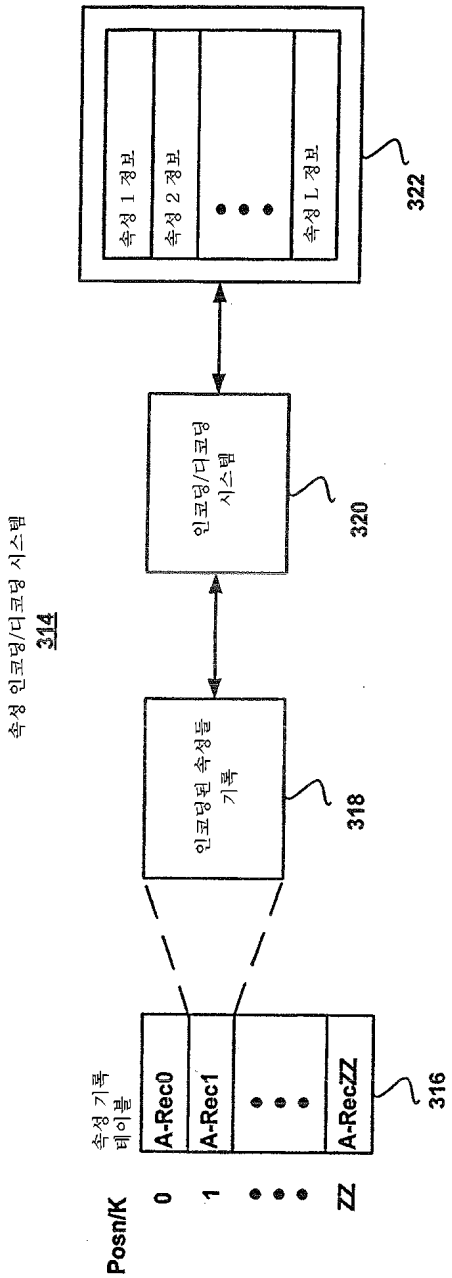
도면3A


인코딩 시스템
300
302
사전-처리기
(선택사항)
304
델타 인코더
(선택사항)
306
가변 길이
데이터 인코더
정보의 리스트
인코딩된 그룹들

