질의분리를 이용한 단계적 웹통합 방법

백주흠*, 조성배**
*연세대학교 인지과학통합과정
**연세대학교 컴퓨터과학과
e-mail:bjh@yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Procedural Web Integration Using Query Separation

Joo-Huem Baek*, Sung-Bae Cho**
*Graduate Program in Cognitive Science, Yonsei University
**Dept of Computer Science, Yonsei University

요 약

인터넷상의 정보가 빠르게 변화함에 따라 정보의 최신성을 유지하며 웹 상의 정보를 통합하기 위한 실시간 웹 통합 방법들이 제안되고 있다. 하지만 대부분이 실시간 추출에 소요되는 시간이 너무 긴 단점을 가진다. 본 논문에서는 추출시간을 줄이기 위해 입력 질의를 분리하여 사용자와 상호작용하며 단계적으로 웹 통합을 수행하는 방법을 제안한다. 실험을 통해 기존 방식과 응답시간을 비교하여 유용성을 확인하였다.

1. 서론

일반적으로 웹 통합 시스템들은 서비스에 이용될 정보들을 사전에 수집해두고 사용자 질의에 응답한다. 이 방법은 응답속도는 빠르지만 정보의 최신성, 정확성을 보장해주지 못하기 때문에 빠르게 업데이트되거나 사전에 모두 수집하기가 어려울 만큼 방대한 양의 정보를 대상하는 서비스시스템에서는 실시간으로 웹정보를 통합할 수 있는 방식이 필요하다. 하지만 이전의 웹 통합시스템 분야의 연구에서 드러났듯 실시간 웹 통합 방식은 응답시간이 매우 길다[1]. 이런 이유로 현재 인터넷에서 사용되고 있는 대부분의 웹 통합서비스들은 사전 추출후 응답 방식을 채택하고 있다. 본 논문은 실시간 방식의 통합에서 응답시간을 줄이기 위해 단계적으로 웹 통합을 수행하는 방법을 제안한다. 이 방법은 사용자 선택에 따라 단계적으로 정보에 접근할 수 있도록 하여 추출에 소요되는 웹 이동 횟수를 최소화 시키고자 한다. 2장에서 웹 통합 시스템에 대해 살펴보고 3장에서 본 논문이 제안하는 방식에 대해 구체적으로 기술한다. 4장에서 제안하는 방식과 기존 방식의 응답시간을 비교한 뒤 5장에서 결론을 서술한다.

2. 웹 통합 시스템

실시간 웹 통합 시스템은 일반적으로 그림1과 같이 질의처리, 정보추출, 내부질의의 3가지 모듈로 구성되는데 이 중 질의처리 모듈은 시스템에 입력된 이용자 질의로부터 추출계획을 수립하는 역할을 담당한다. 정보 추출 모듈은 사전에 기술된 웹 추출 규칙(래퍼)을 이용해 특정 웹 문서상에서 필요한 정보를 선택적으로 추출해내는 역할을 담당하고 내부 질의 모듈은 추출된 정보를 시스템 내부에서 추가 질의를 통해 필터링, 조인할 수 있도록 한다.

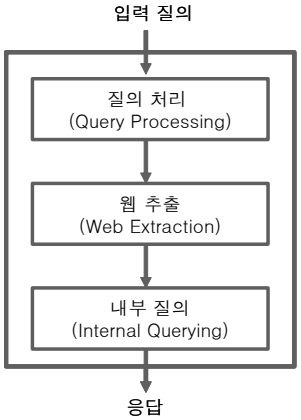

그림 1. 실시간 웹 통합 시스템

이 3가지 모듈은 순차적으로 진행되는데 표2에서 보는 바와 같이 웹 추출과정에서 일어나는 웹 이동에서 소요되는 시간이 전체 응답시간에 90%이상을 차지한다. 몇몇 연구들이 웹 이동을 줄이기 위해 제안되어 왔는데 Ariadne 시스템[1]의 경우 사전에 적절한 정보가 포함된 웹 싸이트만을 선택하는 과정을 질의 처리모듈에 추가하였고, Emerac시스템[2]에서는 Greedy 최소화 알고리즘을 이용해 추출대상에서 중복되는 정보출처를 제거하는 방법을 제안하였다. 이와 유사하게 Information Manifold 시스템[3]에서는 데이타베이스 분야에서 이용되는 사용자 질의 최적화 알고리즘인 Bucker알고리즘을 확장하여 최적의 웹 싸이트만을 선택할 수 있도록 질의 변환과정을 제안하고

있다.

