

동적 로봇에이전트를 이용한 주문형 검색엔진의 설계 및 구현

김 성[†]·박 철 우^{††}·이 충 석[†]·박 규 석^{†††}

요 약

전자상거래 관련기술은 짧은 기간에 급속한 발전을 이루었으며, 최근에는 그 영역이 B2B로까지 확장되고 있다. 이와 같은 전자상거래의 발전에 발맞추어 수많은 사이트들의 상품들에 대한 비교·분석 정보도 요구되고 있다. 현재 국내에도 쇼핑몰간의 가격비교 정보가 제공되고 있으나 그 갱신주기가 길어 효율적이지 못하며 보다 빠른 갱신을 위한 무절제한 정보 수집으로 인해 대상 쇼핑몰에 많은 부하를 발생시키고 있다.

본 논문에서는 대상 쇼핑몰의 상태에 따라 로봇의 동작이 동적으로 변경되는 동적 로봇에이전트를 이용하여 대상 쇼핑몰의 상품 정보에 대해 대상 서버의 부하를 최소로 줄이면서 최단 시간 내에 수집·분석하여 고객에게 동일 상품에 대한 최저가의 쇼핑몰을 제시하며 맞춤서비스를 제공할 수 있는 주문형 검색엔진을 설계 및 구현하였다.

Design and Realization of Retrieval Engine On Demand Using a Dynamic Robot Agent

Sung Kim[†] · Chol Woo Park^{††} · Chung Seok Lee[†] · Kyoo Seok Park^{†††}

ABSTRACT

The technologies relevant to e-business have rapidly developed during very short period of time and recently it is expanding to the area of B2B. Keeping pace with this development in e-business, the information of comparison or analysis on commodities of a lot of sites is also required. Though the information on price comparison among internal shopping malls are now being offered, it's not efficient for its renewing intervals are long and, due to some indiscreet collection of information for the purpose of fast renewal, much loads are being generated on the pertinent shopping malls. In this article, the retrieval engine on demand is designed and realized using a dynamic robot agent changing kinetically on the status of the pertinent shopping malls that can offer the customized service and presents the shopping malls with the lowest price for each commodity under e-business after the shortest time of collection and analysis while not giving loads to the pertinent shopping malls.

키워드 : 전자상거래(EC), 검색 에이전트(Search Agent)

1. 서 론

인터넷의 폭발적인 증가세에 따른 전자상거래 영역의 많은 발전이 기대된다.

이와 같이 계속적으로 발전하는 전자상거래 사업에 발맞추어 수많은 사이트들의 상품에 대한 비교·분석 정보가 요구되고 있다.

본 논문에서는 전자상거래상의 상품의 정보를 최단시간 내에 수집, 분석하기 위한 주문형 검색엔진을 설계 및 구현하였다. 제안 시스템은 분산되어 있는 많은 정보를 쉽고 빠

르게 수집하여 갱신해 줄 수 있는 지능적인 로봇에이전트와 주기적으로 네트워크의 부하를 모니터링하고 그 결과를 기반으로 전체 시스템의 성능을 향상시킬 수 있는 실시간 로봇 제어 알고리즘으로 구성되어 있다.

2. 관련 연구

2.1 전자상거래

최근 정보통신 기술의 발달로 컴퓨터 네트워크를 이용하여 경제주체간 상품 및 서비스의 상업적인 이용이 증대되고 인터넷 기술이 발전하면서 가스를 교환하는 전자상거래가 확산되고 있다. 더욱이 인터넷의 상업적인 이용이 증대되고 인터넷 기술이 발전하면서 가상 상점(Cyber Shopping Mall), 전자 지불 체계, 전자양식, 정보보안, 정보인증 등 인터넷을 기반으로 한 전자상거래가 다양하게 이용되고 있다.[1, 2]

* 본 논문은 2000년도 경남대학교 학술연구조성비 지원에 의하여 쓰여진 것임.

