(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2009-0016361
(43) 공개일자 2009년02월13일

(51) Int. Cl.
G06F 17/30 (2006.01)
(21) 출원번호 10-2007-0080939
(22) 출원일자 2007년08월10일
심사청구일자 2007년08월10일

(71) 출원인
고려대학교 산학협력단
서울 성북구 안암동5가1 고려대학교 내
(72) 발명자
강상원
서울 성북구 종암동 I-Park 2차 207동 903호
김종완
서울 성북구 장위1동 232-70 (2층)
(뒷면에 계속)
(74) 대리인
현종철

전체 청구항 수 : 총 20 항

(54) 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을이용한 동적인 질의 처리 시스템

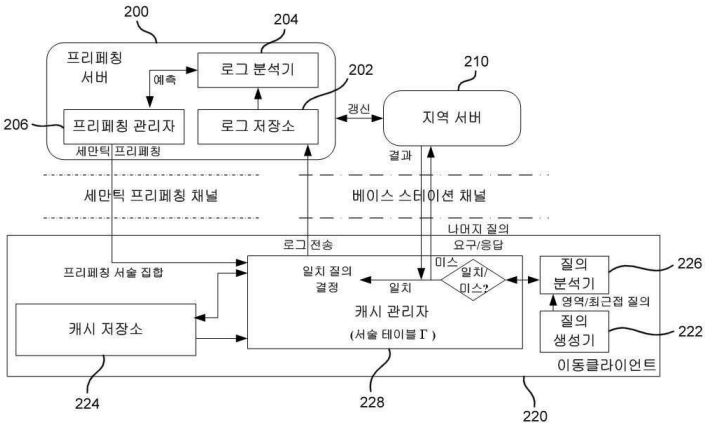
(57) 요 약

위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템이 개시된다.

본 발명에 따른 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템은, 동적인 질의 처리, 캐시 관리를 담당하는 이동 클라이언트, 및 세만틱 프리페칭의 관리 및 프리페칭 세그먼트의 집합인 프리페칭 서술을 상기 이동 클라이언트에게 전달하는 기능을 포함하는 서버를 포함한다.

본 발명에 의하면, 제한적인 대역폭을 사용하는 이동 컴퓨팅 환경에서 세만틱 프리페칭 기법을 이용하여 서버로부터 미리 수신하여 캐시에 저장해 놓은 세만틱 정보를 이용하여 사용자의 기호나 관심사항이 반영된 질의 처리를 수행하여, 이동 클라이언트의 캐시로부터 질의 결과를 얻을 수 있기 때문에 질의에 대한 결과를 구하는데 소요되는 지연 시간과 서버와의 통신 비용을 감소시키며, 개선된 캐시 교체 기법도 추가되어 위치 종속 데이터 환경에서 영역 질의와 최 근접 질의에 대해 높은 질의 절감률, 낮은 네트워크 트래픽 부하 및 향상된 캐시 성능과 효율적인 질의 처리를 제공하여 사용자의 편이성을 대폭 향상시킬 수 있는 효과가 있다.

대 표 도 - 도2



(72) 발명자

임석진

경기 이천시 창전동 416-2 옥당빌딩 2층

정하림

서울 성북구 안암동5가 고려대학교 자연계캠퍼스 아산이학관226호

특허청구의 범위

청구항 1

동적인 질의 처리와 캐시 관리를 담당하는 이동 클라이언트; 및

세만틱 프리페칭 관리와 프리페칭 세그먼트의 집합인 프리페칭 서술을 상기 이동 클라이언트에게 전달하는 기능을 포함하는 서버를 포함하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 2

제 1 항에 있어서,

상기 서버는,

상기 이동 클라이언트로부터 전송된 로그 파일을 저장하는 로그 저장소;

상기 로그 파일을 분석하는 로그 분석기; 및

상기 이동 클라이언트에게 전송되는 프리페칭 서술을 결정하는 프리페칭 관리자를 포함하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 3

제 2 항에 있어서,

상기 프리페칭 관리자는,

상기 이동 클라이언트가 세만틱 프리페칭 영역(Semantic Prefetching Area; SPA)으로 진입할 때마다 상기 이동 클라이언트의 식별자를 검사하며,

이전에 전송받은 로그 파일 내의 위치 종속 데이터(Location Dependent Data;LDD)의 질의 정보를 기반으로 하여, 상기 질의 정보에 대해 로그 파일 내의 질의 정보 및 질의 결과와 관련된 내용을 포함하는 프리페칭 세그먼트들의 합을 상기 이동 클라이언트가 이전에 방문한 세만틱 질의 정보와 상기 세만틱 질의 정보에 따른 질의 결과를 기술하는 프리페칭 서술로 결정하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 4

제 2 항에 있어서,

상기 이동 클라이언트는

세만틱 프리페칭 영역을 벗어날 경우, 상기 세만틱 프리페칭 영역에서 실행한 위치 종속 데이터의 질의 정보를 포함하는 로그 파일을 상기 서버에 전송하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 5

제 1 항에 있어서,

상기 이동 클라이언트는,

영역 질의(Range Query; RQ)와 최 근접 질의(Nearest Neighbor Query; NNQ)를 생성하는 질의 생성기;

캐시된 정보들이 저장되고 소정의 캐시 교체 전략에 따라 서술 테이블이 갱신되는 캐시 저장소;

상기 질의 생성기에서 생성된 질의의 종류를 분석하는 질의 분석기; 및

상기 질의 분석기에 의해 분석된 질의가 상기 캐시 저장소에 저장된 캐시의 내용과 일치하는지를 확인하여 일치 질의를 결정하고, 상기 서버에 나머지 질의를 재전송하는 캐시 관리자를 포함하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 6

제 5 항에 있어서,

상기 캐시 관리자는

상기 캐시 저장소에 저장되어 있는 세만틱 세그먼트들과 상기 서버로부터 얻은 프리페칭 세그먼트들을 서로 결합시켜 서술 테이블을 생성하고, 상기 생성된 서술 테이블을 상기 캐시 저장소에 전송하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 7

제 6 항에 있어서,

상기 캐시 저장소는

상기 이동 클라이언트가 최초로 상기 세만틱 프리페칭 영역으로 진입한 경우, 상기 이동 클라이언트가 상기 서버에 위치 종속 데이터 질의를 하여 얻은 위치 종속 데이터 질의 정보와 질의 결과들을 세만틱 세그먼트의 형태로 저장하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 8

제 5 항 내지 제 7 항 중 어느 한 항에 있어서,

상기 서술 테이블은

상기 이동 클라이언트가 이전에 방문했던 세만틱 프리페칭 영역으로 진입하는 경우 상기 서버로부터 수신한 프리페칭 서술을 기반으로 세만틱 세그먼트들의 갱신데이터를 포함하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 9

제 8 항에 있어서,

상기 서술 테이블은

가장 최근에 참조된 세만틱 세그먼트들만 유지하며,

상기 서술 테이블 내의 가장 오래된 접근 시간을 가지는 서술 테이블의 세만틱 세그먼트들은 프리페칭 서술로 교체되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 10

제 8 항에 있어서,

상기 서술 테이블은

참조 횟수가 가장 많은 세만틱 세그먼트들만 유지하며,

참조 횟수가 가장 적은 세만틱 세그먼트들은 프리페칭 서술로 교체되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 11

제 8 항에 있어서,

상기 서술 테이블의 모든 세만틱 세그먼트들은 하기의 식 1에 따라 상기 서술 테이블에 유지되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

$$Pref(Si) = \frac{\omega \times l + (1-\omega) \times f}{Q_{num}} \quad (1)$$

(f 와 l 은 각각 빈도와 지역성에 대한 값, 세만틱 세그먼트 Si 에 대하여, $Pref(S_i)$는 세만틱 세그먼트 Si 의 선호도, Q_{num} 은 질의 수를, ω ($0 \le \omega \le 1$)는 지역성에 대한 가중치)

청구항 12

제 5 항에 있어서,

상기 이동 클라이언트는

상기 캐시 저장소의 세만틱 캐시에 저장된 서술 테이블을 통해 질의 결과를 생성하고, 상기 서술 테이블을 통해 얻지 못한 결과들은 상기 서버에게 추가적으로 질의를 요구하여 질의 결과를 생성하는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 13

제 12 항에 있어서,

상기 추가적으로 생성된 질의에 따른 질의 결과는

상기 캐시 저장소의 서술 테이블에 추가되거나 또는 기존 세그먼트 정보를 갱신할 때에 사용되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 14

제 5 항에 있어서,

상기 질의 생성기의 영역 질의 유무는,

상기 캐시 저장소에 저장된 서술 테이블의 세만틱 세그먼트 영역들과 질의 영역의 겹침 정도에 따라 결정되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 15