3. 질의분리를 이용한 단계적 웹통합 방법

입력질의에 대해 일괄 추출 후 정보에 접근하는 기존의 방식은 그림 2-A와 같이 "웹 이동 및 추출"과정에서 상당한 응답지연(984s)을 발생시킨다. 이와 달리 제안하는 "단계적 웹 통합 방법"은 사용자의 선택에 따라 선택적, 단계적으로 정보접근이 일어나도록 한다. 따라서 그림 2-B와 같이 초기 응답지연이 짧고(16s) 총 응답소요시간(47s)도 짧다.

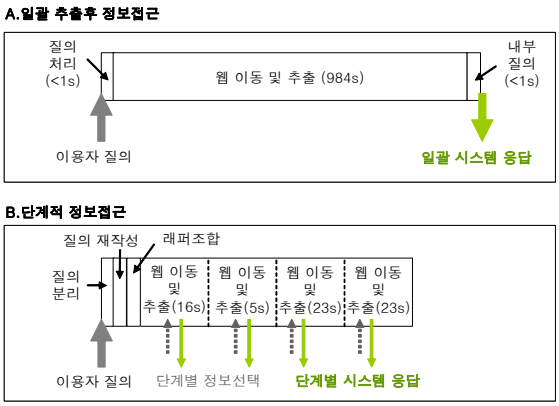

그림 2. 단계적 정보접근과 일괄추출후 정보접근 비교

제안하는 방법은 이용자 질의가 입력되면 먼저 단계별로 정보 추출이 일어날 수 있도록 최초의 입력질의를 분리시키는 것이 가장 큰 특징이다.

질의 분리과정은 단일질의에 두 가지 이상의 개념이 포함될 경우 개념별로 질의를 분리하는 "상-하위 질의분리" 단계와 웹 싸이트의 정보 구조에 따라 질의를 분리하는 "목록-상세 질의분리"단계가 순차적으로 수행된다.

만약 런던에 있는 숙소들에 대한 기본 정보와 각각의 숙소에 대한 리뷰들을 함께 제공받기 위한 질의가 가상의 웹 통합 시스템에 입력되었다면 입력 질의는 그림 3과 같이 4개의 질의로 분리되어 단계별 웹 추출에 이용된다.

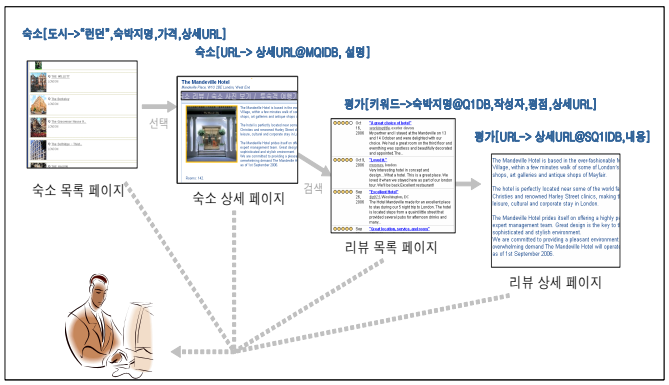

그림 3. 질의[1] 분리의 예

● 상-하위 질의 분리

하나의 숙소에 대한 기본 정보들은 일반적인 숙소예약 웹 싸이트에서 얻을 수 있지만 각 숙소에 대한 리뷰정보들은 TripAdvisor와 같은 독립된 싸이트에서 제공되고 있다. 따라서 이용자는 기본적인 숙소정보를 획득한 뒤 선택적으로 리뷰 싸이트를 검색하며 정보를 통합한다. 이런 이용자의 정보검색패턴은 다른 도메인에서도 유사하게 발견되는데 상-하위 질의분리는 바로 이러한 정보 이용자의 정보검색패턴을 모방한 것이다. 그 과정은 사전에 정의된 도메인 모델을 기반으로 입력 질의에서 포함(HAS-A)관계를 가지는 하나이상의 개념쌍이 존재할 경우, 표1과 같이 각 쌍에 대해 상하위 질의로 분리하는 방법으로 처리된다.

