사용자 맞춤의 문서 요약을 제공하는 정보 여과 에이전트 시스템

조영희, 김교정
숙명여자대학교 정보과학부

The Information Filtering Agent System with a Customized Document Summary

Yeong-Hui Jo, Kio-Chung Kim,
Dept. of Computer Science, Sookmyung Women's University

요약

현재의 정보 과적재(information overload) 상황은 대량의 정보 가운데서 사용자의 관련 정보에 대한 요청을 도와 불필요한 정보로부터 막기 위한 도구가 매우 필요한 실정이다.

이러한 도구중 대표적으로 사용되는 웹 검색 엔진과 같은 정보 검색 시스템의 단점은 적합한 검색 용어를 선택해야만 한다는 점과, 결과에 대한 효율적인 요약이 제공되지 않는다는 점이다.

따라서 본 논문에서는 이러한 검색 엔진에서의 단점을 보완하여 사용자를 정보 과잉 상황에서의 불필요한 정보로부터 보호하기 위해, 사용자의 프로파일을 기반으로 하여 정보를 개인화된 요약과 함께 제공하는 정보 여과 에이전트(information filtering agent)인 '사용자 맞춤의 문서 요약을 제공하는 정보 여과 에이전트 시스템'을 제안한다.

1. 서론[1)]

최근 급변하는 정보화 사회 속에서 정보통신 분야 등의 발달에 힘입어 컴퓨터 사용자에게 사용 가능한 정보의 양은 그 방대함이 다루기 힘들 정도에까지 이르렀다. 이러한 현재의 정보 과적재(information overload) 상황은 원하는 정보 획득을 위해 많은 시간 및 노력의 소비를 유발한다. 따라서 사용자의 관련 정보에 대한 요청을 도와 대량의 정보 가운데서 불필요한 정보로부터 막기 위한 도구가 매우 필요한 실정이며, 검색 엔진(Search Engine), 인텔리전트 에이전트(intelligent agent) 등이 대표적으로 사용되고 있다[1][2][3][4][5].

고정된 문서 집합에서 관련 문서를 검색하기 위해 이용하는 검색 엔진은 검색 내용은 줄여주지만 그 내용을 파악하기 위한 시간과 노력이 역시 필요하다. 또한 사용자는 자신의 필요 정보에 대해 매우 확실한 이해를 가지고 있는 것으로 가정되지만, 실제로는 자신의 필요한 정보를 공식적인 언어로 정확히 표현하는데 어려움을 느끼고 있다[5]. 즉, 지금까지 대부분의 웹 검색 엔진은 정확율(precison)[2)]보다는 재현율(recall)[3)]을 최대화시키고, 속도를 높이는데 초점을 두어 설계되어 왔다. 따라서 사용자는 여전히 관련 정보를 찾기 위해 대량의 정보를 살펴보아야만 한다[6].

이러한 검색 엔진에서의 문제점을 보완하여 사용자를 정보 과잉 상황에서의 불필요한 정보로부터 보호하기 위해, 다양한 출처로부터 정보를 모은 후 사용자의 개인적 선호도에 기반하여 여과된 정보를 사용자에게 제시하는 정보 여과 에이전트가 이용된다[7]. 하지만 정보 여과 에이전트에서 제공하는 잠재적으로 관심이 있는 모든 텍스트를 사용자가 다루는 일 또한 많은 시간을 필요로 한다.

이에 본 논문에서는 이렇게 정보 여과를 통해 제공되는 문서에 대해 텍스트 마이닝(Text Mining)의 텍스트 요약 기법 및 인공 지능의 여러 기법을 활용하여 사용자 프로파일에 따른 개인화된 문서 자동 요약의 기능을 제공함으로써, 더욱 생산성 및

1) 본 연구는 과기부/KISTEP 지원 여자대학 연구기반 확충사업으로 수행되었슴.