제 14 항에 있어서,

상기 이동 클라이언트는,

상기 질의 영역과 상기 서술 테이블의 세그먼트 영역이 완전히 겹쳐서 질의 결과가 일치되는 경우 상기 캐시 저장소에 저장된 정보로부터 질의에 대한 응답을 구하고,

질의 결과의 일부가 서술 테이블과 일치되는 경우 상기 일치되는 부분에 대응하는 질의에 대하여 상기 캐시 저장소에 저장된 정보로부터 질의에 대한 응답을 구하고 일치하지 않는 부분에 대한 질의는 상기 서버로 전송하며,

상기 질의 영역과 세만틱 세그먼트 영역이 모두 겹치지 않는 경우 상기 질의 모두를 상기 서버로 전달하여 상기 질의에 대한 응답을 얻는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 16

제 15 항에 있어서,

상기 서버로 전달된 질의에 대한 응답은

상기 서술 테이블에 새로운 세만틱 세그먼트로 추가되어 저장되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 17

제 5 항에 있어서,

상기 질의 생성기의 최 근접 질의는,

상기 이동 클라이언트의 질의 지점이 상기 캐시 저장소의 서술 테이블의 세만틱 세그먼트들 내에 포함되는지의 여부에 따라 상기 서버로의 질의 유무가 결정되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 18

제 17 항에 있어서,

상기 질의에 따른 질의 결과는

상기 질의 지점이 상기 세만틱 세그먼트 내에 존재할 경우 질의 지점에서 가장 가까운 질의 조건을 만족하는 객체로 결정되며,

상기 질의 지점이 상기 서술 테이블의 세만틱 세그먼트 외부에 존재하면 소정의 연산을 통해 상기 질의 지점에서 가장 가까운 세만틱 세그먼트를 포함하는 최소 영역과 상기 서술 테이블의 세그먼트 영역과의 겹침 여부에 따라 상기 서버로의 질의 유무가 결정되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 19

제 18 항에 있어서,

상기 질의 지점이 상기 서술 테이블의 세만틱 세그먼트 외부에 존재하면 새로 설정된 상기 최소 영역은 상기 서술 테이블에 새로운 세만틱 세그먼트로 추가되어 저장되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

청구항 20

제5항 내지 제7항 또는 제12항 내지 제19항 중 어느 한 항에 있어서,

상기 영역 질의 및 최 근접 질의는

각각 또는 혼용하여 수행되는 것을 특징으로 하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템.

명 세 서

발명의 상세한 설명

기 술 분 야

<1> 본 발명은 무선 통신에 관한 것으로서, 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템에 관한 것이다.

<2> 무선 네트워크 환경에서 위치 기반 서비스 (Location based services; 이하 LBS라고 함)는 무선 단말의 위치 정보와 사용자의 위치에 따라 변하는 위치 종속 데이터(Location dependent data; 이하 LDD라고 함)를 이용하여 사용자에게 서비스를 제공한다.

배 경 기 술

<3> 무선 네트워크 환경에서 위치 기반 서비스는 응급 서비스, 여행자 가이드, 길 찾기 정보, 지역 지도 정보, 자녀의 위치 추적 서비스 등이 LDD 기반 LBS의 대표적인 예이다. 기본적으로 LBS는 사용자 프로파일 기반의 서비스를 제공한다. 사용자 프로파일에는 사용자의 식별자와 주소 같은 기본 정보뿐만 아니라, 사용자의 선호도, 이동 패턴, 질의 패턴과 같은 세만틱(semantic) 정보도 포함될 수 있다.

<4> 사용자 맞춤형 LBS를 위해 사용자 위치와 관련된 많은 연구가 이동 컴퓨팅 환경에서 수행되었다. 그러나, 종래의 연구는 이동 클라이언트가 공간 질의 (Spatial Query)를 하였을 때 서버가 이동 클라이언트의 위치에 따라 질의 결과를 제공해 주는 것에 초점을 두었기 때문에, 세만틱 정보를 이용하여 사용자의 기호까지 반영한 특화된 맞춤형 서비스를 제공하기 위한 연구는 미미하였다. 종래에 사용자의 세만틱 정보는 서버에서 유지되므로, 세만틱 정보 처리는 서버 측에서만 가능하였다. 이동 클라이언트에 의해 서버에 전송된 질의는 R-tree 색인, 검색, 모니터링 기법과 같은 서버 기반 질의 처리 방법을 이용하여 처리되었다. 질의 처리 관점에서 이러한 방식은 서버로부터 이동 클라이언트가 질의한 결과를 얻기 위해 많은 네트워크 부하를 일으키게 된다.

<5> LDD 환경에서 이동 클라이언트와 서버간의 데이터 전송은 제한된 통신 대역폭등으로 인해 과도한 네트워크 부하를 초래할 수 있다. 이동 클라이언트와 서버가 네트워크 부하를 줄이기 위해 부분적인 정보만을 교환한다면, 이동 클라이언트는 부정확한 정보를 유지하게 되고 결국 사용자 서비스 품질(QoS: Quality of Service)은 저하된다.

<6> 특히, 제한된 통신 대역폭은 서버가 이동 클라이언트에게 사용자가 관심 있어하는 LDD 정보를 원활히 전송시키지 못하게 할 수 있다. 이러한 문제점을 극복하기 위해, 프리페칭 기법, 방송 기법, 유연한 네트워크 구조 등이 효율적인 데이터 전송을 위해 연구되어왔으며, 또한 사용자가 관심 있는 정보의 효율적인 관리를 위해 다양한 캐시 교체 전략들이 연구되었다.

<7> 그러나, 상기와 같은 종래의 방법들은 사용자의 질의 패턴이나 특정 서비스에 대한 사용자의 기호나 관심사항이 반영된 질의 처리에는 한계를 지닌다는 문제가 있다.

발명의 내용

해결 하고자하는 과제

<8> 따라서, 본 발명이 해결하고자 하는 과제는, 사용자의 질의 패턴이나 특정 서비스에 대한 사용자의 기호나 관심사항이 반영된 질의 처리를 수행할 수 있는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템을 제공하는 것이다.

과제 해결수단

<9> 상기 과제를 해결하기 위하여 본 발명은,

<10> 동적인 질의 처리와 캐시 관리를 담당하는 이동 클라이언트 및 세만틱 프리페칭 관리와 프리페칭 세그먼트의 집합인 프리페칭 서술을 상기 이동 클라이언트에게 전달하는 기능을 포함하는 서버를 포함하는 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템을 제공한다.

<11> 그리고, 상기 서버는, 상기 이동 클라이언트로부터 전송된 로그 파일을 저장하는 로그 저장소, 상기 로그 파일을 분석하는 로그 분석기, 및 상기 이동 클라이언트에게 전송되는 프리페칭 서술을 결정하는 프리페칭 관리자를 포함하는 것을 특징으로 한다.

<12> 아울러, 상기 프리페칭 관리자는, 상기 이동 클라이언트가 세만틱 프리페칭 영역(Semantic Prefetching Area; SPA)으로 진입할 때마다 상기 이동 클라이언트의 식별자를 검사하며, 이전에 전송받은 로그 파일 내의 위치 종속 데이터(Location Dependent Data;LDD)의 질의 정보를 기반으로 하여, 상기 질의 정보에 대해 로그 파일 내의 질의 정보 및 질의 결과와 관련된 내용을 포함하는 프리페칭 세그먼트들의 합을 상기 이동 클라이언트가 이전에 방문한 세만틱 질의 정보와 상기 세만틱 질의 정보에 따른 질의 결과를 기술하는 프리페칭 서술로 결정하는 것을 특징으로 한다.

<13> 또한, 상기 이동 클라이언트는 세만틱 프리페칭 영역을 벗어날 경우, 상기 세만틱 프리페칭 영역에서 실행한 위치 종속 데이터의 질의 정보를 포함하는 로그 파일을 상기 서버에 전송하는 것을 특징으로 한다.

<14> 한편, 상기 이동 클라이언트는, 영역 질의(Range Query; RQ)와 최 근접 질의(Nearest Neighbor Query; NNQ)를 생성하는 질의 생성기, 캐시된 정보들이 저장되고 소정의 캐시 교체 전략에 따라 서술 테이블이 갱신되는 캐시 저장소, 상기 질의 생성기에서 생성된 질의의 종류를 분석하는 질의 분석기, 및 상기 질의 분석기에 의해 분석된 질의가 상기 캐시 저장소에 저장된 캐시의 내용과 일치하는지를 확인하여 일치 질의를 결정하고, 상기 서버에 나머지 질의를 재전송하는 캐시 관리자를 포함하는 것을 특징으로 한다.