표1. 상-하위 질의 분리의 예

입력질의: 숙소[지역->"런던", 숙소명, 설명, 가격, 리뷰[작성자, 내용, 평점]] → 상위질의: 숙소[지역->"런던", 숙소명, 설명, 가격] → 하위질의: 리뷰[작성자, 내용, 평점]

● 목록-상세질의 분리

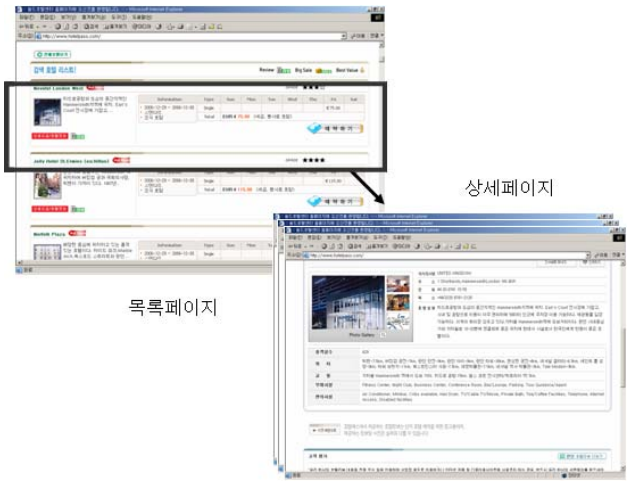

그림 4. 웹 싸이트 구조

정보제공을 목적으로 하는 대부분의 웹 싸이트는 그림 4와 같이 목록페이지와 상세페이지로 구분되어 있기 때문에 목록페이지에 있는 속성을 추출하는 것보다 상세페이지에 있는 속성을 추출하는데 훨씬 많은 시간이 소요된다(표2 참조).

표2. 웹 추출 시간

목록 웹 추출 시간 = 목록페이지 이동시간 + 추출시간 상세 웹 추출 시간 = 목록페이지 이동시간 + (추출항목의 수 * 상세페이지 이동시간) + 추출시간

따라서 이용자가 관심없는 상세정보페이지의 속성들까지 모두 추출하는 것은 비효율적이다. 하지만 꼭 통합된 화면

1) 질의 : 숙소[지역->"런던", 숙소명, 설명, 가격, 리뷰[작성자, 내용, 평점]]

상에서 목록으로 포함되어야하는 속성이 있을 경우, 이를 모두 추출하는데 걸리는 시간(총 추출시간)과 해당속성의 유용성(목록적합도)을 비교해 추출여부를 판단할 필요가 있다.

속성 M에 대한 목록 적합도는 표3과 같이 전체 추출대상 웹 싸이트 중에서 해당 속성을 목록에 가지고 있는 비율로 나타내며 총 추출 시간은 각각의 웹 싸이트에서 계산된 속성 M의 추출시간들 중 가장 큰 값이다.

표3. 속성에 대한 목록적합도와 추출시간

속성 M에 대한 목록적합도 = 속성 M을 출력속성으로 가지는 목록래퍼의 수 / 전체 목록래퍼의 수
속성 M에 대한 총 추출시간 = 각각의 웹 싸이트의 "단일 웹 싸이트에서의 속성 M 추출시간"값의 최대값
단일 웹 싸이트에서의 속성 M 추출시간 = (해당 속성이 목록페이지에 위치할 경우) 해당 웹싸이트의 목록 웹 추출 시간 (상세페이지에 위치할 경우) 해당 웹싸이트의 상세 웹 추출 시간

표4의 예와 같이 각 속성의 목록적합도와 총 추출시간이 주어질 경우 "목록적합도 > K * 총 추출시간 (K : 도메인 상수)" 인 속성 M을 찾아 표5와 같이 해당 속성보다 총 추출시간이 짧은 모든 속성을 목록질의로, 나머지를 상세질의로 분리해낸다.

표4. 목록적합도와 총 추출시간 테이블

속성(M)	목록적합도	총 추출시간
설명	0%	148
가격	50%	141
숙소명	100%	24

표5. 목록-상세 질의 분리의 예

입력질의: 숙소[지역->"런던", 숙소명, 설명, 가격] → 목록질의: 숙소[도시->"런던",숙박지명,가격] → 상세질의: 숙소[URL-> 설명]

두 단계의 질의 분리가 끝나면 분리된 질의는 사용자 인터페이스와 상호작용하며 단계적으로 웹 추출하는데 이용된다.

4. 실험

본 실험에서는 정보의 변경이 빈번하게 발생하는 숙박정보분야에서 통합 시스템을 구축하여 "단계적 웹 통합방법"과 기존 "일괄추출후 응답방법"에 대해 응답시간을 비교하였다.

그림 6은 수평 통합 싸이트와 수직 통합 싸이트의 수를 증가시켜 나갈 때 응답시간의 변화를 나타낸 그래프이다.