2) 검색된 전체 문서 수 중 관련 문서 수의 비율
3) 전체 관련 문서 수 중 검색된 관련 문서 수의 비율

효율성을 높이고 사용자의 부담을 경감시킬 수 있도록 하는 정보 여과 에이전트 시스템인 '사용자 맞춤의 문서 요약을 제공하는 정보 여과 에이전트 시스템'을 제안한다. 즉, 이 시스템은 사용자의 선호도(preference) 프로파일에 기반하여, 부합하는 정보를 개인화된 요약과 함께 제공하는 정보 여과 에이전트 시스템이다.

2. 관련연구

2.1 웹 검색 에이전트(Web Search Agents)

정보 검색(Information Retrieval, IR)이란 저장된 대량의 문서들로부터 사용자 질의에 적합한 유용한 문서를 검색하여 제공하는 것을 말한다. 이것은 사용자 질의에 의미적으로 관련이 있는 문서를 찾아주는 일이라고 할 수 있다[8][9][10].

인터넷 정보 검색 환경에서 사용되는 웹 검색 에이전트는 모든 웹 사용자의 관심에 부응하기 위하여, 선행적으로 새로운 URL을 발견하고 문서를 검색하여 목록을 만드는 기능을 한다. 이러한 의미에서 검색 엔진(search engine)을 하나의 에이전트로 보기도 한다[7].

고정된 문서 집합에서 관련 문서를 검색하기 위해 이용하는 검색 엔진은 검색 내용은 줄여주지만 그 내용을 파악하기 위한 시간과 노력이 역시 필요하다. 또한 사용자는 자신의 필요 정보에 대해 매우 확실한 이해를 가지고 있는 것으로 가정되지만, 실제로는 자신의 필요한 정보를 공식적인 언어로 정확히 표현하는데 어려움을 느끼고 있다[5]. 즉, 지금까지 대부분의 웹 검색 엔진은 정확율(precison)보다는 재현율(recall)을 최대화시키고, 속도를 높이는데 초점을 두어 설계되어 왔다. 따라서 사용자는 여전히 관련 정보를 찾기 위해 대량의 정보를 살펴보아야만 한다[6].

2.2 정보 여과 에이전트(Information Filtering Agents)

정보 여과(information filtering)는 사용자의 선호도(preference) 프로파일과 부합하는 정보의 선택적 보급이라고 정의된다[6]. 때로는 정보 검색의 부속물로 정의되기도 하는데, 사용자 질의로부터 저장된 추가의 정보를 사용하여 불필요한 문서를 제거함으로써, 검색된 문서 집합을 정제하기 때문이다[6].

정보 여과 에이전트는 웹 검색 에이전트(Web Search Agent)와 마찬가지로 대량의 온라인 정보를 줄임으로써 정보 과적을 다루는데, 낮은 정확율의 문제점을 보완하고자 사용자 정보 프로파일에 따라 다룬다는데 그 차이가 있다. 따라서, 웹 검색 에이전트(web search agent)는 사용자가 관심을 갖는 특정 웹 싸이트를 찾는데 유용한 반면, 정보 여과 에이전트(information filtering agent)는 다양한 정보 근원을 이용하여 사용자가 관심을 갖는 특정 내용을 찾는데 유용하다. 즉, 정보 여과 에이전트는 정보를 다양한 근원지로부터 모은 후, 사용자의 개인적 선호도에 기반하여 여과된 정보를 사용자에게 제시한다[7].

하지만 정보 여과 에이전트의 경우에도 사용자에게 제시되는 모든 텍스트를 다루는데 많은 시간과 노력이 필요하다는 문제점은 여전히 남는다.

이에 본 논문에서는 사용자 프로파일을 이용한 개인화된 문서 요약을 함께 제공하도록 함으로써 이러한 단점을 보완하고, 이를 통해 더욱 생산성 및 효율성을 높이고 사용자의 부담을 경감시킬 수 있도록 하고자 한다.

2.3 텍스트 요약(Text Summarization)

텍스트 요약은 텍스트 데이터 마이닝 기능 중의 하나로서, 본래의 문서가 가지고 있는 기본적인 의미를 유지하면서 문서의 길이나 정보의 복잡도 등을 줄이는 작업이라고 할 수 있다[11][18].