<15> 그리고, 상기 캐시 관리자는 상기 캐시 저장소에 저장되어 있는 세만틱 세그먼트들과 상기 서버로부터 얻은 프

리페칭 세그먼트들을 서로 결합시켜 서술 테이블을 생성하고, 상기 생성된 서술 테이블을 상기 캐시 저장소에 전송할 수 있다.

<16> 또한, 상기 캐시 저장소는 상기 이동 클라이언트가 최초로 상기 세만틱 프리페칭 영역으로 진입한 경우, 상기 이동 클라이언트가 상기 서버에 위치 종속 데이터 질의를 하여 얻은 위치 종속 데이터 질의 정보와 질의 결과들을 세만틱 세그먼트의 형태로 저장할 수 있다.

<17> 한편, 상기 서술 테이블은 상기 이동 클라이언트가 이전에 방문했던 세만틱 프리페칭 영역으로 진입하는 경우 상기 서버로부터 수신한 프리페칭 서술을 기반으로 세만틱 세그먼트들의 갱신데이터를 포함할 수 있다.

<18> 더욱이, 상기 서술 테이블은 가장 최근에 참조된 세만틱 세그먼트들만 유지하며, 상기 서술 테이블 내의 가장 오래된 접근 시간을 가지는 서술 테이블의 세만틱 세그먼트들은 프리페칭 서술로 교체되는 것을 특징으로 한다.

<19> 그리고, 상기 서술 테이블은 참조 횟수가 가장 많은 세만틱 세그먼트들만 유지하며, 참조 횟수가 가장 적은 세만틱 세그먼트들은 프리페칭 서술로 교체되는 것을 특징으로 한다.

<20> 여기서, 상기 서술 테이블의 모든 세만틱 세그먼트들은 수학식

<21> $$Pref(Si) = \frac{\omega \times l + (1-\omega) \times f}{Q_{num}}$$ (f 와 l 은 각각 빈도와 지역성에 대한 값, 세만틱 세그먼트 Si 에 대하여, Q_{num} 은 질의 수를, ω ($0 \le \omega \le 1$)는 지역성에 대한 가중치)의 선호도에 따라 상기 서술 테이블에 유지되는 것을 특징으로 한다.

<22> 또한, 상기 이동 클라이언트는 상기 캐시 저장소의 세만틱 캐시에 저장된 서술 테이블을 통해 질의 결과를 생성하고, 상기 서술 테이블을 통해 얻지 못한 결과들은 상기 서버에게 추가적으로 질의를 요구하여 질의 결과를 추가적으로 생성하는 것을 특징으로 한다.

<23> 아울러, 상기 추가적으로 생성된 질의에 따른 질의 결과는 상기 캐시 저장소의 서술 테이블에 추가되거나 또는 기존 세그먼트 정보를 갱신할 때에 사용되는 것을 특징으로 한다.

<24> 그리고, 상기 질의 생성기의 영역 질의 유무는, 상기 캐시 저장소에 저장된 서술 테이블의 세만틱 세그먼트 영역들과 질의 영역의 겹침 정도에 따라 결정되는 것을 특징으로 한다.

<25> 또한, 상기 이동 클라이언트는, 상기 질의 영역과 상기 서술 테이블의 세그먼트 영역이 완전히 겹쳐서 질의 결과가 일치되는 경우 상기 캐시 저장소에 저장된 정보로부터 질의에 대한 응답을 구하고, 질의 결과의 일부가 서술 테이블과 일치되는 경우 상기 일치되는 부분에 대응하는 질의에 대하여 상기 캐시 저장소에 저장된 정보로부터 질의에 대한 응답을 구하고 일치하지 않는 부분에 대한 질의는 상기 서버로 전송하며, 상기 질의 영역과 세만틱 세그먼트 영역이 모두 겹치지 않는 경우 상기 질의 모두를 상기 서버로 전달하여 상기 질의에 대한 응답을 얻는 것을 특징으로 한다.

<26> 그리고, 상기 서버로 전달된 질의에 대한 응답은 상기 서술 테이블에 새로운 세만틱 세그먼트로 추가되어 저장되는 것을 특징으로 한다.

<27> 아울러, 상기 질의 생성기의 최 근접 질의는, 상기 이동 클라이언트의 질의 지점이 상기 캐시 저장소의 서술 테이블의 세만틱 세그먼트들 내에 포함되는지의 여부에 따라 상기 서버로의 질의 유무가 결정되는 것을 특징으로 한다.

<28> 한편, 상기 질의에 따른 질의 결과는 상기 질의 지점이 상기 세만틱 세그먼트 내에 존재할 경우 질의 지점에서 가장 가까운 질의 조건을 만족하는 객체로 결정되며, 상기 질의 지점이 상기 서술 테이블의 세만틱 세그먼트 외부에 존재하면 소정의 연산을 통해 상기 질의 지점에서 가장 가까운 세만틱 세그먼트를 포함하는 최소 영역과 상기 서술 테이블의 세그먼트 영역과의 겹침 여부에 따라 상기 서버로의 질의 유무가 결정되는 것을 특징으로 한다.

<29> 그리고, 상기 질의 지점이 상기 서술 테이블의 세만틱 세그먼트 외부에 존재하면 새로 설정된 상기 최소 영역은 상기 서술 테이블에 새로운 세만틱 세그먼트로 추가되어 저장되는 것을 특징으로 한다.

<30> 여기서, 상기 영역 질의 및 최 근접 질의는 각각 또는 혼용하여 수행되는 것을 특징으로 한다.

효 과

<31> 본 발명에 의하면, 제한적인 대역폭을 사용하는 이동 컴퓨팅 환경에서 세만틱 프리페칭 기법을 이용하여 서버로부터 미리 수신하여 캐시에 저장해 놓은 세만틱 정보를 이용하여 사용자의 기호나 관심사항이 반영된 질의 처리를 수행하여, 이동 클라이언트의 캐시로부터 질의 결과를 얻을 수 있기 때문에 질의에 대한 결과를 구하는데 소요되는 지연 시간과 서버와의 통신 비용을 감소시키며, 개선된 캐시 교체 기법도 추가되어 위치 종속 데이터 환경에서 영역 질의와 최 근접 질의에 대해 높은 질의 절감률, 낮은 네트워크 트래픽 부하 및 향상된 캐시 성능과 효율적인 질의 처리를 제공하여 사용자의 편이성을 대폭 향상시킬 수 있는 효과가 있다.

발명의 실시를 위한 구체적인 내용

<32> 이하에서 첨부된 도면을 참조하여 본 발명의 바람직한 실시예를 상세히 설명한다.

<33> 도 1은 본 발명에 따른 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템 구성의 블록도를 도시한 것이다.

<34> 이 시스템은, 동적인 질의 처리와 캐시 관리를 담당하는 이동 클라이언트(110) 및 세만틱 프리페칭 관리와 프리페칭 세그먼트의 집합인 프리페칭 서술을 이동 클라이언트(110)에게 전달하는 기능을 포함하는 서버(100)를 포함한다.

<35> 도 2는 도 1의 시스템의 상세한 구성의 예를 블록으로 도시한 것이다.

<36> 도 2를 참조하면, 프리페칭 서버(200)는 본 발명에 따른 기능이 포함된 서버이며, 지역 서버(210)는 프리페칭 서버(200)의 기능을 제외한 통상의 서버 기능을 포함하고 있다. 이하에서는 주로 프리페칭 서버(200)의 기능에 대해 상술하며, 이하에서는 서버라고 하는 것은 프리페칭 서버(220)를 의미하는 것으로 사용한다.

<37> 서버(200)는 이동 클라이언트(220)로부터 받은 로그 파일을 저장하는 로그 저장소(202), 상기 로그 파일을 분석하는 로그 분석기(204) 및 이동 클라이언트(220)에게 전달될 프리페칭 서술을 결정하는 프리페칭 관리자(206)를 포함한다.

<38> 그리고 이동 클라이언트(220)는, 영역 질의 (Range Query; RQ)와 최 근접 질의 (Nearest Neighbor Query; NNQ)를 생성하는 질의 생성기(222), 캐시된 정보들이 저장되고 소정의 캐시 교체 전략에 따라 서술 테이블이 갱신되는 캐시 저장소(224), 상기 질의 종류를 분석하는 질의 분석기(226) 및 질의 분석기(226)에 의해 분석된 질의가 캐시 저장소(224)에 저장된 캐시의 내용과 일치하는지를 확인하여 일치 질의를 결정하고, 서버(200)에 상기 캐시 저장소(224)에 저장된 캐시의 내용과 일치하지 않는 나머지 질의를 재전송하는 캐시 관리자(228)를 포함한다.