수평통합은 그림 5와 같이 도메인 모델상에서 동일한 개념 혹은 IS-A하위관계의 개념에 속한 웹 싸이트들의 정보를 통합하는 것이다. 수직통합은 HAS-A관계를 가지는 개념들에 속한 웹 싸이트들의 정보를 통합하는 것이다.

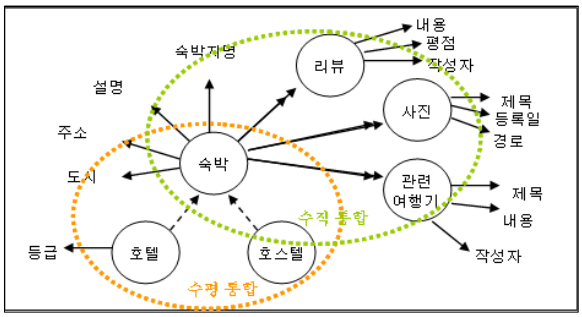

그림 5. 수직/수평 통합

그림 6-A 그래프는 수평통합의 대상을 증가시켜나감에 따라 응답시간이 어떻게 증가하는지 알기 위해 동일한 질의에 대해 숙소정보제공 싸이트들(호텔패스, HostelTimes, booking. com, HostelWorld)을 통합대상으로 차례로 추가시키며 응답시간을 측정하여 비교한 것이다. 그림6-B 그래프는 수직통합의 대상을 증가시켜나감에 따라 응답시간이 어떻게 증가하는지 살펴보기 위해 사용자 질의의 대상이 되는 개념을 숙소부터 리뷰, 사진, 관련여행기로 순으로 증가시켜가며 비교한 것이다. 각각의 개념에는 1개의 웹 싸이트만을 통합대상으로 사용하였다.[2)]

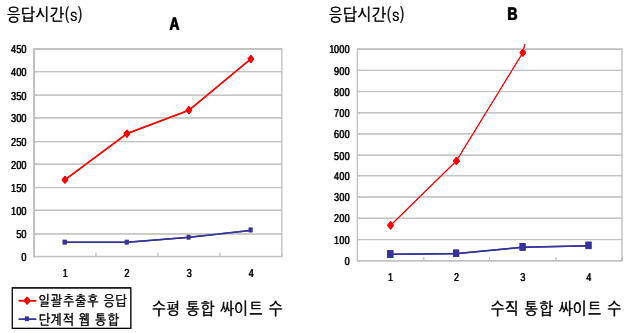

그림 6. 응답시간 비교

그 결과 수평과 수직 통합 싸이트 수의 증가에 따라 "단계적 웹 통합방법"이 상당한 응답시간 개선을 보였으며 특히 수직 통합 싸이트 증가에서 더욱 큰 차이를 보였다[3)]. 이는 "일괄추출후 응답방법"이 984초동안 호텔패스

2) 숙소: 호텔패스, 리뷰: TripAdvisor, 사진: Flickr, 관련 여행기: 네이버블로그

3) 통합싸이트 수가 3개일때 "일괄추출후 응답방법"은 984

웹 싸이트에서 조건이 맞는 숙소(ex. 런던내의 숙소)에 대한 모든 리뷰, 사진, 관련 여행기를 시스템에 가져온 뒤에 응답한데 반해 “단계적 웹 통합방법”은 이용자가 선택한 2개의 숙소에 대해서만 이런 추가 추출을 수행해 불필요한 웹 추출시간을 줄일 수 있었기 때문이다.

5. 결론

본 논문은 기존 실시간 방식 시스템의 응답시간 문제를 개선하기 위해 질의를 분리하여 단계적으로 웹 통합하는 방법을 제안하였다. 실험을 통해 “단계적 웹 통합방법”이 기존의 “일괄추출후 응답방법”에 비해 응답시간을 상당히 줄일 수 있음을 확인하였다.

참고문헌

[1] C. Knoblock, et al.,, "The ARIADNE approach to Web-based information integration," Int. Journal of Cooperative Information Systems, 10(1-2):145-169, 2000.

[2] S. Kambhampati, et al., "Optimizing Recursive Information Gathering Plans in EMERAC," Journal of Intelligent Information Systems , 22(2), 119 - 153, 2004.

[3] Q. Bai, et al.,, "Bucket-Based Query Rewriting with Disjunctive Data Source," In Proc. of the 2004 IEEE / WIC / ACM Int. Conf. on Web Intelligence, 566-569, 2004

초, “단계적 웹 통합방법”은 63초