

메타 검색 기법을 이용한 이동 에이전트 기반의 상품 가격 비교 검색 모델

문 황, 왕경아, 고 현, 이연식
군산대학교 컴퓨터정보과학과

Product Price Comparison and Search Model based on Mobile Agent Using the Meta-Search Method

Hwang Moon, Kyoung-A Wang, Hyun Ko, Yon-Sik Lee
Dept. of Computer Information Science, Kunsan National University

요 약

최근 인터넷이 보편화됨에 따라 상품구입을 위한 검색서비스는 필수가 되었다. 특히, 동일한 품목에 대해 소비자가 원하는 상품을 검색하고, 판매자에 따른 상품 가격의 차이를 비교하는 서비스가 지원되고 있다. 하지만 가격 비교를 제공하는 사이트들은 각 가격 비교 사이트별로 확보하고 있는 쇼핑몰들만을 대상으로 검색을 하기 때문에, 동일 상품에 대하여 제시하는 가격이 다를 수 있다. 따라서 소비자는 정확한 최소의 가격을 알기위해 많은 시간을 소요하며, 수시로 변동되는 상품의 최소 가격에 대한 정확한 정보를 얻기 어렵다. 따라서 본 논문에서는 가격 비교를 제공하는 사이트들에 상품 가격 비교 정보를 요청하기 위하여 이들에게 각기 질의를 보냄으로써 소비자가 찾고자하는 상품과 최소 가격을 빠르고 정확하게 서비스해줄 수 있는 메타 검색기법을 이용한 이동 에이전트 기반의 상품 가격 비교 검색 모델을 설계 및 구현한다.

1. 서론

최근 인터넷이 보편화함에 따라 현대인들의 빠른 생활 패턴에 맞춰 상품구입을 위한 검색서비스 제공은 필수가 되었다. 특히, 동일한 품목에 대해 소비자가 원하는 상품을 검색하고, 판매자에 따른 상품 가격의 차이를 비교하는 서비스가 지원되고 있다[3,4,5,6]. 하지만 가격 비교를 제공하는 사이트들은 각 가격 비교 사이트별로 확보하고 있는 쇼핑몰들만을 대상으로 검색을 하기 때문에, 동일 상품에 대하여 제시하는 가격이 다를 수 있다. 따라서 소비자는 정확한 최소의 가격을 알기위해 많은 시간을 소요하며, 수시로 변동되는 상품의 최소 가격에 대한 정확한 정보를 얻기 어렵다. 따라서 본 논문에서는 가격 비교를 제공하는 사이트들에 동시에 상품 가격 비교 정보를 요청하기 위하여 이들에게 각기 질의를 보냄으로써 소비자가 찾고자하는 상품과 최소 가격을 빠르고 정확하게 서비스해줄 수 있는 메타 검색기법을 이용한 이동 에이전트 기반의 상품 가격 비교 검색 모델을 설계 및 구현한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구로 메타 검색기법과 이동에이전트에 대해 간단히 알아보고, 3장에서 메타 검색기법을 이용한 상품 가격 비교 검색 모델의 구조를 설계하고, 4장에서는 상품 가격 비교 검색 모델을 구현한다. 마지막으로 5장에서는 결론 및 향후 연구 과제를 제시한다.

2. 관련연구

2.1 메타 검색기법

메타 검색기법은 기존 개별 검색엔진의 특성을 파악하여 검색엔진을 별도로 선택할 필요 없이 한번의 질의어 입력으로 여러 가격 비교를 제공하는 사이트의 결과 값을 한눈에 볼 수 있게 만든 검색엔진으로, 기존의 많은 검색엔진들에 대하여 일관된 인터페이스 및 질의를 제공하는 기능이 있으며, 검색에 대한 효율을 높이기 위해 멀티-쓰레드를 이용하여 대상 가격 비교 사이트들에 접속하고 질의한 후, 전송된 결과를 효율적으로 파싱하여 URL 및 요약된 내용 등을 소비자 옵션에 따라 보여줄 수 있는 기능을 고려하여 개

발되고 있다[7,9,11].