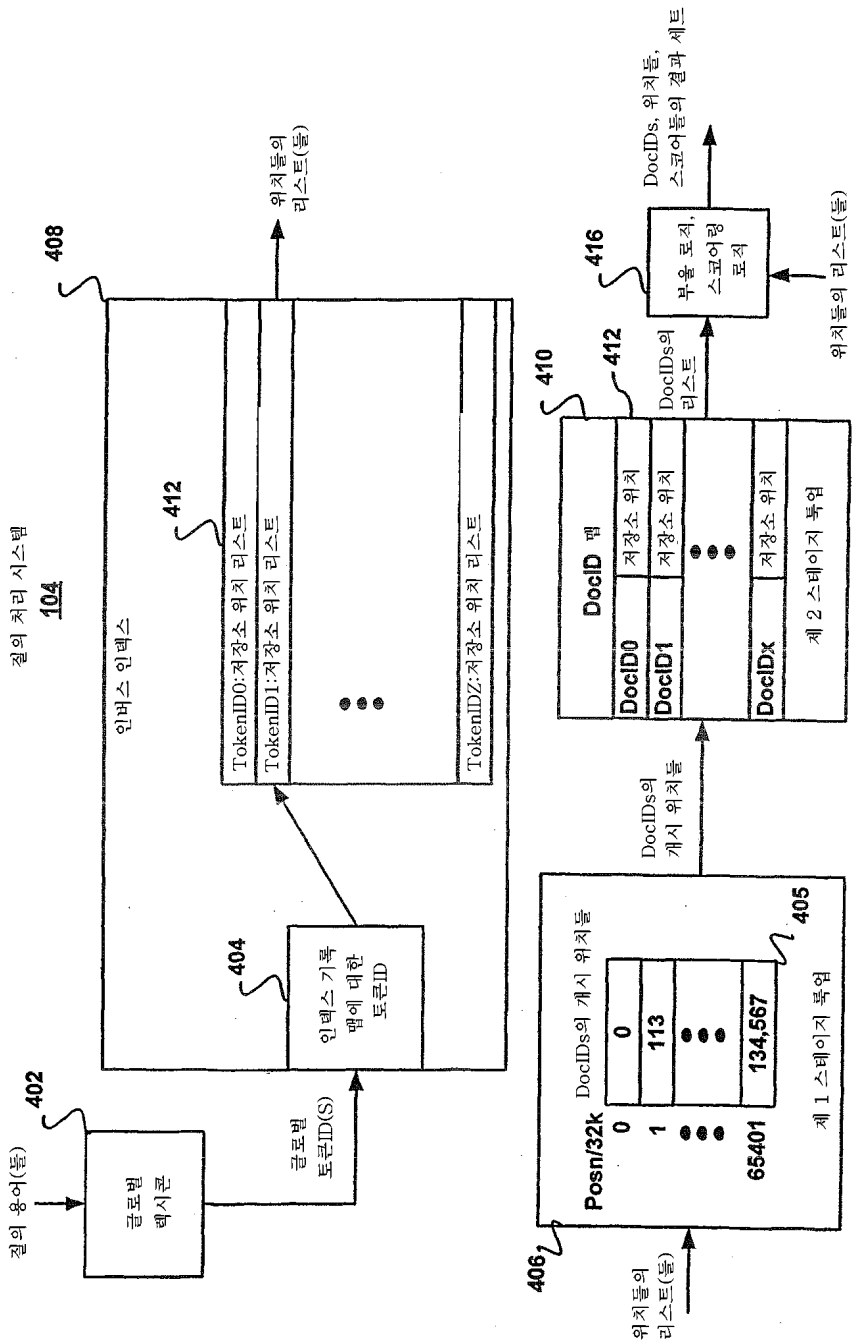
도면3B



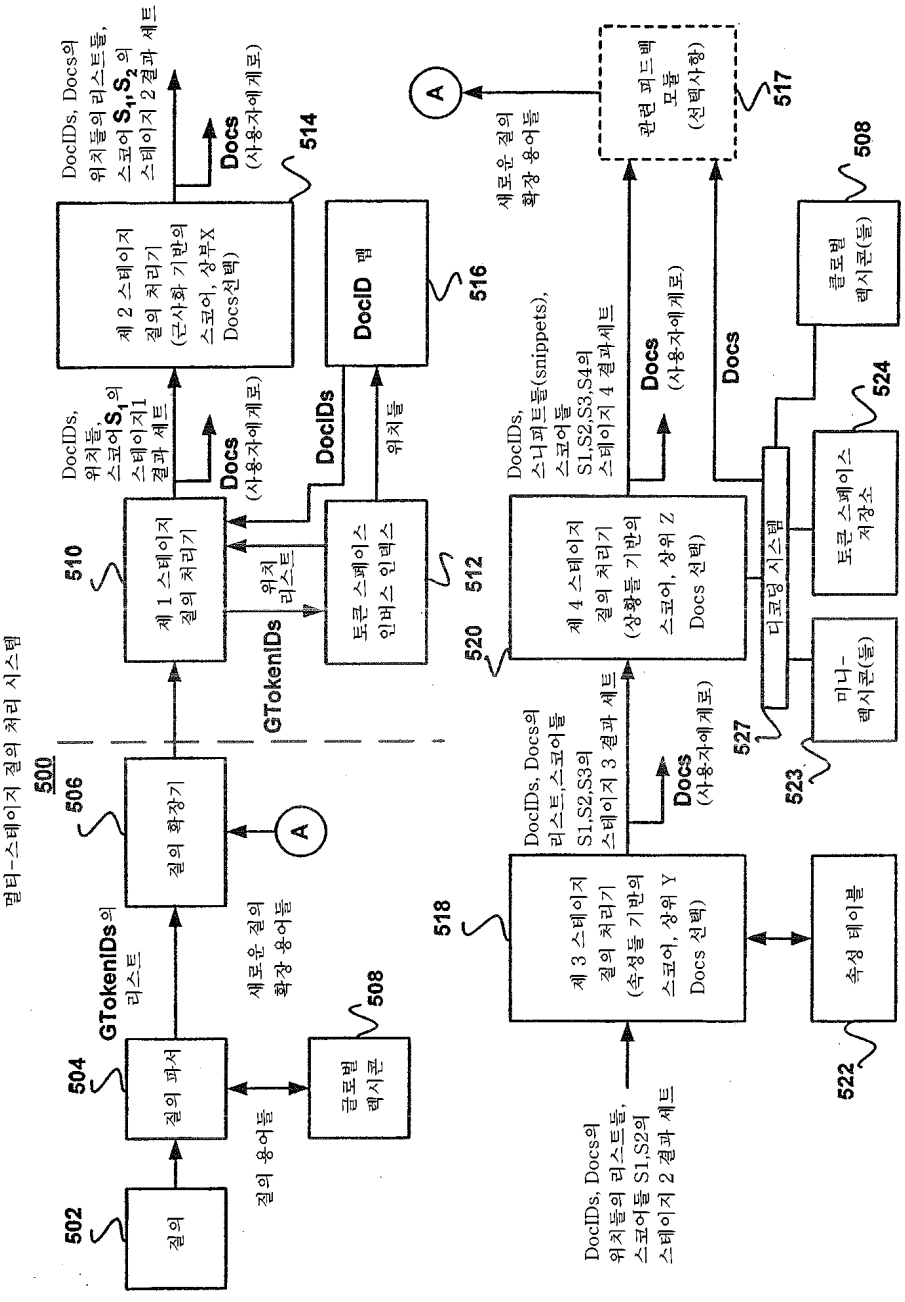
도면3C


속성 인코딩/디코딩 시스템
314
속성 1 정보
속성 2 정보
속성 L 정보
322
인코딩/디코딩 시스템
320
인코딩된 속성들 기록
318
속성 기록 테이블
A-Rec0
A-Rec1
A-RecZZ
316
Posn/K
0
1
ZZ

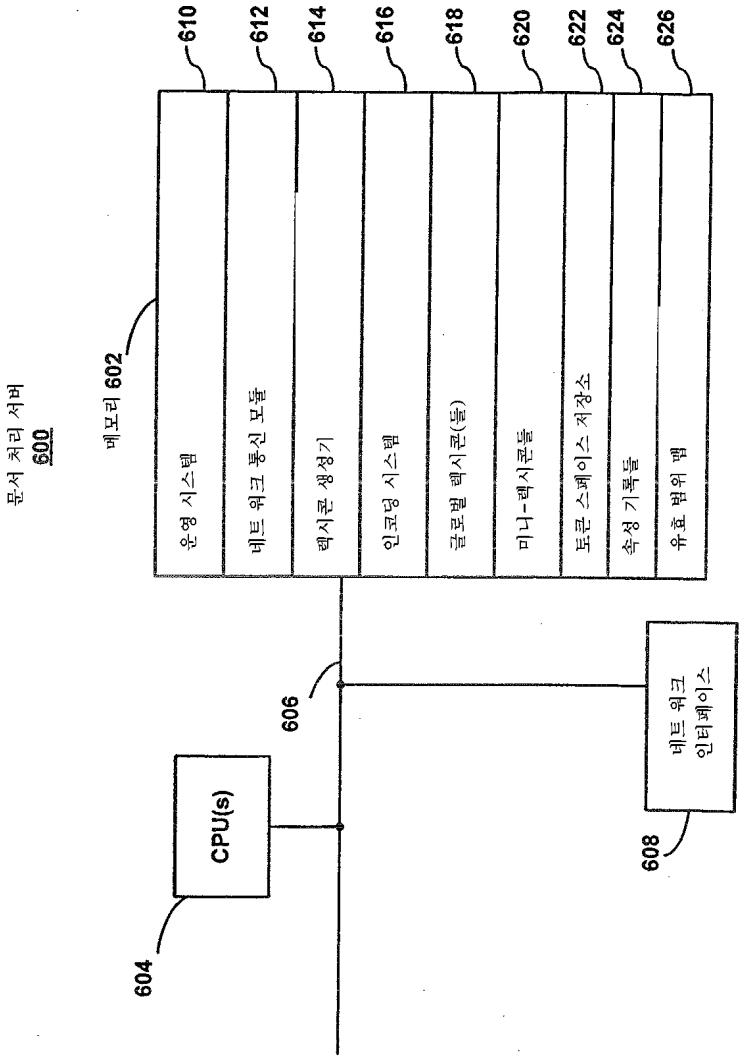
도면4


질의 처리 시스템
104
질의 용어(들)
402
글로벌 해시코
글로벌 토큰ID(S)
408
인버스 인덱스
412
TokenID0:저장소 위치 리스트
TokenID1:저장소 위치 리스트
TokenIDZ:저장소 위치 리스트
404
인덱스 기록 맵에 대한 토큰ID
위치들의 리스트(들)
406
Posn/32k
DocIDs의 캐시 위치들
0
1
65401
0
113
134,567
405
제 1 스테이지 룩업
위치들의 리스트(들)
DocIDs의 캐시 위치들
410
412
DocID 맵
DocID0
DocID1
DocIDx
저장소 위치
저장소 위치
저장소 위치
제 2 스테이지 룩업
DocIDs의 리스트
416
부울 로직, 스코어링 로직
위치들의 리스트(들)
DocIDs, 위치들, 스코어들의 결과 세트