† 준 회원 : 경남대학교 대학원 컴퓨터공학과

†† 준 회원 : 경남대학교 대학원 컴퓨터공학과

††† 정 회원 : 경남대학교 정보통신공학부 교수

논문접수 : 2001년 6월 28일, 심사완료 : 2001년 8월 14일

2.2 로봇 에이전트

로봇 에이전트란 원하는 정보를 얻기 위하여 웹 상의 문서들을 검색하고, 참조되는 문서들을 재귀적으로 검색하면서 웹의 하이퍼텍스트 구조를 추적하여 정보를 저장해 주는 프로그램으로 정의되며, 로봇(Robot)은 일반적으로 Wanderer, Spider, Web Crawler 등으로 불리기도 한다[3].

로봇들은 통계분석, 유지보수, 미러링(mirroring), 리소스 발견, 복합적인 사용 등에 이용되며, 상대 시스템에 과다한 부하를 부여, 로봇 프로세스의 동작 오류, URL 검사시의 로봇 성능 저하, 인덱싱 결과의 관리등과 같은 문제들을 발생시킬 수도 있다.

상대 시스템의 과부하를 막기 위한 방법으로 페이지 요구문 사이에 지연문을 넣거나 robots.txt 파일을 참고하여 로봇 배제 규약을 준수하도록 하고 있다[4, 5].

2.3 로봇 혼합 정렬 스케줄러

기존의 검색 시스템의 경우 문서를 수집하기 위해 특정한 로봇 스케줄링기법을 사용하지 않고 등록된 순서대로 로봇을 보내어 문서를 수집한다. 하지만 이 방법의 문제점은 해당 서버의 네트워크 부하를 고려하지 않기 때문에 부하가 많이 발생하고 있는 시점에도 로봇이 문서를 수집하는 경우가 발생하며 이로 인하여 로봇의 성능이 저하된다는 점이다.

이러한 문제를 보완한 혼합정렬 스케줄러는 정렬스케줄러에 의해 서버관리가 생성한 서버 리스트와 최적 부하 시간 탐색기가 생성한 리스트를 결합하여 우선순위 테이블을 생성한다[3].

2.4 언어 분석 기술

인터넷 공간의 정보는 멀티미디어 데이터로 되어 있지만 정보검색의 초점은 주로 문자로 표현된 언어 정보에 맞춰져 있다. 일반적으로 정보검색은 사용자의 질의를 분석하여 사용자가 원하는 정보를 찾아주는 함수라고 할 수 있다.

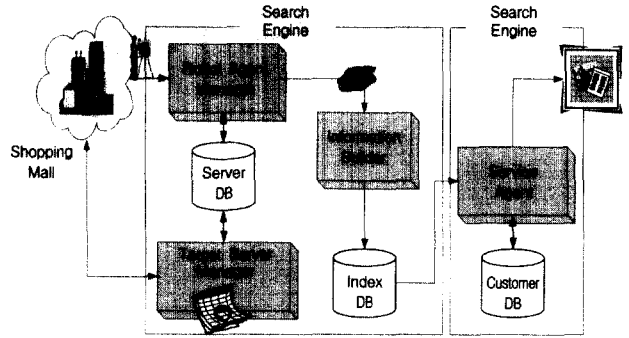
사용자 질의 속에 담긴 의도를 알아내기 위하여 정보검색 시스템은 질의의 구성과 핵심어의 의미 등을 분석할 수 있어야 하고, 검색 대상이 되는 정보가 무엇에 관한 것이며 사용자의 의도와 얼마나 일치하는지를 판단할 수 있어야 한다.

언어처리는 문장이나 구절 형태로 주어진 텍스트로부터 가능한 많은 정보를 추출하는 것을 목표로 하는데, 그 형태는 단어나 구절의 처리에서부터 대용어, 불용어 처리에 이르기까지 그 수준이 다양하다. 그러나 최근의 웹 정보검색 엔진에서는 언어처리를 위한 알고리즘 개발보다는 사전 및 지식정보에 의존하는 경향이다[6].

3. 시스템 설계

3.1 시스템 구성

제안하는 주문형 검색엔진 시스템의 구성은(그림 1)과 같다.



(그림 1) 전체 시스템 구성도

제안시스템은 Search Engine과 Service Engine으로 구성된다. Search Engine은 인터넷상에 산재되어 있는 쇼핑몰들에서 상품에 관한 정보를 수집하여 이를 유용한 정보로 가공하며, Service Engine은 가공된 정보들을 소비자의 요구에 맞게 제공해주는 역할을 담당한다.