이러한 요약 기능은 사람들이 매일 다루어야하는 정보의 과적재(information overload)를 피하는데 매우 유용하며, 따라서 텍스트 자동 요약은 정보 검색 환경에서 그 필요성이 매우 절실하다[10][18]. 즉, 이를 통해 사용자는 쉽게 문서 접근에 대한 결정을 할 수 있고, 또한 여러 문서의 내용을 쉽고 빠르게 파악 가능하다[11].

하지만 비구조적인 텍스트의 요약은 여전히 힘들고 정의하기 어려운 문제로 남아있으며[3][10], [18]에서는 자동요약의 난이성을 다음과 같이 서술하였다.

첫째, 요약문의 질(quality)이 요약문을 읽는 사람의 요구에 따라 좌우되어, 모든 사람의 요구에 맞는 요약문을 작성할 수는 없다는 점이다.

둘째, 문서에서 다양하게 쓰이고 있는 지시어를 처리하기가 어렵다는 점이다.

셋째, 요약된 내용의 객관적인 평가 결과의 도출이 쉽지 않다는 점이다.

마지막으로 요약문을 생성하는데 있어서 최적 길이에 대한 문제를 들고 있는데, 요약문의 길이가 길수록 사용자에게 많은 정보를 줄 것은 자명하지만 그만큼 요약문의 가치는 저하된다는 점이다.

본 시스템에서는 이러한 첫 번째와 네 번째의 문제점을 보완할 수 있도록 하였다. 즉, 구축된 사용자 프로파일에 기반한 요약문은 사용자의 구미에 맞는 요약을 제공할 수 있으며, 사용자와의 상호작용에 의해 요약문으로 제공할 문서의 길이를 비율로 입력받음으로써 요약의 길이 문제 역시 보완할 수 있도록 하였다.

3. 사용자 맞춤의 문서 요약을 제공하는 정보 여과 에이전트 시스템

본 시스템은 검색 엔진에서의 단점을 보완하여 사용자를 정보 과잉 상황에서의 불필요한 정보로부터 보호하도록 하는 정보 여과 에이전트 시스템으로서, 정보 여과를 통해 제공되는 문서에 대해 사용자 프로파일에 따른 개인화된 문서 자동 요약을 제공하도록 하여 더욱 생산성 및 효율성을 높이고 사용자의 부담을 경감시킬 수 있도록 하는 개선된 정보 여과 에이전트 시스템이다.

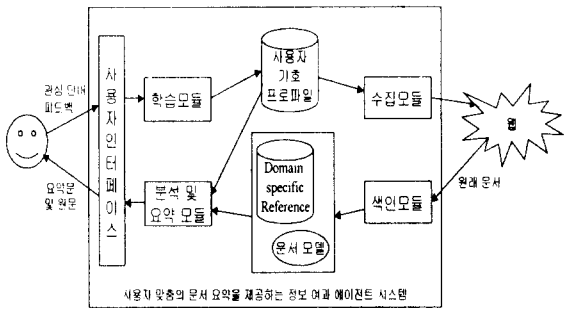
이를 통해 정보 검색(Information Retrieval)에서의 적합한 검색 용어 선택 문제와 관련한 사용자 불필요 문서 수를 감소시키는 것은 물론, 사용자가 문서의 접근 결정 및 내용 파악을 더욱 빠르고 쉽게 할 수 있도록 돕는다. 즉, 사용자에게 정보여과로 제공된 문서를 다시 쉽게 여과할 수 있도록 하는 최상의 정보 여과의 효과를 제공한다.

3.1 시스템의 구조

정보 여과 모델에는 일반적으로 정적 여과 모델(Static Filtering Models)과 동적 학습 여과 모델(Dynamic Learning Filtering Models)이 있다[13].

먼저 정적 여과 모델(Static Filtering Models)은 학습을 수행하지 않는 모델로서, 중대한 수정을 제외하고는 사용자 선호도 변화를 반영하지 않는다. 예로는 사용자 선호도의 규칙 기반 모델을 들 수 있는데, 이 모델은 규칙 집합의 완전한 습득이 필요하기 때문에 높은 초기 셋업 비용이 들게 된다. 이에 비해 동적 학습 여과 모델(Dynamic Learning Filtering Models)은 계속 업데이트를 수행함으로써 쉽게 사용자 선호도를 반영할 수 있는데, 수행이 이루어지는 동안 사용자 피드백이 반영되어 여과와 학습이 동시에 일어나는 것이라고 할 수 있다.