<39> 이동 클라이언트(220)가 고속 데이터 전송을 지원하는 세만틱 프리페칭 영역(Semantic Prefetching Area; 이하 SPA라고 함)에 진입했을 때, 서버로부터 미리 위치 종속 데이터(Location Dependent Data:LDD)에 대한 세만틱 정보를 얻도록 함으로써 이동 클라이언트(220)의 서버에 대한 빈번한 질의 요구를 줄이고 통신 부하를 감소시키는 본 발명에 따른 세만틱 프리페칭 (Semantic Prefetching; 이하 SP라고 함) 방법을 적용한다. 그리고 서버와의 통신 부하를 효율적으로 줄이기 위해서 기존 캐시 교체 전략들을 개선한 본 발명에 따른 캐시 교체 방법을 적용한다.

<40> 본 발명에 따른 세만틱 프리페칭 방법은 클라이언트-서버 모델을 기반으로 하며, SPA에서 이동 클라이언트(220)와 서버(200)는 다음과 같이 동작한다.

<41> 처음으로 동작을 시작한 이동 클라이언트(220)가 특정 SPA로 최초 진입하였을 때, 이동 클라이언트(220)의 캐시에는 데이터가 저장되어 있지 않다. SPA에 상주하는 동안 이동 클라이언트(220)는 서버에 필요할 때마다 LDD 질의를 수행하게 되고, 서버로부터 얻은 각 LDD 질의 정보와 질의 결과들은 세만틱 세그먼트(Semantic Segment)라고 불리는 형태로 세만틱 캐시에 저장된다.

<42> 현재 상주하는 SPA에서 다른 SPA로 이동할 때, 이동 클라이언트(220)는 상주하고 있는 SPA에서 수행한 모든 LDD 질의 정보를 담고 있는 로그 파일(Log file)을 서버(200)에게 기지국 채널(Base Station Channel:BSC)을 이용하여 전송한다. 이후 이동 클라이언트(220)는 새로운 SPA로 이동할 때마다 상기의 과정을 반복한다.

<43> 이동 클라이언트(220)가 예전에 방문했던 SPA를 재 방문하면, 서버는 이전에 전송 받았던 로그 파일내의 LDD 질의 정보를 분석하여, 각 질의 정보에 대해 프리페칭 세그먼트(Prefetching segment)라고 불리는 형태의 데이터

를 생성한다. 각 프리페칭 세그먼트는 로그 파일내의 질의 정보 및 질의 결과와 관련된 내용을 담고 있다. 서버에 의해 생성된 프리페칭 세그먼트 집합은, 예를 들면 고속의 다운링크 채널인 세만틱 프리페칭 채널(Semantic Prefetching Channel:SPC)과 같은 수단을 통해 이동 클라이언트(220)에게 전송된다.

<44> 이동 클라이언트(220)는 자신의 세만틱 캐시 저장소(224)에 저장되어 있는 세만틱 세그먼트들과 서버(200)로부터 얻은 프리페칭 세그먼트들을 서로 결합시켜 서술 테이블을 생성한 후 세만틱 캐시 저장소(224)에 저장한다. 현재 상황에 적합한 서술 테이블을 생성하고, 생성된 서술 테이블을 저장하기 위해 필요에 따라 캐시 교체 전략이 사용된다. 이후 이동 클라이언트(220)의 질의 결과는 먼저 세만틱 캐시 저장소(224)에 저장된 서술 테이블을 통해 얻게 되며, 서술 테이블을 통해 얻지 못한 결과들은 서버(200)에게 추가적으로 질의하여 얻게 된다.

<45> 서버(200)를 통해 추가적으로 얻은 질의 결과들은 캐시 저장소(224)의 서술 테이블에 반영되어 서술 테이블에 추가되거나 기존 세그먼트 정보를 갱신하게 된다.

<46> 종래의 세만틱 캐싱(Semantic Caching:SC) 기법에서는 이동 클라이언트(220)의 LDD 질의와 질의 결과들을 세만틱 캐시에 저장하였다. 이런 기법에서는 이동 클라이언트(220)가 이전에 방문한 SPA들에서 생성한 정보를 세만틱 캐시에 유지하기 때문에, 캐시 용량이 충분히 크지 않은 경우에는 빈번한 캐시 교체로 인해 새로이 재 진입한 SPA에서 필요한 내용과는 맞지 않는 LDD 관련 정보를 가지고 있을 확률이 높다. 이 경우 빈번한 캐시 실패로 이동 클라이언트(220)와 서버는 상호간에 빈번한 질의 요청과 질의 응답을 하게 되어 네트워크 부하를 가중시킨다. 또한, 질의 결과를 얻을 때까지 발생하는 지연 시간으로 즉각적인 서비스가 이루어지지 않아 QoS를 저하시키게 되는 문제가 있다.

<47> 도 2에서 프리페칭 세그먼트들의 효율적인 전송을 위해, 예를 들면 세만틱 프리페칭 채널(Semantic Prefetching Channel:SPC)이라는 고속의 전용채널이 사용된다고 가정한다. 기지국 채널(Base Station Channel:BSC)은 로그 정보 전송과 나머지 질의에 대한 요구와 응답을 위해 사용된다. 또한, 이동 클라이언트(220)는 적어도 한번 이상 각 SPA를 방문하여 프리페칭 서버에는 이동 클라이언트(220)의 로그 파일이 존재한다고 가정한다. 그리고, 무선 통신 상에서 발생할 수 있는 메시지 손실은 고려하지 않으며, 서버(200)는 이동 클라이언트(220)의 정보를 유지할 수 있는 충분한 용량과 처리 능력을 보유하고 있다고 가정한다.

<48> 한편, LDD 질의 처리를 위해서는 이동 클라이언트(220)와 LDD 데이터의 위치를 표현하는 방법이 필요하다. 본 발명에서는 정수 값을 가지는 2차원 공간 좌표를 이용하여 다음과 같이 위치를 표현한다.

<49> 공간 좌표를 구성하는 X 와 Y에 의해 형성된 그리드 영역 G = X × Y 에서, LDD 데이터 집합 D_f는 그리드 영역 G 에 존재한다. D_f의 각 구성요소와 사용자의 위치는 좌표 (L_x, L_y)로 표현되며, $L_x \in X$이고 $L_y \in Y$이다.

<50> 이동 클라이언트(220)는 SPA 영역을 벗어날 때, SPA에서 수행한 세만틱 질의 정보를 담고 있는 로그 파일 L_f를 서버(200)에 전송한다. 이동 클라이언트(220)가 SPA로 재 진입하면, 서버의 프리페칭 관리자(206)는 L_f를 분석하고 이동 클라이언트(220)에게 전달할 내용들을 결정한다. 이때, 각 내용들은 프리페칭 세그먼트 $F_i = < F_r, F_a, F_p, F_l, F_c >$의 형태로 표현되며 프리페칭 세그먼트의 집합인 프리페칭 서술 F_d가 이동 클라이언트(220)에게 전송된다.

<51> L_f로부터 생성된 프리페칭 서술 F_d는 이전 방문했을 때의 세만틱 질의 정보와 질의 결과를 담고 있다. 프리페칭 세그먼트 F_i의 F_r과 F_a는 내용의 관계와 속성, F_p는 F_d의 튜플들이 만족하는 SELECT 조건을 표현한다. F_l은 이동 클라이언트(220)의 위치를, F_c는 F_i의 실제 내용을 표현한다.

<52> $< F_r, F_a, F_p, F_l, F_c >$로 표현되는 F_i의 각 튜플은 하기의 수학식 1과 같이 정의된다.

수학식 1

<53> $F_r \in L_f$, $F_a \subseteq$ A (A는 L_f의 속성 집합),

<54> $F_p = P_1 \vee P_2 \vee \cdots \vee P_j \vee \cdots \vee P_m$ 이며 각 P_j 는 $P_j = b_{j1} \wedge b_{j2} \wedge \cdots b_{jk} \wedge \cdots \wedge b_{jn}$ 로 비교 수식 b_{jk}의 논리곱으로 표현된다. 비교 수식 b_{jk}는 서버의 로그 파일 L_f와 로그 파일의 속성 집합 A에서, $a\ op\ c$ 의 형태로 표현되는 수식이다. 여기서, $a \in$ A, $op \in \{\leq, <, \geq, >, =\}$, 그리고 c는 지정 영역에 대한 상수 값이다.