Search Engine은 Target Server Manager와 Robot Agent Manager, Information Builder 모듈로 구성된다.

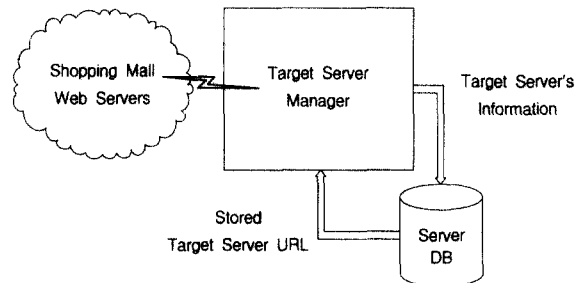
Target Server Manager에서 미리 구성된 대상 쇼핑몰 서버에 대한 부하량 정보와 Robot Agent Manager의 내부 모듈인 Robot이 실시간에 체험한 부하량 정보를 바탕으로 효율이 높은 시간대에 대상 쇼핑몰에 접속하여 정보를 수집한다. 수집된 정보는 Information Builder에서 상품에 관한 정보를 추출한 후 가공하여 Index Database에 저장한다.

Service Engine은 Service Agent와 Customer Database로 구성되며 Service Agent는 Search Engine이 구성한 Index Database의 정보를 소비자가 원하는 정보로 문서화하여 서비스해주며 소비자의 개인 특성을 관리하여 개인이 원하는 정보를 특정주기별로 메일 시스템을 이용해 부가서비스를 제공한다.

3.2 Target Server Manager

Target Server Manager는 Server Database에 구축되어 있는 쇼핑몰 서버들의 존재유무와 시간대별 부하량을 측정한다.

Target Server Manager의 구성은 (그림 2)와 같다.



(그림 2) Target Server Manager 구성도

대상 쇼핑몰 서버에 대한 부하량은 매번 평균값으로 저장되며 Robot이 정보수집 시 체험한 부하량과 함께 대상

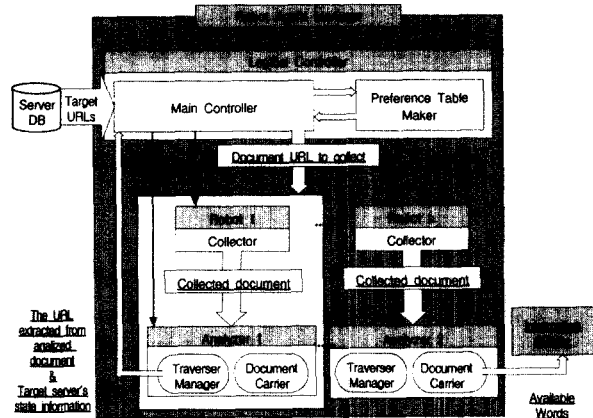
서버의 수집 활동 시작 시간대를 설정하는데 사용된다. 이로서 대상 서버의 급작스러운 변화에 즉각 대처할 수 있는 능력을 가지게 되며 대상 쇼핑물 서버에 심한 부하를 주지 않는 범위 내에서 정보수집을 수행할 수 있게 된다.

또한 Target Server Manager는 대상서버들의 URL에 대한 정확성 여부를 판단하고 대상서버가 서비스 중인지 아닌지를 결정하며 대상서버가 제공하는 정보의 갱신주기를 판단하여 정보수집 작업에 대한 상태정보를 갱신주기에 맞추어서 필요한 정보가 빠른 시간 내에 갱신될 수 있게 한다.

Target Server Manager에 의해 시간대별 효율이 높은 순으로 서버들의 리스트를 생성하며, 각 서버들은 한 시간당 최대로 배정될 수 있는 부하량의 합산 최고치를 기준으로 균등히 분배되는데 만약 한 시간에 너무 많은 부하량이 배정되면 배정된 서버 중 최하순위의 서버를 다음 최적시간대에 배정하며, 수집활동을 하면서 해당 시간의 서버에 대해 모든 문서를 수집하였을 경우 다음 시간의 서버들에 대해 수집활동을 시작한다. 그리고 만약 해당 시간에 모든 문서를 수집하지 못했을 경우에는 다음시간대의 작업 후 남은 시간에 수집활동을 계속하게 된다.