본 논문에서 제안한 시스템은 동적 학습 여과 모델에 해당하며, 그 구조는 <그림 1>과 같다.



<그림 1> 시스템의 구조

사용자는 사용자 인터페이스를 통해 관심 단어, 뉴스 서버, 뉴스 그룹, 피드백, 문서 요약 비율 등의 기호 정보를 입력하게 되고, 시스템을 통해 관련 있을 것으로 추정되는 문서를 원문과 요약으로써 제공받게 된다.

3.1.1 각 모듈의 기능

수집 모듈은 사용자 기호 프로파일 데이터베이스를 기반으로 특정 웹 싸이트나 뉴스그룹으로부터 새로운 웹 문서를 수집한다. 색인 모듈은 문서 모델링을 담당하는데, 즉 수집 모듈로부터 받은 문서들을 색인하여 문서 모델 데이터베이스를 만든다.

학습 모듈은 사용자 프로파일의 관리를 수행하는데, 프로파일의 생성, 수정, 삭제를 담당한다. 이를 위해 먼저 사용자로부터 관심 단어를 입력받아 사용자 프로파일을 구축한다. 이렇게 구축된 프로파일을 기반으로 문서를 추천하고, 사용자로부터 문서의 관련성에 대해 직접 응답을 받는 방식으로 하여 연관성 피드백 학습을 수행하여 사용자 프로파일의 업데이트가 계속 이루어진다.

분석 모듈의 목적은 사용자의 기호에 적합한 정보를 제공하는데 있다. 따라서 분석 모듈은 학습 모듈에서 구축한 사용자 프로파일을 이용하여 새로운 문서들 중에서 가장 적절한 것들을 추론하는 기능을 담당하고 있다. 즉, 구축된 사용자 모델과 문서 모델 데이터베이스를 이용하여 서로 비교함으로써 문서의 사용자에 대한 관련성을 측정하고 이에 따라 문서 제공 여부를 결정한다.

요약 모듈은 분석 모듈로부터 사용자에게 제공하기로 결정된 문서에 대해 문서 요약을 위한 수행을 하는데, 이를 통해 사용자에게 원래 문서와 함께 요약을 제공하게 된다. 사용자의 구미에 맞는 문서 요약을 위해서 사용자 모델이 이용되며, 사용자 요약을 제공할 때는 사용자가 선택한 비율에 따라 요약의 길이가 결정된다.

3.1.2 시스템의 수행 과정

본 시스템은 다음과 같은 과정으로 수행이 이루어진다. 우선 사용자의 관심 사항을 파악하여 사용자 프로파일을 구축하는 사용자 모델링 단계와 검색된 문서를 표현하기 위한 문서 모델링 단계가 수행되고, 다음으로 검색된 문서가 사용자의 관심 문서인지를 판단하기 위해 사용자 모델과 문서 모델 사이의 유사도 측정이 수행된다. 유사도가 일정 임계치 이상인 선택 문서에 대해서는 요약을 수행하여 원문과 함께 사용자에게 제공하게 되며, 사용자의 피드백을 받아 학습을 통해 변화하는 사용자의 관심에 적응해 나가게 된다.

<그림 2>는 이러한 과정을 순서별로 자세히 나타내고 있다.