<55> F_l은 (L_x, L_y)로 이동 클라이언트(220)의 위치를 표현하며, F_c는 $F_c = \pi F_a \sigma F_p(F_r)$로 표현된다. 여기서 σ는 SELECT 연산자이고, π는 PROJECT 연산자를 의미한다. 프리페칭 세그먼트들은 비교 수식들을 포함하는 선택조건을 가진 SELECT-PROJECT 연산의 결과들임을 알 수 있다.

<56> 서버에서 생성된 프리페칭 서술 F_d는 내용과 색인으로 구성되며, 프리페칭 서술내의 모든 프리페칭 세그먼트 F_i는 하나 또는 여러 개의 연결 페이지에 저장된다.

<57> 예를 들어, 음식점과 와인 바에 대한 restaurant (Rtype, Rname) 과 wine bar (Wname, Wrank)이라는 두 가지 관계를 가지는 위치 종속 데이터베이스가 존재하고 사용자가 특정 SPA를 이전에 방문했을 때 하기과 같은 질의를 수행했다고 가정한다.

<58> 첫 번째로, 점심 시간에 위치 (32,40) 에서 이동 클라이언트(220)는 “3킬로미터 내에 있는 한정식 음식점의 이름을 모두 알려주세요.” 라는 질의 Q_1 을 수행하고,

<59> 두 번째로, 저녁 시간에 위치 (35,38) 에서 이동 클라이언트(220)는 “5킬로미터 내에서 가격이 50000원 이하인 와인을 판매하는 상점에 대한 정보와 상점 이름들을 알려주세요.” 라는 질의 Q_2를 수행한다.

<60> SPA를 재 방문하면, 이동 클라이언트(220)는 서버로부터 표 1과 같은 프리페칭 서술 색인과 질의 Q_1, Q_2의 결과를 수신한다. 하기의 표 1에서 F_c의 값 7, 11은 질의 Q_1, Q_2의 결과를 저장하고 있는 첫 번째 페이지를 나타낸다.

표 1

<61>

F	F_r	F_a	F_p	F_l	F_c
F_1	Restaurant	(Rname, Rtype)	(L_x -3 ≤ Rxposition ≤ L_x +3) ∧ (L_y -3 ≤ Ryposition ≤ L_y +3) ∧ Type=2	32,40	7
F_2	Wine bar	(Wname, Wrank)	(L_x -5 ≤ Wxposition ≤ L_x +5) ∧ (L_y -5 ≤ Wyposition ≤ L_y +5) ∧ (Price <50000)	35,38	11

<62> 여기서 타입(Type)을 지정하는데, 타입 1은 유럽식, 타입 2는 한정식, 타입 3은 일식, 타입 4는 중식, 타입 5는 멕시코식이라고 정하기로 한다.

<63> 그리고, 프리페칭 세그먼트가 표현된 형식과 유사하게 질의 Q와 세만틱 세그먼트 S_i로 표현될 수 있다. 질의 Q는 $< Q_r, Q_a, Q_p, Q_l, Q_c >$와 같이 표현하며 질의는 결과를 얻기 전이므로, $Q_c = \varnothing$이다. 세만틱 세그먼트 S_i는 $S_i =< S_r, S_a, S_p, S_l, S_c >$와 같이 표현한다. 세만틱 세그먼트의 집합인 세만틱 서술 S_d에는 이동 클라이언트(220)가 거쳐간 SPA 영역에서 질의했던 질의 관련 세만틱 정보와 질의 결과가 저장되며, 세만틱

서술 S_d도 프리페칭 서술과 유사하게 내용과 색인으로 구성된다. 프리페칭 세그먼트와 세만틱 세그먼트는 동일한 유형을 가지는 형태이지만, 서버와 이동 클라이언트(220)의 세만틱 정보에 대한 구별을 위해, 본 발명에서는 서버 측 세만틱 정보는 프리페칭 세그먼트라고 부르고, 이동 클라이언트(220) 측의 세만틱 정보는 세만틱 세그먼트라고 부른다.

<64> 서버로부터 수신한 F_d와 S_d의 효율적인 관리를 위해 모든 프리페칭 세그먼트와 세만틱 세그먼트들의 합으로 구성되는 서술 테이블 $\Gamma(\Gamma = A \cup B, A = \forall F_i, B = \forall S_i)$을 세만틱 캐시 저장소(224)에 저장한다. 서술 테이블 Γ가 생성되기 이전의 S_d내의 S_i와 F_i가 서로 동일한 형태를 가지면서 Γ에 포함되기 때문에, 본 발명에서는 이후 용어상의 혼돈이 없는 한 Γ의 원소를 세만틱 세그먼트라고 칭하기로 한다.

<65> 일반적으로, 이동 클라이언트(220)의 이동성이나 질의 유형과 같은 특성에 따라, 특정 LDD 질의의 결과 데이터들은 다른 LDD 질의의 결과들보다 빈번하게 사용될 수 있다. 이런 결과들을 일정 기간 동안 이동 클라이언트(220)의 세만틱 캐시(혹은 캐시 저장소)에 유지하는 것은 캐시의 효율성을 높이는데 도움이 된다.

<66> 또한, 질의 Q에 대해 이동 클라이언트(220)가 Γ로부터 그 결과를 얻을 수 있다면, 서버(200)와의 통신 비용 없이 빠른 응답을 사용자에게 제공할 수 있으므로, 효율적인 Γ의 구성은 매우 중요하다.

<67> 그러므로, 본 발명에서는 Γ의 각 세만틱 세그먼트에 서술 매개변수의 집합, $\kappa = \{t, f, l\}$을 추가한다. 여기서, t, f, l은 각각 접근 시간, 빈도, 지역성을 각각 나타낸다.

<68> Γ내의 세만틱 세그먼트 $S_i' = < S'_r, S'_a, S'_p, S'_l, S'_c >$와 질의 $Q = < Q_r, Q_a, Q_p, Q_l, Q_c >$가 $S'_r = Q_r$, $S'_a = Q_a$, 이며, $S'_p = Q_p$의 관계라면, S_i'과 Q는 관계와 속성이 일치하고, Q의 질의 영역과 S_i'의 세만틱 영역도 완전히 일치하여 전체적으로 겹침을 의미한다. 이러한 관계를 동등한(equivalent)한 관계라고 부른다.

<69> Q의 질의 결과는 질의 정돈(Query trimming) 처리에 의해 일치 질의(prove query)와 나머지 질의(remainder query)라고 불리는 두 가지 종류로 나뉘어진다. 일치 질의는 S_i'에 의해서 만족되는 Q의 질의 결과 부분을 나타내며, 나머지 질의는 S_i'에서 찾을 수 없는 질의 결과 부분을 나타낸다.

<70> Q와 S_i'가 Equivalent한 관계일 때에는 Q는 S_i'에서 찾을 수 없는 질의 결과 부분을 나타낸다. Q와 S_i'가 Equivalent한 관계일 때에는 Q는 S_i'로부터 질의 결과를 모두 얻을 수 있기 때문에 질의 결과는 일치 질의이다.

<71> Q가 S_i'로부터 일부의 질의 결과를 얻는 경우에는 연관되는(Related) 관계가 있다고 하며, Q의 질의 결과는 일치 질의와 나머지 질의로 나누어진다. Q와 S_i'가 Related 관계일 경우에는 나머지 질의 처리를 위해 추가적인 통신 비용이 발생한다.

<72> 이하에서는 세만틱 프레페칭 기법이 적용된 환경에서, 본 발명에 따른 이동 클라이언트(220)의 영역 질의(range query)와 최 근접 질의(nearest neighbor query)가 처리되는 세부적인 질의 처리 과정과 캐시 교체 전략을 상술하기로 한다.

<73> 영역 질의(range query)에 대한 세부적인 질의 처리과정은 다음과 같다. 도 3은 영역 질의의 예를 나타낸다.

<74> 도 3을 참조하면, 각 사각형 영역은 서술 테이블 Γ의 세만틱 세그먼트 영역들과 질의 영역의 겹침 정도에 의존

적이다. 겹침 정도에 따라, 질의 Q의 처리는 세 가지 경우로 나누어질 수 있다.

<75> 첫 번째는, 질의 결과 모두를 이동 클라이언트의 Γ로부터 얻을 수 있는 경우이다. 즉, 질의 영역과 세만틱 세그먼트 영역이 완전히 겹치고 질의 결과가 일치 질의인 경우이다.