3.3 Robot Agent Manager

Robot Agent Manager는 (그림 3)과 같이 Logical Controller와 Robot 과 Analyzer의 세 가지 모듈로 구성된다.



(그림 3) Robot Agent Manager 구성도

Robot Agent Manager는 본 시스템의 핵심부분으로 대상 쇼핑물에서 상품에 대한 정보를 수집하고 분석하는 역할을 담당한다.

Logical Controller 모듈은 현재시간대에 할당되어진 검색 대상 쇼핑물들을 Server Database에서 전달받아 쇼핑물들의 우선순위 테이블을 만들고 이에 따른 로봇엔진과 분석엔진의 수행을 총괄 관리한다.

Robot 모듈은 Main Controller에 의해 제어되며 하나의 로봇이 하나의 URL을 담당하여 문서를 수집하게 된다.

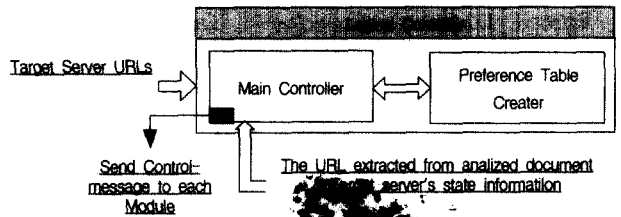
Robot 모듈이 수집한 문서는 Analyzer 모듈에 의해 문서

내에 포함된 링크들이 모두 추출되고 사용자에게 제공하기 위한 정보로 가공하기 위해 Information Builder 모듈에 전달된다.

추출된 링크는 다시 Logical Controller에게 전달되어 위의 과정이 반복된다.

3.3.1 Logical Controller

Logical Controller는 (그림 4)와 같이 Main controller와 Preference Table Creator로 구성된다.



(그림 4) Logical Controller 구성도

Main Controller는 Target Server URL들을 받아 Preference Table Creator를 이용하여 해당 시간대에 효율이 가장 뛰어난 쇼핑물을 기준으로 정보수집을 수행한다. 실제 수집 활동을 하는 중에 변동된 대상 서버의 상태를 확인하여 Server Database의 대상서버의 정보를 업데이트하며 검색대상 서버의 급격한 변화가 발생하면 Robot 과 Analyzer 모듈의 전반적인 설정사항을 동적으로 변경하고 조절한다. 즉, 정보 수집 수행 후부터 부하량이 급격히 증가하게 되면 Main Controller는 Robot의 수를 감소시키고, 문서의 수집 효율이 급격히 증가하면 Robot의 수를 증가시켜 보다 짧은 시간에 정보를 수집할 수 있게 제어한다.

Preference Table Creator는 (그림 5)와 같이 Target Server URLs의 정보와 Robot이 수집 시 체험한 실시간 정보를 이용하여 검색대상의 URL들의 우선순위를 결정하여 Main Controller에게 돌려준다.

검색 대상 URL			
No	URL	속도	최근상태
1	203.253.177.203/~cbs/index.html	20	1000
2	www.wowbook.com/center/center.asp	185	200
3	kr.dfr.yahoo.com/Entertainment/Virtual_Cards/	85	200
4	www.lyungnam.ac.kr/Graduate/Graduates/Gra_intro.htm	46	1000
5	www.mbc.co.kr/NETOP/0004/magazine/	120	200
6	www.cbs.pe.kr/cbs/index.htm	12	200

검색완료된 URL			
No	URL	속도	최근상태
1	kr.broadcast.yahoo.com/		20000422
2	www.mbc.co.kr/News/track/newsdesk@home.html		20000422
3	203.253.177.203/~cbs/index.htm		20000422
4	www.lyungnam.ac.kr/Graduate/Graduates/Gra_intro.htm		20000422
5	www.mbc.co.kr/NETOP/0004/magazine/		20000423
6	http://www.wowbook.com/center/center.asp		20000423

우선순위에 결정된 URL			
No	URL	속도	최근상태
1	www.cbs.pe.kr/cbs/index.htm		
2	kr.dfr.yahoo.com/Entertainment/Virtual_Cards/		
3	www.mbc.co.kr/NETOP/0004/magazine/		
4	www.wowbook.com/center/center.asp		