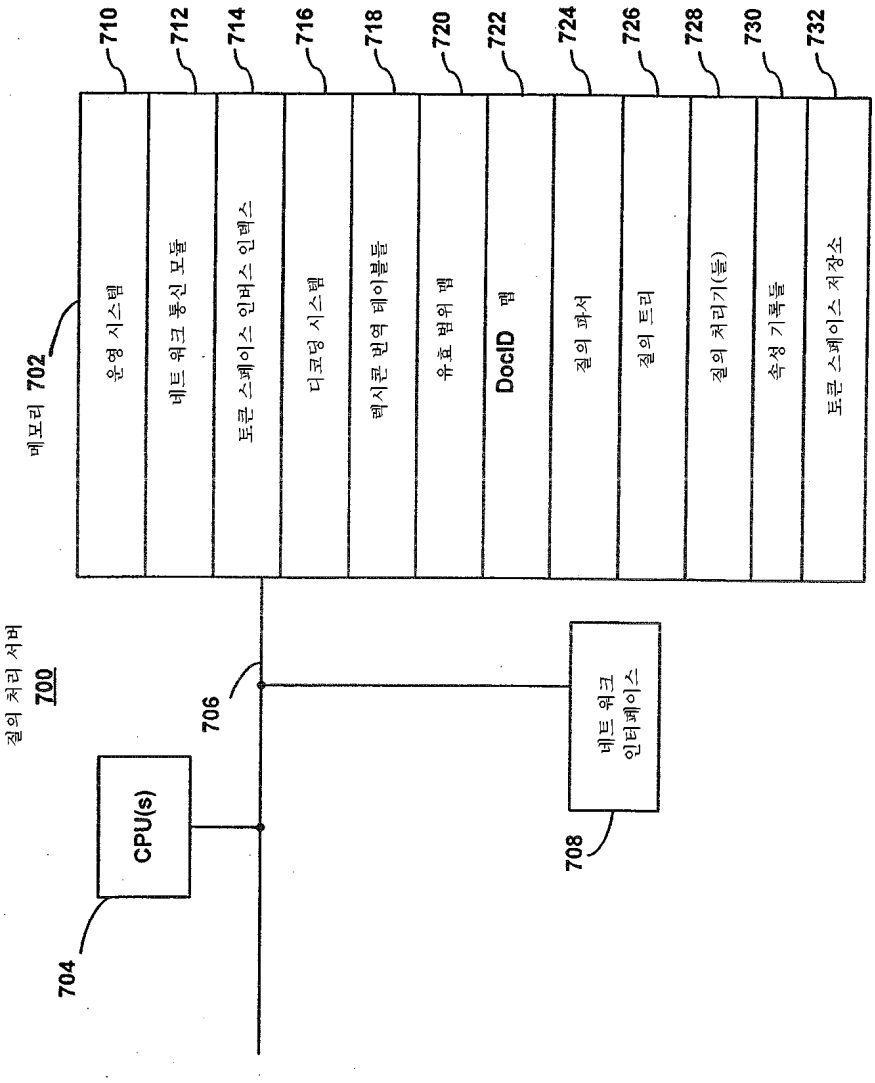
도면5


멀티-스테이지 질의 처리 시스템
500
질의
502
질의 파서
504
글로벌 렉시콘
508
질의 확장기
506
제 1 스테이지 질의 처리기
510
토큰 스페이스 인버스 인덱스
512
제 2 스테이지 질의 처리기 (근사화 기반의 스코어, 상부X Docs선택)
514
DocID 맵
516
제 3 스테이지 질의 처리기 (속성들 기반의 스코어, 상위 Y Docs 선택)
518
속성 테이블
522
제 4 스테이지 질의 처리기 (상황들 기반의 스코어, 상위 Z Docs 선택)
520
디코딩 시스템
527
미니-렉시콘(들)
523
토큰 스페이스 저장소
524
글로벌 렉시콘(들)
508
관련 피드백 모듈 (선택사항)
517
GTokenIDs의 리스트
GTokenIDs
위치 리스트
DocIDs
Docs (사용자에게로)
질의 용어들
새로운 질의 확장 용어들
A
DocIDs, Docs의 위치들의 리스트들, 스코어 S_1,S_2의 스테이지 2 결과 세트
DocIDs, 위치들, 스코어 S_1의 스테이지1 결과 세트
위치들
DocIDs, Docs의 리스트,스코어들 S1,S2,S3의 스테이지 3 결과 세트
DocIDs, 스니펫트들(snippets), 스코어들 S1,S2,S3,S4의 스테이지 4 결과세트
DocIDs, Docs의 위치들의 리스트들, 스코어들 S1,S2의 스테이지 2 결과 세트
새로운 질의 확장 용어들

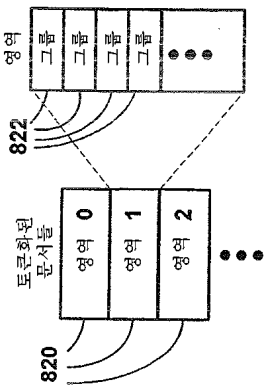
도면6


문서 처리 서버
600
메모리 602
운영 시스템
네트워크 통신 모듈
택시콘 생성기
인코딩 시스템
글로벌 택시콘(들)
미니-택시콘들
토큰 스페이스 저장소
속성 기록들
유효 범위 맵
610
612
614
616
618
620
622
624
626
CPU(s)
604
606
네트워크 인터페이스
608