<76> 두 번째는, 질의 결과의 일부만을 Γ로부터 얻을 수 있는 경우이다. 즉, 도 3과 같이 질의 Q의 질의 영역이 세만틱 세그먼트 S_2, S_3, 그리고, S_5의 영역들과 부분적으로 겹치는 경우이다. 본 발명에서는 편의상 S_3와 겹치는 질의 영역에 대해서만 상술하기로 한다. 이 경우에 질의 Q의 질의 영역은 Γ로부터 질의 결과를 얻을 수 있는 일치 질의 영역과 서버에게 나머지 질의를 하여 추가적인 결과를 얻을 수 있는 나머지 질의 영역으로 나누어진다. 완전한 질의 결과는 이 두 영역으로부터 얻을 수 있다.

<77> 세 번째는, 질의 결과를 Γ로부터 전혀 얻을 수 없는 경우이다. 즉, 질의 영역과 세만틱 세그먼트 영역이 전혀 겹치지 않는 경우이다. 이 경우에는 질의 Q가 서버에 전달되어야만 질의 결과들을 얻을 수 있다.

<78> 도 3에서, 이동 클라이언트는 객체 c, f, g의 정보를 세만틱 캐시로부터 얻을 수 있지만, 객체 d의 정보는 나머지 질의를 수행하여야만 얻을 수 있다. 질의 Q에 대한 영역 질의 결과는 객체 c, d, f, g의 세만틱 정보가 되며, 질의 결과는 새로운 세만틱 세그먼트 S_{new}로 서술 테이블 Γ에 추가된다. 이러한 방법을 사용하여 유사 질의 발생시 이동 클라이언트의 세만틱 캐시 유용성을 높일 수 있다.

<79> 앞에서 설명한 것처럼 Γ내의 각 세만틱 세그먼트들은 서술 매개변수 집합 κ를 포함한다. 세만틱 세그먼트 영역과 질의 영역의 겹침의 정도에 따라서, κ의 각 매개변수는 다음과 같이 갱신된다.

<80> 만일, 질의 Q가 Γ내의 세만틱 세그먼트와 겹칠 때, 겹친 세만틱 세그먼트에 대한 접근 시간은 질의한 시간으로 교체된다. 그리고, 겹친 세만틱 세그먼트의 빈도 값은 1씩 증가한다. 마지막으로, 나머지 질의 영역 크기에 대한 일치 질의 영역 크기의 비율을 나타내는 지역성은 하기의 수학식 2와 같이 갱신된다.

수학식 2

<81>
$$l = l' + \frac{M}{M+R}$$

<82> 여기서 l'는 지금까지 유지된 지역성의 값, M은 질의 영역과 세만틱 세그먼트 영역 간에 겹친 일치 질의 영역의 크기, R은 질의 영역과 세만틱 세그먼트 영역 간에 겹치지 않은 나머지 질의 영역의 크기를 의미한다.

<83> 이하에서는 최 근접 질의(NNQ)에 대한 세부적인 질의 처리 절차를 상술하기로 한다. 최 근접 질의는 영역 질의와 함께 공간 데이터 처리를 위해 주로 사용된다.

<84> 도 4는 최 근접 질의 처리의 일 예를 도시한 것이다.

<85> 도 4를 참조하면, 사각형 영역은 Γ의 세만틱 세그먼트들의 영역을 나타내며, a부터 h까지 점 정보로 표현된 LDD 객체들을 포함한다. 영역 질의와 달리, 최 근접 질의 Q의 영역은 작은 점으로 표현되며 질의를 수행한 위치를 의미한다. Q에서 시작된 두 점선은 질의와 세만틱 세그먼트 영역 간의 최소 거리를 나타낸다.

<86> 최 근접 질의의 결과는 이동 클라이언트의 질의 지점이 세만틱 세그먼트들 내에 포함되는지의 여부에 따라 달라진다. 만일 질의 지점이 세만틱 세그먼트 내에 존재하면, 질의 결과는 질의 지점에서 가장 가까운 질의 조건을

만족하는 객체로 결정된다. 도 4에 도시된 것과 같이 질의 지점이 세만틱 세그먼트 외부에 존재하면, MINDIST 연산을 사용하여 먼저 질의 점에서 가장 가까운 세만틱 세그먼트인 S_3가 선택된다.

<87> 가장 가까운 세만틱 세그먼트가 선택된 후에, 이동 클라이언트는 질의 지점과 선택된 세만틱 세그먼트의 영역을 포함하는 최소 사각형 영역을 새로운 세만틱 세그먼트 S_{new}로 생성한다. 이 때, S_{new}의 확장 부분(410)은 영역 질의 처리의 나머지 질의처럼 서버에 요청하여 얻어진다. 이동 클라이언트는 S_{new}에서 네 개의 객체 c, d, f, g를 얻게 되며, 최종적으로 S_{new}로부터 질의 지점에서 가장 가까운 객체인 f를 얻게 된다. 이 S_{new}는 영역 질의에서처럼 서술 테이블 Γ에 추가된다.

<88> 최 근접 질의도 영역 질의처럼 각 세만틱 세그먼트들은 서술 매개변수 집합 K를 가진다. 접근 시간과 빈도에 대한 갱신 방법은 영역 질의에서와 동일하다. 그러나, 최 근접 질의에서 지역성에 대한 갱신 방법은 영역 질의와는 다르다. 최 근접 질의에서 M은 질의 지점에서 MINDIST 연산에 의해 선택된 가장 가까운 세만틱 세그먼트 영역, R은 새로운 세만틱 세그먼트 S_{new}에서 M을 제외한 영역을 각각 표현한다.

<89> 이전에 방문했던 SPA로 진입할 때마다 이동 클라이언트는 서버로부터 전송 받은 프리페칭 서술 F_d를 이용하여 서술 테이블 Γ를 갱신하기 때문에 이동 클라이언트의 현재 상황에 적합한 서술 테이블의 구성을 위해서는 효율적인 캐시 교체 전략이 필요하다. 본 발명에서는 세 가지의 캐시 교체 전략을 사용한다. 이와 같은 캐시 교체는 캐시 관리자에 의해 실행될 수 있다.

<90> 첫 번째의 캐시 교체 전략은 가장 최근에 참조된 세만틱 정보(Semantic least recently used:S-LRU)로서, 기존 LRU 전략의 변형으로, 서술 테이블 Γ에는 가장 최근에 참조된 세만틱 세그먼트들만 유지한다. Γ 내에서 가장 오래된 접근 시간을 가지는 Γ의 세만틱 세그먼트들이 F_d로 교체된다.

<91> 한편, 두 번째의 캐시 교체 전략은 가장 빈번하게 사용된 세만틱 정보(Semantic least frequently used:S-LFU)로서, 이는 기존 LFU 전략의 변형으로 Γ에는 빈번히 참조된 세만틱 세그먼트들만 유지한다. Γ에서 가장 참조되지 않은 세만틱 세그먼트들이 F_d로 교체된다. 그러나, 세만틱 세그먼트들이 F_d보다 높은 빈도를 가지면 교체하지 않을 수 있다.

<92> S-LRU와 S-LFU 전략들은 캐시 교체 처리시 SPA의 지역성을 고려하지 못한다. 그러므로, 본 발명에서는 SPA에서 세만틱 세그먼트의 빈도와 지역성을 동시에 고려할 수 있는 선호도 우선 교체 전략(Preference Priority Replacement:PPR)을 제안한다.

<93> 세번째는 상술한 선호도 우선 교체 전략(Preference Priority Replacement:PPR)으로서, 이는 SPA에서 공간 지역성과 빈도수를 고려한 캐시 교체 전략으로 Γ의 모든 세만틱 세그먼트들은 하기의 수학식 3과 같은 선호도를 유지한다.

수학식 3

<94>
$$Pref(S_i) = \frac{\omega \times l + (1-\omega) \times f}{Q_{num}}$$

<95> 상기 수학식 3에서 f와 l은 빈도와 지역성에 대한 값을 각각 나타낸다. 세만틱 세그먼트 S_i에 대하여, Q_{num}은 질의 수를, $\omega\ (0 \le \omega \le 1)$는 지역성에 대한 가중치를 표현한다.

<96> 빈도 f 와 지역성 l 만으로도 클라이언트의 시간적 공간적 특성을 각각 고려할 수 있기 때문에, 접근 시간 t 는 PPR에서는 사용되지 않는다.

<97> 본 발명에 따른 세만틱 프리페칭 기법의 알고리즘은 크게 서버와 이동 클라이언트에서 동작하는 알고리즘으로 나눌 수 있다. 이동 클라이언트의 알고리즘은 질의의 종류에 따라서 영역 질의 처리와 최 근접 질의 처리 알고리즘으로 나누어진다.