(그림 5) Preference Table Creator의 우선 순위 결정

서버들의 우선순위 결정 기준사항은 Target Server

Manager에 의한 시간대별 부하량을 70%, 로봇이 실시간으로 체험한 부하량에 대하여는 30%의 가중치를 부여하여 시간대별 점수를 계산한 후 그 점수에 따라 가장 효율적인 시간대에 대해 우선권을 높게 주는 방법으로 우선순위를 결정하게 된다. 각 서버들마다 효율적인 시간대는 적어도 하나는 존재하며 하루의 주기 내에 균등히 분배되어 모든 등록된 쇼핑몰에 대해 하루 내에는 그 정보가 갱신될 수 있음을 보장한다.

3.3.2 Robot

Robot은 Logical Controller로부터 Target URL을 받아 문서를 수집하고 그 문서를 수집하는 당시의 대상 서버의 정보를 반환한다.

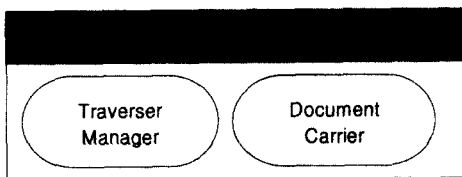
반환하는 대상서버의 정보로는 대상 서버의 로봇 체험 부하량과 대상 서버의 현재 URL에 대한 서비스 여부 등이다. 대상서버의 상태정보는 논리제어기에서 우선순위 테이블의 생성에 실시간으로 반영되어 급변하는 네트워크 상황에 즉각적으로 대처할 수 있게 된다.

Robot은 Thread로 작동하며 수시로 변하는 대상 쇼핑몰 서버에 대한 Thread수는 Logical Controller에 의해 실시간으로 조절된다.

정보수집의 효율이 좋은 경우 Logical Controller에 의해 Robot의 동작 수는 증가하여 짧은 시간에 더 많은 정보를 수집할 수 있게 하고, 대상 쇼핑몰 서버의 부하량이 급속도로 증가하면 Logical controller는 자동으로 Robot의 수를 줄여 대상 서버의 부하를 초래하지 않는 범위에서 정보수집이 이루어 질 수 있게 한다.

3.3.3 Analyzer

Analyzer는 (그림 6)과 같이 Traverser Manager와 Document Carrier로 구성된다.



(그림 6) Analyzer 구성도

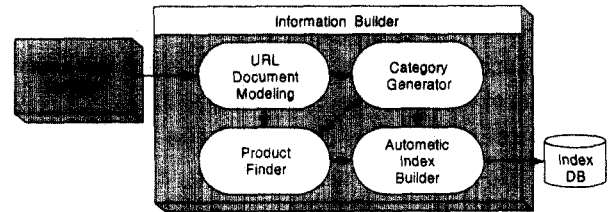
Analyzer는 로봇의 수만큼 Thread로 동작하면서 로봇이 수집해온 문서를 분석하여 다른 대상URL을 추출하고 원하는 정보들을 분류하는 역할을 담당한다.

Traverser Manager는 분석된 문서에서의 URL과 대상 서버의 부하정보량 등의 대상서버 상태정보를 Logical Controller에게 전달한다.

Document Carrier는 문서내의 Hyper-link들을 제거한 내용을 Information Builder에게 전달하여 준다.

3.4 Information Builder

Information Builder의 구성도는 (그림 7)과 같다.



(그림 7) Information Builder 구성도

Information Builder는 Robot Agent Manager의 Words Collector가 수집한 상품 정보들을 동일 상품, 동일 Category별로 분류하고 통일성을 가지는 정보로 가공한다. 상품 정보의 가공처리는 모두 자동적으로 처리되는데 기존의 정보로는 유추할 수 없는 상품들에 대해서는 관리자가 수정을 할 수 있도록 한다.

URL Document Modeling 모듈은 수집한 URL 문서에 대한 Modeling을 통하여 상품정보와 가격정보의 추출에 대한 효율을 증대시킨다.