1. 사용자는 짧은 개인 정보, 기호(preference), 요약 문서의 길이 비율을 입력하고, 학습 모듈은 이를 바탕으로 사용자의 초기 프로파일을 생성한다.
2. 색인 모듈은 컨텐츠 제공자(웹 싸이트, 뉴스 그룹 등)로부터 지정된 토픽의 새로운 문서를 모은 후, 토큰화(tokenization), 불용어 제거, 어간 복구(stemming) 등의 문서 선처리 과정을 수행하고, 이를 통해 문서 모델 및 문서 집합 정보 데이터베이스를 구축한다.
3. 분석 모듈은 초기 사용자 프로파일과 문서 모델 데이터베이스 사이의 유사도를 측정함으로써 사용자와 관련이 있다고 추정되는 개인화된 문서를 선택한다.
4. 요약 모듈은 분석 모듈에서 선택된 개인화된 문서와 사용자 기호 프로파일 등을 이용하여 개인화된 요약을 생성하고, 사용자에게 원문과 함께 제공한다.
5. 다시 학습 모듈은 제공한 문서에 대한 연관성 피드백을 사용자로부터 입력받아 관련 문서를 학습함으로써 사용자 프로파일을 업데이트한다.
6. 업데이트된 사용자의 선호도를 바탕으로 새로운 문서에 대해 추천하고, 사용자와의 상호작용을 통해 다시 사용자의 기호를 학습한다.
7. 이에 따라 사용 횟수가 증가할수록 사용자의 기호와 매치되는 좀 더 개인화된 내용을 제공할 수 있다.

<그림 2> 시스템의 수행 과정

3.2 사용자 모델링

새로운 문서에 대한 사용자와의 관련성 여부 측정 및 결과의 우선 순위 결정에 이용되는 사용자 기호의 프로파일 정보는 사용자 모델링을 통해 구축된다. 이 때, 사용자의 기호는 북마크 리스트 등의 지속적인 사용자 기호(persistent user preference)와 특정 탐색 세션 동안 찾아간 링크 등의 현재의 사용자 기호(current user preference) 등으로 분류된다[7].

사용자의 정보에 대한 기호(user preference)를 결정하는데는 특정 정보와 관련된 사용자 행동의 관측이 필요한데, 사용자의 정보 기호를 얻기 위한 방법으로는 사용자에게 직접 물어보는 방법 또는 사용자의 정보 사용을 관측하는 간접적인 방법이 사용된다[7]. 본 연구에서는 사용자의 기호 학습을 위해 사용자로부터 직접 관심 사항을 입력받는 방법을 택하였는데, 보다 정확한 정보를 쉽게 수집하기 위해서였다.

사용자로부터 직접 입력을 받는 내용으로는 초기 관심 질의어와 시스템으로부터 제공받은 문서에 대한 사용자의 연관성 피드백 등이 있다. 초기 관심 질의어는 초기 사용자 프로파일 구축에 사용되며, 연관성 피드백은 구축된 사용자 프로파일을 학습을 통해 계속 업데이트하는데 사용된다.

3.2.1 초기 사용자 프로파일(user profile) 구축

사용자 프로파일은 하나 이상의 사용자 선호의 질의어로 구성되는데, 이를 위해 먼저 사용자의 관심 사항을 단어로 입력받아 벡터 표현법으로 변환하여 초기 사용자 프로파일을 생성한다.

이를 통해 사용자 프로파일이 <질의어, 질의어의 가중치>의 집합으로 표현되며, 이렇게 벡터로 표현한 것은 문서와 질의 모두를 벡터로 표현하는 벡터 표현법을 이 시스템에서 사용하기 때문이다. 즉, 문서 및 질의가 <그림 3>과 같이 표현되며, 이를 통해 문서와 질의간의 유사도를 측정하는데 유용하다. 질의와 문서 가중치는 벡터 크기와 방향의 기반이 된다.

$$D_i = w_{d_{i1}}, w_{d_{i2}}, \ldots, w_{d_{in}}$$

$$Q = w_{q1}, w_{q2}, \ldots, w_{qt}$$

$w = 0$ if a term is absent

w_{di} : 문서 D_i에서의 용어 d_i에 대한 중요도 가중치

w_q : 질의 Q에서의 용어 q에 대한 중요도 가중치

<그림 3> 문서 및 질의의 벡터 표현법

3.2.2 연관성 피드백(relevance feedback) 학습을 이용한 사용자 프로파일 업데이트

3.1절에서 언급하였듯이, 본 시스템은 동적 학습 여과 모델로서 사용자 프로파일에 대해 계속 업데이트를 수행함으로써 쉽게 사용자 선호도를 반영할 수 있고, 수행이 이루어지는 동안 사용자 피드백이 반영되어 여과와 학습이 동시에 일어나게 된다.