<98> 도 5는 이동 클라이언트에게 보낼 프리페칭 서술 F_d 를 결정할 수 있는 서버(200) 알고리즘을 보여준다. 이동 클라이언트가 SPA로 진입할 때마다 서버는 먼저 이동 클라이언트의 식별자를 검사한 후 로그 파일 L_f 를 이용하여 프리페칭 세그먼트들의 합 $\bigcup_{i \in I} F_i$ 를 F_d 로 결정한다. 이동 클라이언트가 SPA를 벗어날 때 서버로 보낸 로그 메시지 log_m 을 이용하여 서버는 로그 파일 L_f 를 갱신한다.

<99> 도 6은 이동 클라이언트의 세만틱 영역의 질의 처리 알고리즘을 도시한 것이다.

<100> 도 6을 참조하면, 영역 질의 Q_i 는 서술 테이블의 세만틱 세그먼트들과 비교되어 세 가지 경우에 따라 서로 다른 처리를 진행한다. 질의 영역과 세만틱 세그먼트 영역의 겹침 정도에 따라 영역 질의 Q_i 의 결과는 일치 질의 Q_{probe} 와 나머지 질의인 $Q_{remainder}$ 로 나누어 진다. Q_{probe} 영역은 R_{buffer} 에 저장되며, 서버로 요청된 $Q_{remainder}$ 의 결과도 계속해서 R_{buffer} 에 저장된다. 이것은 Q_i 의 결과로 서술 테이블에 S_{new} 로 저장되며 S_{new} 내에서 질의 결과인 R_i 를 얻을 수 있다.

<101> 도 7은 세만틱 최 근접 질의를 처리하기 위한 이동 클라이언트에서 사용되는 알고리즘을 나타낸다. 최 근접 질의 Q_i 가 요청되면 이동 클라이언트는 먼저 Γ 의 세만틱 세그먼트들 영역에 질의 지점이 포함되는지를 확인한다. 특정 세만틱 세그먼트에 포함된다면 세만틱 세그먼트 영역내의 객체 중 질의 조건을 만족하는 최단 거리의 최 근접 객체를 찾는다. 그렇지 않다면, 질의 지점과 Γ 의 세만틱 세그먼트 중 가장 가까운 세만틱 세그먼트들을 찾아 우선순위 큐에 저장한다. 우선순위 큐의 원소를 하나씩 꺼내어 질의 지점을 포함하는 확장된 영역의 세만틱 세그먼트를 생성한 후, 생성된 세만틱 세그먼트 영역 내의 객체 중에서 질의 조건을 만족하는 최단 거리의 최 근접 객체를 갖는다.

<102> 상기 상술한 본 발명에 따른 세만틱 프리페칭(Semantic Prefetching:SP) 방법은 이동 클라이언트가 SPA에 진입하고 나갈 때 종래의 세만틱 캐싱 방식에 비해 추가적인 네트워크 트래픽을 발생시킨다. 그러나, SPA로 진입할 때 서버로부터 미리 사용자의 기호나 관심 사항이 반영된 LDD 정보를 제공받기 때문에, 캐시 실패를 감소시킬 수 있어 네트워크 부하를 감소시키고 캐시 정보를 이용한 즉각적인 서비스도 제공해 주어 향상된 QoS를 제공할 수 있다.

<103> 본 발명에 따른 시스템을 사용할 때의 캐시 적중률의 변형인 질의 절감율(Query Saving Ratio:QSR)과 네트워크 부하를 시뮬레이션을 통하여 검증한 결과 종래의 경우에 비하여 더 향상된 성능을 확인할 수 있다.

<104> 또한 상기 상술한 영역 질의 및 최 근접 질의는 각각 별도로 실시할 수도 있고 또는 혼용하여 실시 가능하며, 사용자의 이동성과 질의 유형을 고려하여 영역 질의와 최 근접 질의 처리 방법을 혼용하여 사용하는 경우 시뮬레이션의 결과 특정 질의 처리 방법 하나만을 사용하는 것보다 더 효율적인 것을 알 수 있다.

<105> 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 본 발명에 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 변형된 형태로 구현될 수 있음을 이해할 수 있을 것이다. 그러므로, 본 개시된 실시예들은 한정적인 관점이 아니라 설명적인 관점에서 고려되어야 한다. 상기의 설명에 포함된 예들은 본 발명에 대한 이해를 위해 도입된 것이며, 이 예들은 본 발명의 사상과 범위를 한정하지 아니한다. 상기의 예들 외에도 본 발명에 따른 다양한 실시 태양이 가능하다는 것은, 본 발명이 속한 기술 분야에서 통상의 지식을 가진 자에게는 자명할 것이다. 본 발명의 범위는 전술한 설명이 아니라 특허청구범위에 나타나 있으며, 그와 동등한 범위 내에 있는 모든

차이점은 본 발명에 포함된 것으로 해석되어야 할 것이다.

<106> 또한 본 발명에 따른 상기의 각 단계는 일반적인 프로그래밍 방법을 이용하여 소프트웨어적으로 또는 하드웨어적으로 다양하게 구현할 수 있다는 것은 이 기술분야에 통상의 기술을 가진 자라면 용이하게 할 수 있는 것이다.

<107> 그리고, 본 발명의 일부 단계들은, 컴퓨터로 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 코드로서 구현하는 것이 가능하다. 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록장치를 모두 포함한다. 컴퓨터가 읽을 수 있는 기록매체의 예로는 ROM, RAM, CD-ROM, CD-RW, 자기 테이프, 플로피디스크, HDD, 광 디스크, 광자기 저장장치 등이 있으며, 또한 캐리어 웨이브(예를 들어, 인터넷을 통한 전송)의 형태로 구현되는 것도 포함한다. 또한 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산방식으로 컴퓨터가 읽을 수 있는 코드로 저장되고 실행될 수 있다.

도면의 간단한 설명

<108> 도 1은 본 발명에 따른 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템의 구성을 블록으로 도시한 것이다.

<109> 도 2는 도 1의 위치 기반 서비스에서의 서술 기반의 세만틱 프리페칭을 이용한 동적인 질의 처리 시스템의 상세 블록도를 도시한 것이다.

<110> 도 3은 영역 질의의 일 예를 나타낸다.

<111> 도 4는 최 근접 질의 처리의 일 예를 보여준다.

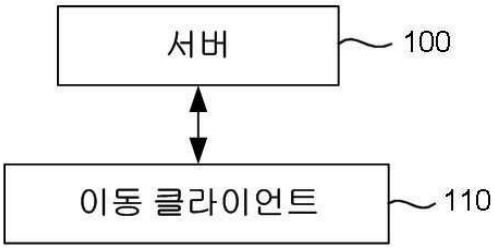
<112> 도 5는 이동 클라이언트에게 전송하는 프리페칭 서술을 결정할 수 있는 서버 알고리즘을 도시한 것이다.

<113> 도 6은 이동 클라이언트의 세만틱 영역 질의 처리 알고리즘을 도시한 것이다.

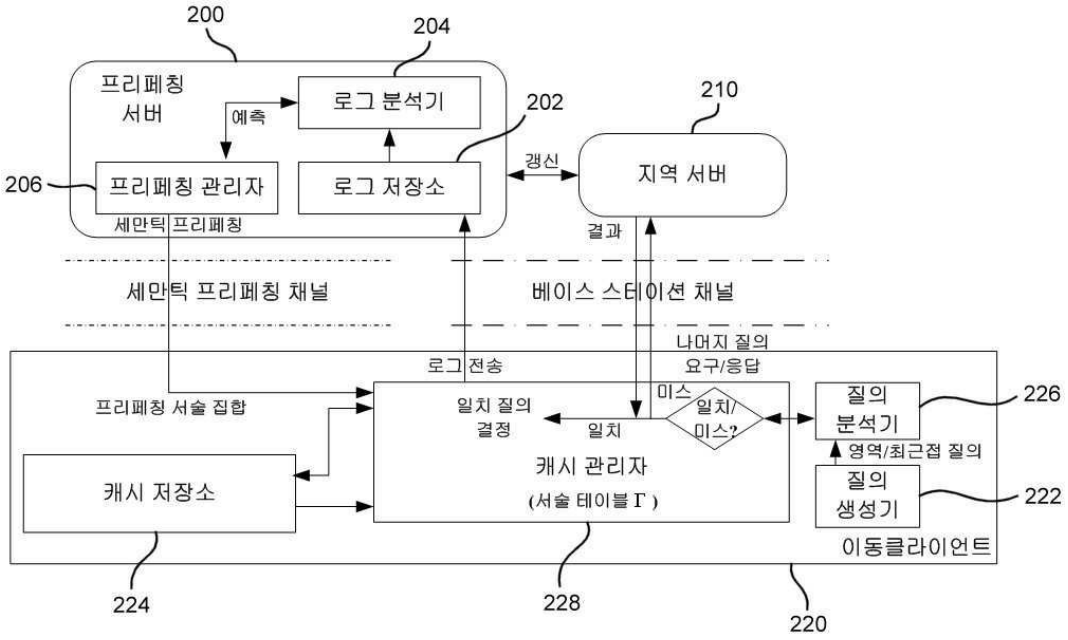
<114> 도 7은 세만틱 최 근접 질의를 처리하기 위한 이동 클라이언트에서 사용되는 알고리즘을 기술한 것이다.