Product Finder 모듈은 Modeling된 결과를 가지고 해당 쇼핑몰에서 판매하는 상품정보와 가격정보를 찾는다.

Category Generator 모듈에서는 URL Document Modeling 모듈이 Modeling한 결과를 토대로 해당 Category를 검색하여 선정하고 판단이 불가능할 경우 미등록 Category로 분류하여 관리자의 판단에 의해 분류될 수 있도록 한다.

Automatic Index Builder는 Product Finder 모듈이 추출한 상품정보와 가격정보, 그리고 Category Generator가 구분한 Category를 기준으로 색인하여 Index Database에 저장한다.

3.5 Service Agent

Service Agent는 사용자와 직접 관련된 모듈인데 Service Agent 모듈은 미리 구성된 Index Database의 정보를 사용자에게 제공하기 위한 Web문서로 자동 생성하여 주는 모듈이다.

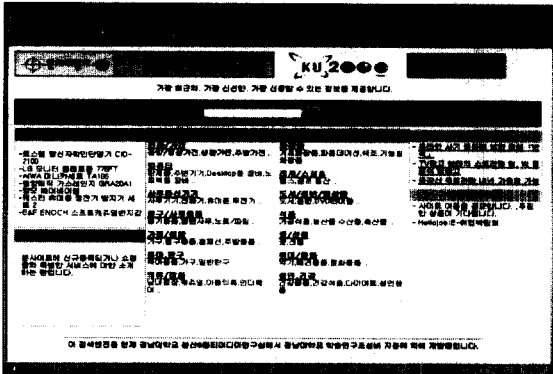
세부기능으로 미리 구분된 Category별로 분류하여 같은 분류의 상품들을 가격이 싼 제품부터 비싼 제품까지 일률적으로 한 페이지에 나타날 수 있게 구성하며 Index Database에 갱신된 시간대를 기준으로 정확도를 표시해준다.

상품의 갱신이 대상 서버의 오류에 의해 지연되었을 경우에는 그 상품에 대한 정보와 함께 정확도는 낮게 표현되어 사용자가 직접 재확인 할 수 있도록 한다.

그리고 Service Agent는 Customer Database에 저장된 고객 정보를 바탕으로 고객의 관심부분에 관련된 상품에 관한 정보를 메일로 발송하여 주는 서비스를 제공한다. 메일은 특정 주기로 발송되게 되며 고객의 회원 등록 시 작성된 관심부분을 바탕으로 제품의 최저가 쇼핑몰의 정보만 제공되어진다.

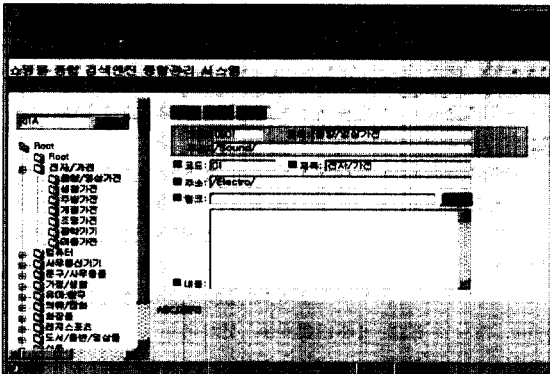
4. 구현 및 평가

사용자 인터페이스에 관련된 모든 웹 문서는 PHP언어를 이용하여 구현하였으며, PHP언어는 대부분의 운영체제 및 데이터베이스를 지원할 수 있고 타 언어와 달리 서버에 독립적인 프로세스를 생성시키지 않으므로 서버에 대한 자원 부담을 줄일 수 있어서 속도가 빠른 것으로 평가되고 있다.

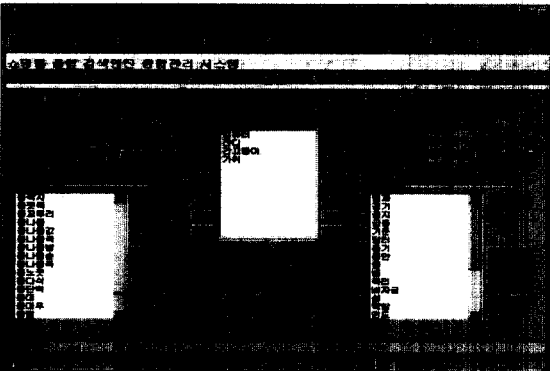


(그림 8) 쇼핑물 종합 검색 시스템

(그림 8)은 제안 시스템이 생성한 쇼핑물 종합 검색 시스템의 화면이다.