이를 위해 본 시스템에서는 연관성 피드백 학습을 수행하는데, 이 학습에는 자동적인 방법과 사용자가 관련 문서를 선택하는 방법이 있다. 본 시스템에서는 질의와 문서사이의 유사도 측정을 통해 사용자에게 제공된 문서에 대해 사용자로부터 관련성 판정을 직접 받아 학습을 수행하도록 한다.

이를 통해 기존 질의를 수정하게 되는데, 즉 관련 문서로부터 용어를 추출하여 질의에 추가하거나, 기존의 질의인 경우 가중치를 재계산하게 된다[14]. 이렇게 연관성 피드백을 통해 새로운 용어와 관련한 질의를 확장할 수 있으며, 질의 용어의 가중치를 재계산할 수 있다. 이러한 지속적인 학습을 통해 변화해 가는 사용자 기호의 프로파일을 형성하게 된다.

검색된 문서 중 관련이 있다고 판정된 문서와 비슷한 용어 또는 문서를 제시하는 연관성 피드백의 방법외에도, 질의를 재형성하는 방법으로 용어와 유사한 질의 용어를 제시하는 시소러스를 이용한 확장 방법이 있지만 본 논문에서는 적용하지 않았다[14]. 현재 연관성 피드백은 정보 검색에서 검색을 정제하는데 가장 중요한 방법중 하나로 알려져 있다[15].

이러한 연관성 피드백은 여러 가지 변형이 가능한데, 즉 사용자 프로파일에 대해 관련 문서의 용어로부터 포지티브 가중치를 줄 수도 있고, 비관련 문서의 용어를 이용하여 네거티브 가중치를 줄 수도 있다[14].

벡터 공간 모델에서 사용자 적합성 평가를 이용하여 초기 질의어를 확장하는 방법들의 기본적인 형태는 Rocchio 알고리즘으로, <그림 4>와 같다[14]. 이를 통해 사용자의 적합성 평가를 직접 반영하여 사용자 프로파일의 용어 중요도 가중치를 수정하게 된다. 이는 벡터 공간에서 사용자의 정보 요구를 정확하게 표현해 줄 수 있는 질의 벡터 표현을 위해 크기와 각도를 조절하는 과정이라고 볼 수 있다.

$$Q_1 = Q_0 + \beta \sum_{i=1}^{n_1} \frac{R_i}{n_1} - \gamma \sum_{i=1}^{n_2} \frac{S_i}{n_2}$$

Q_0 : 초기 질의 벡터

R_i : 관련 문서 i의 벡터

S_i : 비관련 문서 i의 벡터

n_1 : 선택된 관련 문서의 수

n_2 : 선택된 비관련 문서의 수

β, γ : 관련 및 비관련 용어의 중요도를 조정하는 역할

<그림 4> Rocchio 알고리즘

본 논문에서도 Rocchio 알고리즘을 통해 관련 문서로부터 새로운 용어를 추가하거나, 용어의 가중치를 재계산하는 방법으로 사용자 프로파일을 업데이트하도록 하였다.

3.3 문서 모델링

문서 모델링은 비구조적인 텍스트를 컴퓨터가 처리할 수 있도록 구조적인 데이터로 변환시키는 과정으로서, 문서의 내용 정보를 효과적으로 표현하는 방법에 관한 것이다.

본 시스템에서는 대표적인 정보 검색의 기법중 벡터 공간 접근법을 사용하였는데, 일반적으로 정보 검색 및 텍스트 학습에서 하나의 문서를 표현하기 위해 벡터 표현(vector representation)이 가장 자주 사용된다. 이 방법에서 문서는 다차원 공간에서의 속성 벡터(attribute vector)로 모델링되며, 이를 통해 질의와 문서간의 부분 매칭 및 랭킹이 가능하다[4][7].

벡터 표현법은 단어 꾸러미 표현(bag-of-words representation)으로도 불리는데, 문서의 모든 단어를 단어 순서 및 텍스트 구조를 사용하지 않고 취하기 때문이다. 즉, 문서 집합에서 각 문서는 나타나는 모든 단어들의 꾸러미로 표현된다[4].