도면

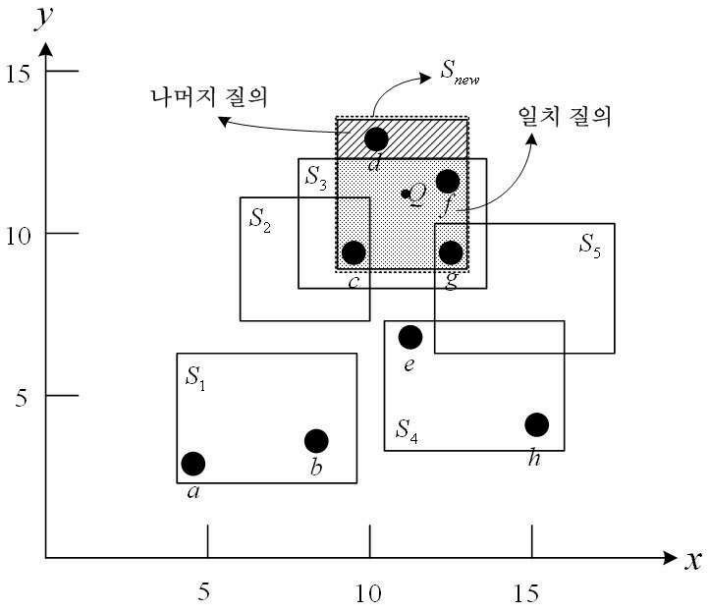
도면1



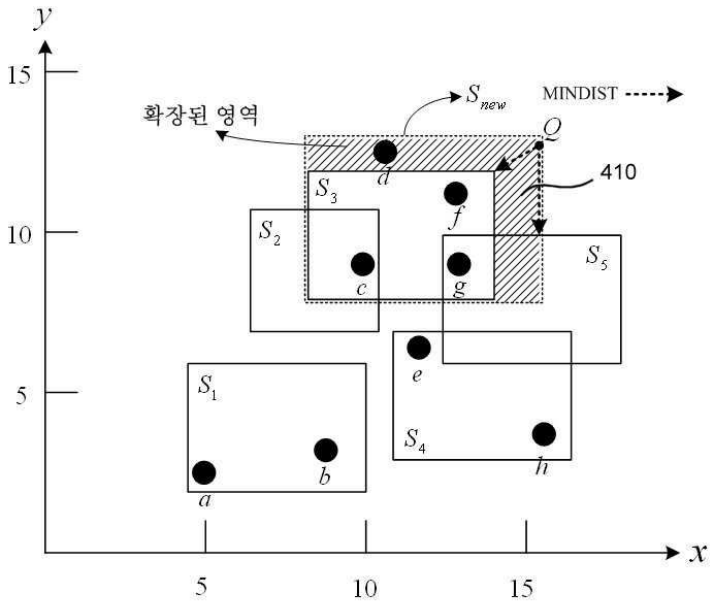
도면2


200
204
프리페칭
서버
로그 분석기
예측
202
210
갱신
지역 서버
206
프리페칭 관리자
로그 저장소
세만틱 프리페칭
결과
세만틱 프리페칭 채널
베이스 스테이션 채널
나머지 질의
요구/응답
로그 전송
프리페칭 서술 집합
일치 질의
결정
미스
일치/
미스?
일치
질의
분석기
226
영역/최근접 질의
캐시 관리자
(서술 테이블 Γ)
질의
생성기
222
캐시 저장소
이동클라이언트
224
228
220

도면3


y
15
10
5
x
5
10
15
나머지 질의
S_{new}
일치 질의
S_1
S_2
S_3
S_4
S_5
a
b
c
d
e
f
g
h
Q

도면4



서술 테이블 $\Gamma = \{S_1, S_2, S_3, S_4, S_5, S_{new}\}$

도면5

프리페칭 서술 결정 알고리즘

사용 기호
F_i : 프리페칭 세그먼트 F_d : 프리페칭 서술
L_f : 로그 파일 $\log_m$: 로그 메시지

입력: 이동 클라이언트의 위치와 식별자(identifier) 출력: F_d

Procedure:
if a mobile client is in a semantic prefetching area **then**
  Check user's identifier;
  if $L_f \leftarrow NULL$ **then**
    $F_i \leftarrow NULL$;
  else
    for each prefetching segment F_i in log file L_f **do**
      $F_d \leftarrow \bigcup_{i \in I} F_i$, where $F_i \in L_f$;
    end for
else
  if the server receives additional log message $\log_m$ **then**
    Update L_f ;
  end if
end if

도면6

세만틱 영역 질의 처리 알고리즘

사용 기호
Q_i : 영역 질의 R_i : Q_i 의 질의 결과
S_i : 세만틱 세그먼트
Q_{probe} : Γ 의 세만틱 정보와 일치하는 질의 결과 영역에 대한 세만틱 정보
$Q_{remainder}$: Γ 에서 얻을 수 없는 질의 결과 영역에 대한 세만틱 정보
R_{buffer} : 일시적인 결과 저장용 버퍼
S_{new} : 새롭게 확장되어 생성된 세만틱 세그먼트

입력: Q_i 출력: R_i

Procedure:
$Q_i \leftarrow EMPTY$;
if a mobile client issues Q_i **then**
  for each S_i overlapped with Q_i **do**
    if Q_i totally retrieves the result in S_i **then**
      $Q_{probe} \leftarrow Q_i$, $R_i \leftarrow Q_{probe}$;
    else if Q_i partially retrieves the result in S_i **then**
      Q_i is trimmed as Q_{probe} and $Q_{remainder}$;
      while $Q_{remainder} != NULL$ **do**
        $R_{buffer} \leftarrow Q_{probe}$;
        Send $Q_{remainder}$ to the server;
        Save the result of $Q_{remainder}$ to R_{buffer} ;
      end while
      $S_{new} \leftarrow R_{buffer}$, $R_i \leftarrow S_{new}$;
      add S_{new} to Γ ;
  end for
  if Q_i does not retrieves the result in any S_i **then**
    Send Q_i to the server;
    Save the result of Q_i to S_{new} ;
    $R_i \leftarrow S_{new}$;
    add S_{new} to Γ ;
end if

도면7

세만틱 최 근접 질의 처리 알고리즘

사용 기호
Q_i : 최 근접 질의 R_i : Q_i 의 질의 결과
Count(Y): Y 의 원소 개수 반환 함수
Loc(Q_i): 질의한 이동 클라이언트의 좌표를 반환하는 함수
GetArea(S_i): 세만틱 세그먼트 S_i 의 영역을 반환하는 함수
GetShortestElement (X,Y): X 와 가장 가까운 Y 의 원소이면서 Q_i 의 질의 조건을 만족하는 원소를 반환하는 함수
priority_queue[ ]: Γ 의 세만틱 세그먼트 원소 중 **Loc(Q_i)**와 가장 가까운 세만틱 세그먼트들을 거리 순서대로 저장하는 자료구조
CreateSemanticSeg(X,Y):X 와 Y 의 영역을 포함하는 새로운 세만틱 세그먼트를 생성한 후 생성된 세만틱 세그먼트를 반환 함수
입력: Q_i 출력: R_i

Procedure:
Criterion ← true;
if a mobile client issues Q_i **then**
  if $\exists S_i \in \Gamma$ such that (Loc(Q_i) ∈ GetArea(S_i)) **then**
    R_i ← GetShortestElement (Loc(Q_i), S_i);
  end if
  if (Not $\exists S_i \in \Gamma$ such that (Loc(Q_i) ∈ GetArea(S_i))) ∨ ($R_i = NULL$)
  then
    for j=1 to Count(Γ) **do**
      priority_queue[ ] ← GetShortestElement (Loc(Q_i), Γ);
    end for
    while Criterion ^ (1 ≤ j ≤ Count(priority_queue[ ])) **do**
      S_k ← CreateSemanticSeg(Loc(Q_i), priority_queue[j])
      R_i ← GetShortestElement (Loc(Q_i), S_k);
      if $R_i \neq NULL$ **then**
        Criterion ← false;
        add S_k to Γ ;
      end if
    end while
  end if
end if