(그림 9) Category관리



(그림 10) 단어사전관리

관리 모듈들은 JAVA Applet과 PHP언어로 구현하였으며, 원격지에서 본 제안시스템의 제반사항을 관리할 수 있는 기능을 제공한다.

(그림 9)는 Category 관리 화면이며 생성된 Category에 대해서 신규, 수정, 삭제 작업을 수행한다. 원격지에서 생성된 자료에 대한 Category 관리가 가능하다.

(그림 10)은 쇼핑물 웹 문서를 분석하기 위한 단어사전 관리 화면이다.

제안시스템의 성능분석을 위한 Simulation 환경은 <표 1>과 같다.

<표 1> Simulation 환경

대상쇼핑물 갯수	표본 쇼핑물 서버 1 곳 실제 존재하는 서버 49 곳
사용 운영체제	Apache 1.3.11
사용 프로세스	Pentium 166 MHz

표본 쇼핑물 서버는 제안시스템의 정보갱신 속도를 확인해 보기 위해 간단히 만든 것이다.

A. 검색 대상 서버리스트

①	203.253.***.***	← 표본 쇼핑물
②	211.192.139.50	
③	211.239.89.170	
④	210.217.220.142	

B. 쇼핑물별 점수값

①	56	64	76	43	87	45	34	23	34	31	45	65
②	34	63	24	35	23	34	32	21	66	72	70	58
③	45	34	36	39	81	62	52	47	25	27	26	25
④	65	69	57	55	45	32	41	57	73	72	67	52

C. 시간대별 할당된 서버리스트

1												
2					②	④		①	③			

(그림 11) 검색대상 리스트와 생성된 서버 정보 테이블

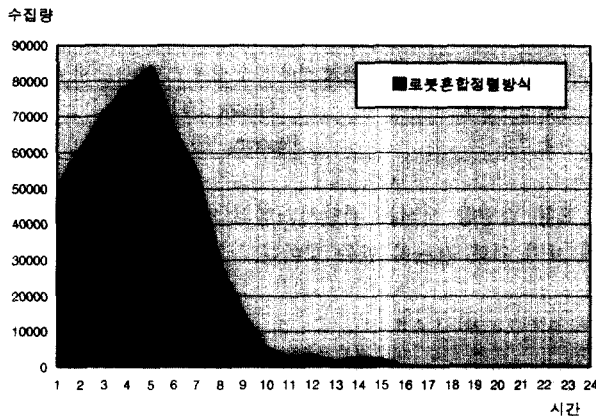
(그림 11. A)는 Simulation을 하기 위한 서버 목록중 샘플을 추출한 것이고 (그림 11. B와 C)는 Simulation한 결과 중 샘플서버에 관한 정보를 정리하였다.

(그림 11. B)처럼 ①번 서버는 8~9시 사이에 가장 점수가 좋으며 ②번 서버는 5~6시 사이, ③번 서버는 9~10시 사이, ④번 서버는 6~7시 사이에 점수가 좋게 나왔다. 이를 기본으로 각 시간대별로 서버들을 배치하면 (그림 11C)와 같은 리스트를 구할 수 있으며 ②번 서버의 경우 8~9시 사이에 놓이면 시간대 총점이 다른 시간대에 비해 높게 된다. 이는 다시 재조정되며 ②번 서버의 두 번째 최적 시간대를 찾아 그 시간대에 할당하게 된다. 이렇게 생성된 리스트를 토대로 해당 시간에 할당된 서버를 기준으로 효율적인 문서 수집을 수행할 수 있게 된다. 이는 다시 재

조정되며 ②번 서버의 두 번째 최적 시간대를 찾아 그 시간대에 할당하게 된다. 이렇게 생성된 리스트를 토대로 해당 시간에 할당된 서버를 기준으로 효율적인 문서 수집을 수행할 수 있다.