벡터 표현 방법 중에서도 가중치 용어의 벡터(vector of weighted terms)가 많이 사용되는데, 본 시스템에서도 가중치 용어의 벡터를 사용하여 3.2.1절의 <그림 3>과 같이 문서 및 질의를 표현하였다. 즉, 하나의 문서를 그 문서에 나타난 단어들을 이용하여 <용어, 용어의 가중치>의 집합으로 표현하였다. 이를 위해 문서에 대한 색인을 수행하고, 각 색인된 용어의 가중치를 계산하였다.

3.3.1 색인(indexing)

문서의 내용 표현을 위한 용어 선택을 위해서는

문서 텍스트에서 단어나 구를 자동으로 추출하는 자동 색인 과정이 필요하다[10]. 색인 용어 생성 및 가중치 부여의 역할을 하고, 결과 랭크를 부여하는데 중대한 영향을 미치는 색인 기법[9]은 정보 검색의 여러 색인 기법들 중에서 선택적으로 적용 가능하다.

본 논문에서는 많은 언어에서 널리 사용되어지는 단어 기반 색인(word-based indexing) 기법을 사용하였다. 이 기법에 따라 색인 용어 생성을 위해 다음의 과정을 거치게 된다. 첫째, 공란(space) 등을 경계 기호(delimiter)로 이용하여 텍스트의 단어를 인식하고(tokenizing), 둘째, 불용어 리스트(stop list)[4]에 해당하는 단어를 제거한다. 마지막으로 파생 어미(derivational ending)나 굴절 어미(inflectional ending)를 제거하는 어간 회복(stemming)[5]의 과정을 거친다.

3.3.2 용어의 가중치(weight) 계산

용어의 가중치는 정보 여과 과정에서 그 단어의 중요도를 계산하는데 사용된다. 즉, 관련 문서인지 여부를 알아내기 위해 사용자 프로파일 벡터와 문서 벡터 사이의 유사도를 측정하는데 이용된다. 또한 자동 문서 요약을 위한 문서의 대표도를 측정하는데도 이용된다.

가중치 계산은 색인 과정에서 중복 사용된 단어들에 대하여는 그 단어의 빈도수를 기본적으로 사용하였으며, 위치 등의 여러 가지 추가 정보도 사용하였다. 이렇게 출현 빈도수에 따라 단어의 중요도를 결정하는 것은 저자가 키포인트를 만들어내기 위해 특정 단어를 반복하는 경향이 있다는 가정에 근거한 것이다[7].

용어 가중치는 특히 TF[6]×IDF[7] 방법에 의해 각 문서와 문서 전체 집합 안에서 용어의 출현 특성을 고려하여 계산하였다[10]. 이 방법은 다른 문서에서는 낮지만 특정 문서에서는 빈도수가 높은 단어의 경우에 좀더 높은 가중치를 부여하는 방법으로, 한 문서내의 단어 빈도수에 비례하고 단어가 등장하는 문서 수에 반비례한다. 이에 따라 문서 Di에서 용어 T_k의 가중치인 $w_{ik} = tf_{ik}$[8]$\times \log(N$[9]$/n_k$[10]$)$로 계산한다.

[16]은 TF를 intra-cluster 유사도 즉, 개체를 설명할 수 있는 특징으로, 용어가 문서 내용을 얼마나 잘 나타내는가를 측정하는 요소라고 하였으며, 또한 IDF를 inter-cluster 유사도로서, 개체를 좀 더 구별지을 수 있는 특징의 측정 요소라고 하였다.