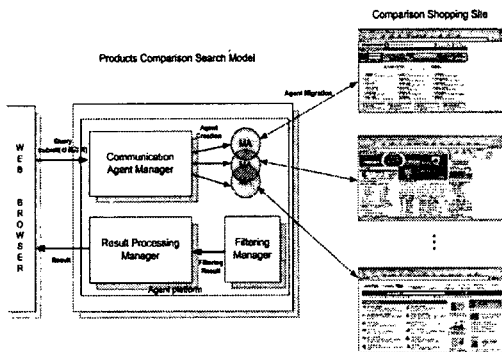
2.2 이동 에이전트

이동에이전트는 에이전트의 특성 중 하나인 이동성(mobility)이 강조된 에이전트로서 클라이언트/서버 형태로 네트워크를 이용해 임무를 수행하는 일반 에이전트와는 달리 시스템 간에 코드와 현재의 에이전트 내부 상태 자체가 이동하여 임무를 수행하는 에이전트로 네트워크 에이전트(network agent) 혹은 순회 에이전트(itinerant agent)라고 불린다. 이동에이전트 환경에서는 클라이언트가 만든 에이전트를 원격 서버 호스트로 보내어 필요로 하는 기능을 서버의 도움을 받아 업무를 수행하거나 단독으로 기능을 수행하게 된다. 서버로 이주해간 에이전트는 적절한 시기에 이동하여 필요한 기능을 수행할 수 있고 필요에 따라서는 또 다른 서버 플랫폼으로 이동하여 필요한 기능을 수행할 수 있다[8,12].

3. 상품 가격 비교 검색 모델 설계

3.1 시스템 구조

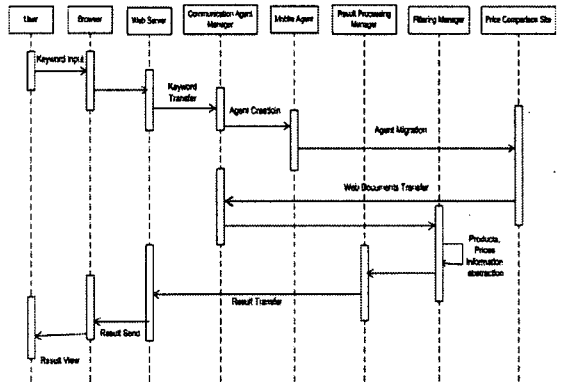
본 논문에서 제안하는 시스템은 통신 에이전트 관리자, 필터링 관리자, 결과처리 관리자로 구성되어 있다. 시스템의 구조는 [그림 1]과 같다.



[그림 1] 상품 비교 검색 시스템의 구조

시스템의 수행과정은 먼저, 소비자가 웹 브라우저를 통하여 상품 검색 및 최소 가격 정보를 요청을 하면 통신 에이전트 관리자(Communication Agent Manager)는 입력받은 검색어를 각각의 가격 비교 사이트들과 통신을 하는 이동 에이전트(Mobile Agent)에게 그 검색어를 전달하고 각각의 에이전트에 대하여 쓰레드를 생성하여 수행시킨다. 이들 이동 에이전트들은 처리한 결과를 필터링 관리자(Filtering

Manager)에 전송하여 하이퍼링크, 상품명, 가격 순서대로 파싱을 수행하게 된다. 파싱된 결과는 소비자에게 더욱 정확한 결과를 서비스하기 위해 결과 처리 관리자(Result Processing Manager)에 의해 재처리되고, 재처리된 정보는 웹 브라우저를 통해 소비자에게 전달한다. [그림 2]는 상품 가격 비교 검색 모델의 수행과정이다.



[그림 2] 상품 가격 비교 검색 모델의 수행과정

3.2 통신 에이전트 관리자

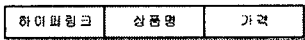
소비자로부터 웹 브라우저를 통해 입력받은 질의어를 전송하는 기능을 담당하고 있다. 통신 에이전트 관리자(Communication Agent Manager)는 가격 비교 사이트들과 통신하는 이동 에이전트들을 쓰레드로 생성하여 동기화하는 기능을 수행하는 부분이다. start() 메소드의 호출과 함께 각각의 이동 에이전트에 대하여 New Thread() 클래스 상속을 통해 생성되어 수행된다. 수행되는 과정에서 각 가격 비교 사이트들과 통신을 통해 원하는 정보(하이퍼링크, 상품명, 가격정보)를 결과값으로 반환하게 된다. 각각의 결과값들은 모든 쓰레드가 종료할 때까지 기다리는 동안 먼저 종료된 쓰레드에 대해서는 join을 통해 나중 종료된 쓰레드와 결과가 합쳐지면서 임시 기억공간에 저장되며 저장된 정보는 필터링 관리자로 전송된다.