제안시스템의 검색 엔진 효율을 점검하기 위해 네트워크상 임의의 서버 50곳을 선정, 새벽 1시부터 500 Gbyte의 정보를 수집한 결과 (그림 12)에서와 같은 결과를 얻을 수 있었다.

여기에서 소요시간은 약 15시간이며, 로봇이 수집활동을 하고 있는 동안에도 보다 효율적인 사이트를 우선 처리함으로써 정보 수집 면에서 더욱 효율적임을 알 수 있다.



(그림 12) 정보 수집량 비교 그래프

5. 결 론

기존의 쇼핑몰 가격비교 사이트는 정보의 갱신이 길게 소요되는 관계로 효율적인 정보를 제공하지 못한다.

본 논문에서 제시한 주문형 검색엔진은 짧은 시간에 많은 문서를 수집할 수 있는 로봇 혼합 정렬 알고리즘을 활용하고 검색엔진의 상황을 동적으로 변경할 수 있는 동적 로봇 에이전트를 이용하였다.

대상 쇼핑몰 서버에 부하를 초래하지 않으면서 하루 내에 상품 정보가 갱신될 수 있으며, 정보수집단계와 정보제공단계를 분리함으로써 시스템의 분산이 가능하다.

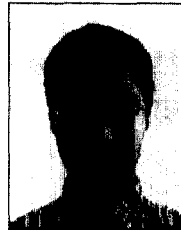
앞으로, 검색 대상 서버들의 부하량 관리에 있어서 각각의 라우터마다의 연결경로를 고려하여 우회하는 방법과, 검색시스템과 원격관리모듈간 작업에서의 보안관련 사항에 대한 지속적인 연구가 요구된다.

참 고 문 헌

[1] 박기철, "전자상거래를 위한 전자결제 시스템에 관한 연구", 홍익대학교 석사학위, 2000.
 [2] 이윤석, "전자상거래 특성을 고려한 웹서버 제작에 관한 연구", 홍익대학교 석사학위, 1999.
 [3] 鄭龍勳, "혼합 정렬 스케줄링을 이용한 검색 로봇엔진", 경남대 석사학위, 1999.
 [4] 박민우, "로봇 에이전트의 설계와 구현", 마이크로소프트웨어, 1996.

[5] 고희대, 유재수, 김병기, "효율적인 정보 검색 시스템 구축을 위한 새로운 프로세스 구조", 한국정보처리학회논문지, 제 14 권 제1호, 1997.

[6] 金 晙, "Design and Implementation of Real-time Control Search Engine for Local Information Network," 경남대 석사학위, 1998.



김 성

e-mail : bgbest@dmlab.net

2000년 경남대학교 컴퓨터공학과 졸업 (공학사)

2000년~현재 경남대학교 컴퓨터공학과 석사과정

관심분야 : 에이전트기술, 웹 프로그래밍, 멀티미디어



박 철 우

e-mail : trats@dmlab.net

2000년 경남대학교 컴퓨터공학과 졸업 (공학사)

2000년~현재 경남대학교 컴퓨터공학과 석사과정

관심분야 : 분산처리시스템, 에이전트기술



이 충 석

e-mail : cslee@sslslab.net

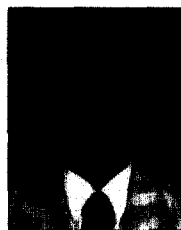
1997년 경남대학교 컴퓨터공학과 졸업 (공학사)

1996년~1997년 日本 Core Dump Group (NTT: 개발연구원)

1997년~1998년 서울 PowerNet(주) LG Soft : 프로그램 개발

2001년~경남대학교 컴퓨터공학과 석사졸업

관심분야 : 전자상거래, 에이전트기술, 분산시스템



박 규 석

e-mail : kspark@hanma.kyungnam.ac.kr

1980년 중앙대학교 전자계산학과 전산학 전공(석사)

1988년 중앙대학교 전자계산학과 전산학 전공(박사)

1982년~현재 경남대학교 정보통신공학부 교수

1992년~1996년 경남대학교 전산정보원 원장

1995년~1996년 한국정보과학회 이사, 영남지부장

1999년~현재 경남대학교 정보통신연구소 소장

관심분야 : 분산처리시스템, 정보통신 소프트웨어, 멀티미디어 시스템