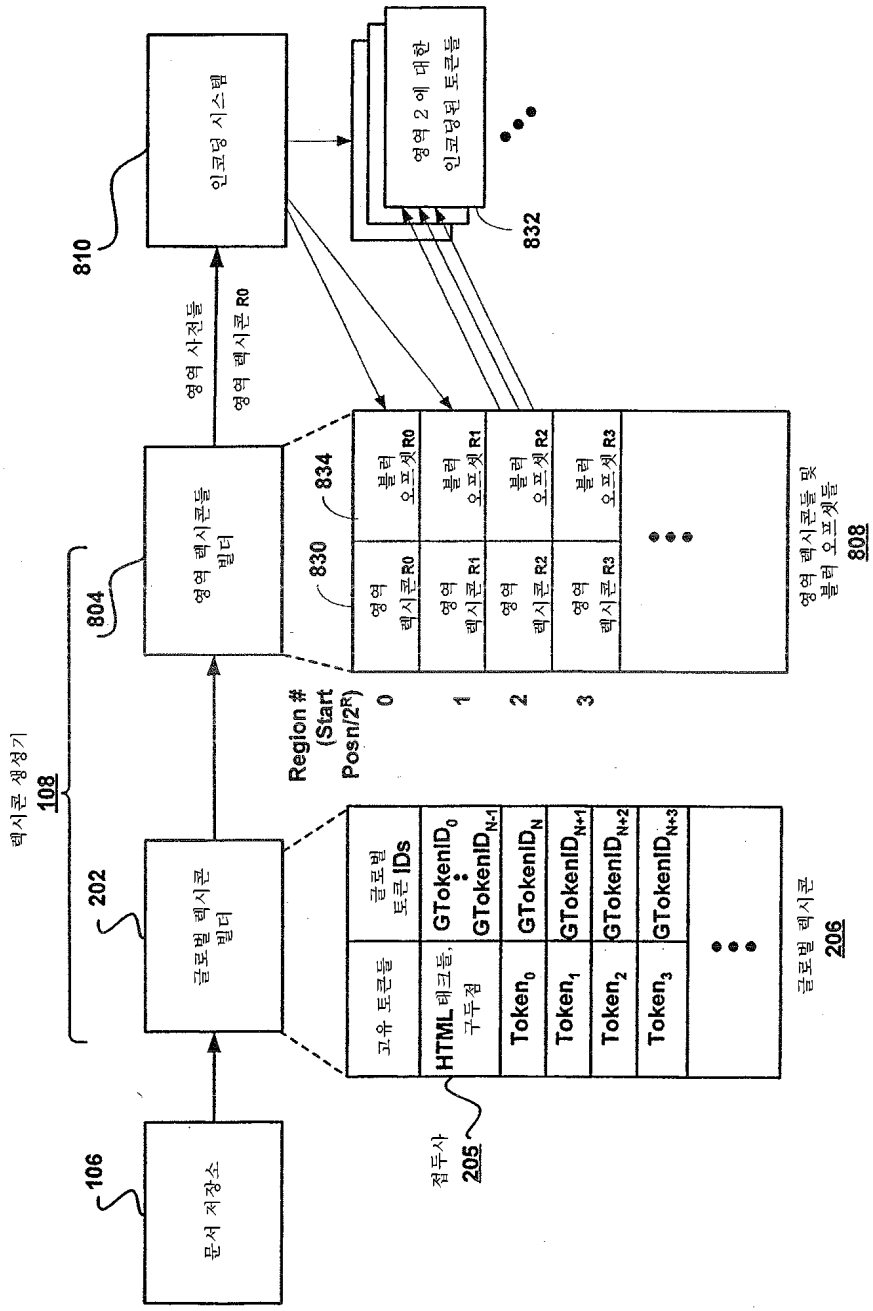
도면7


710
712
714
716
718
720
722
724
726
728
730
732
운영 시스템
네트워크 통신 모듈
토큰 스페이스 인버스 인덱스
디코딩 시스템
렉시콘 번역 테이블들
유효 범위 맵
DocID 맵
질의 파서
질의 트리
질의 처리기(들)
속성 기록들
토큰 스페이스 저장소
메모리 702
질의 처리 서버 700
706
CPU(s)
704
네트워크 인터페이스
708

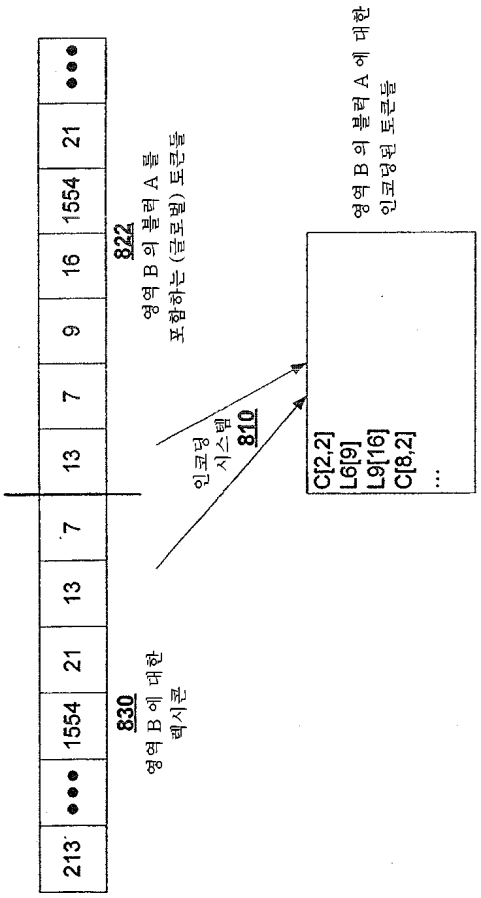
도면8A


영역
그룹
그룹
그룹
그룹
822
토큰화된 문서들
영역 0
영역 1
영역 2
820

도면8B


문서 저장소
106
글로벌 렉시콘 빌더
202
렉시콘 생성기
108
영역 렉시콘들 빌더
804
인코딩 시스템
810
영역 사전들
영역 렉시콘 R0
영역 2 에 대한 인코딩된 토큰들
832
Region # (Start Posn/2^P)
0
1
2
3
830
834
영역 렉시콘 R0
영역 렉시콘 R1
영역 렉시콘 R2
영역 렉시콘 R3
블럼 오프셋 R0
블럼 오프셋 R1
블럼 오프셋 R2
블럼 오프셋 R3
영역 렉시콘들 및 블럼 오프셋들
808
접두사
205
고유 토큰들
글로벌 토큰 IDs
HTML 태그들, 구두점
Token0
Token1
Token2
Token3
GTokenID0
GTokenIDN-1
GTokenIDN
GTokenIDN+1
GTokenIDN+2
GTokenIDN+3
글로벌 렉시콘
206

도면9A


213
•••
1554
21
13
7
13
7
9
16
1554
21
•••
830
영역 B 에 대한
렙시콘
822
영역 B 의 블럭 A 를
포함하는 (글로벌) 토큰들
인코딩
시스템
810
C[2,2]
L6[9]
L9[16]
C[8,2]
...
영역 B 의 블럭 A 에 대한
인코딩된 토큰들

도면9B