3.3 필터링 관리자

가격 비교 사이트로부터 받은 결과는 html문서이다. 필터링 관리자(Filtering Manager)에서는 이 결과에 대하여 파싱을 수행하는 기능을 담당하고 있다. [그림 3]과 같이 파싱 과정을 거치면 하이퍼링크, 상품명, 가격정보가 추출되게 된다. 보통의 결과로 출력되는 html문서들은 table태그 안에 원하는 정보들이

포함되어 있다. 가격정보는 [그림 4]의 첫 if문에서 추출할 수 있다. 상품명은 바로 앞에 추출해낸 <a>태그의 텍스트 정보이다. 마지막으로 link 정보는 <a>태그의 속성 href의 value를 추출하여 알 수 있다. 각각의 조건식을 통해 하이퍼링크, 상품명, 가격정보를 알아낼 수 있다.

```
while(Tag != end of html) {
  if (Tag == '<td>') {
    if (td_Tag의 콘텐츠가 가격정보를 포함) {
      price_info = 가격정보; //가격
    }
    else if (td_Tag_next == '<a>') {
      product_name = a_Tag의 텍스트정보; //상품명
      hyper_link = link 정보; //하이퍼링크
    }
  }
}
```



[그림 3] Html 문서를 파싱하여 정보를 추출하는 과정

```
string[] price_Name = ('가격', '최저가', '제품가격', ...);
for (int i=0; i<price_Name.length; i++)
{
  int price_Position = table_Start_Tag부터 시작하여
  price_Name[i]가 가지고 있는 문자열이 발견되는 위치
  int count = table_Start_Tag부터 시작하여 price_Name[i] 사이에
  포함되어 있는 '<td>'의 개수
}
```

[그림 4] 가격정보를 알아내기 위한 패턴

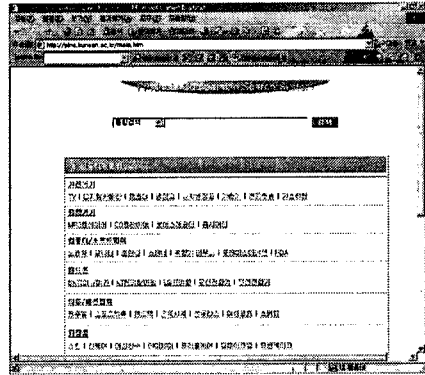
3.4 결과처리 관리자

필터링 관리자로부터 처리된 결과를 받아 작업결과들을 출력하는 모듈이다. 결과처리 관리자(Result Processing Manager)는 별도로 재처리(re-processing) 기능을 포함한다. 이는 가격 비교 사이트들로부터 받아온 결과에 대하여 파싱을 거친 정보(하이퍼링크, 상품명, 가격정보)들 중 상품명에 대하여 필요 없는 정보(숫자 혹은 특수문자)들을 제거하고 순수한 상품명만 남도록 한다. 그리고 같은 상품명에 대하여 집합을 이룰 수 있도록 순서를 재배치한다.

4. 상품 가격 비교 검색 모델 구현

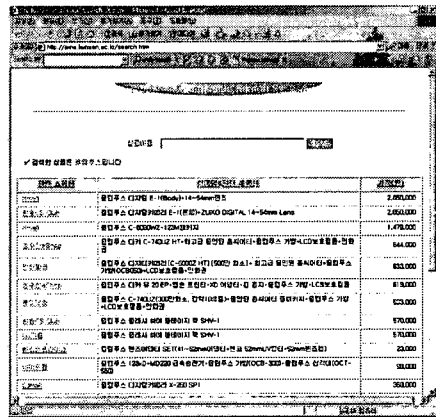
[그림 5]는 본 모델의 소비자 인터페이스를 설계한 웹 브라우저 화면이다. 이 화면은 상품을 12가지 품목

별로 구분하였다. TV 등 상품을 대표하는 품목을 클릭해서 상품을 검색해도 되지만 직접 검색하는 방법도 있다.



[그림 5] 상품 가격 검색 비교 모델 초기화면

특정 상품명에 입력하여 검색하면 [그림 6]과 같이 쓰레드가 수행된 만큼만 화면에 출력시키고 나머지는 background로 수행되고 있음을 보여주고 있다. 모든 작업이 종료되면 검색이 끝났다는 메시지를 화면에 출력함으로써 모든 쓰레드들이 작업을 종료하였음을 알 수 있다.



[그림 6] 특정 상품 검색 결과화면

모든 쓰레드가 작업이 끝나게 되면 검색된 결과물에 대하여 재처리 및 가격별 정렬을 할 수 있도록 하였다. 즉, 관련 쇼핑몰, 상품명 및 관련 제품들, 가격을 클릭하면 각각에 대하여 정렬하게 된다. 가격을 클릭하면 [그림 7]과 같이 해당 상품의 최저가 순으로 관련 쇼핑몰이 정렬되어 나타난다.