그런데 문서 Di에서 용어 T_k의 가중치를 $w_{ik} = tf_{ik} \times \log(N/n_k)$로 계산을 하면, 가중치는 문서의 길이에 비례할 수 있다. 이것을 문서의 길이가 길다고 해서 용어의 가중치가 높아지지 않도록 하기 위해, 모든 값이 특정 범위(주로 0과 1사이) 안에 포함되도록 문서 길이에 대해 정규화를 수행하는데, [1], [10] 등에서 유사한 다른 식을 사용하여 수행하고 있다. 본 논문에서는 [14]가 제시한 <그림 5>의 식을 사용하였다.

$$w_{ik} = \frac{tf_{ik}\log(N/n_k)}{\sqrt{\sum_{k=1}^{t}(tf_{ik})^2[\log(N/n_k)]^2}}$$

tf_{ik} : TF, 용어의 출현 빈도수

$\log(N/n_k)$: IDF

N : 전체 문서수

n_k : 용어 T_k가 나타난 문서수

<그림 5> 용어 가중치 계산식

3.4 사용자 모델과 문서 모델 사이의 유사도(similarity) 측정

사용자에게 잠재적으로 관심이 있을 것으로 추정되는 문서를 추천해주기 위해 사용자 모델과 문서 모델 사이의 유사도를 측정한다. 즉, 사용자 모델과 문서 모델 사이의 유사도 측정은 사용자의 요구 내용과 관심사항에 가장 부합되는 문서를 사용자에게 제안하기 위해 사용자의 프로파일을 구성하는 용어와 문서들의 용어를 비교하는 것이다.

본 시스템에서는 사용자 모델과 문서 모델을 구축하기 위해 TF×IDF 방법과 벡터 공간 표현법을 사용하였다. 이에 따라 단어와 가중치의 쌍으로 표현된 문서 및 질의는 다음 <그림 6>와 같이 벡터 공간에서의 크기와 방향을 가진 벡터로 나타낼 수 있다[14].

---

4) 문서 표현에서 제외되는 단어들로써, 예를 들어 "a", "and", "the", "with" 등의 기능어(function word)를 들 수 있다.

5) 모든 형태론적 변이(morphological variants)를 단일 형태의 단어로 바꾸는 일로써, 예를 들어 "works", "working", "worked"를 "work"로 바꾸는 일을 말한다.

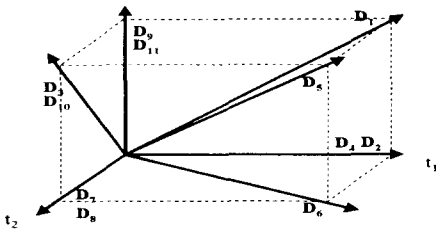
6) Term Frequency, 용어 빈도수

7) Inverse Document Frequency, 역 문서 빈도수

8) tf_{ik} : 문서에서 용어 T_k가 나타난 횟수

9) N : 전체 문서 수

10) n_k : 용어 T_k가 들어있는 문서 수



<그림 6> 벡터 공간에서의 문서와 질의

이러한 벡터 공간에서의 문서와 질의 사이의 유사도를 측정하는 방법 중 TF×IDF 방법의 경우에는 주로 코사인 유사도(Cosine Similarity) 측정법이 많이 사용된다. 코사인 유사도는 두 벡터간의 일치도를 구하는 것으로서, 본 논문에서는 사용자 프로파일의 용어 벡터와 새로운 문서들의 용어 벡터 사이의 유사도를 구하는 것에 해당한다. 이에 따라 가장 큰 코사인 값을 가지는 문서가 가장 사용자의 관심에 유사한 문서라고 할 수 있다.

$$\mathrm{sim}(Q, D_i) = \frac{Q \cdot D_i}{|Q||D_i|} = \frac{\sum_{j=1}^{t} w_{q_j} \times w_{d_{ij}}}{\sqrt{\sum_{j=1}^{t} (w_{q_j})^2 \times \sum_{j=1}^{t} (w_{d_{ij}})^2}}$$

<그림 7> 코사인 유사도 측정법

코사인 유사도 계산식은 <그림 7>과 같은데 [14][16], 두 용어 벡터 사이의 유사도 비교를 위해 두 벡터 사이의 각도의 코사인을 계산하는 것에 해당한다.

이상에서 보듯이, 질의와 문서간의 유사도는 용어의 가중치를 이용하여 계산되며, 유사도 스코어는 양의 실수로 표시되어 관련성 정도를 나타내게 되는데, 0인 경우 관련이 없는 문서에 해당한다.

본 시스템에서는 사용자 모델과 검색된 문서 모델 사이의 유사도를 코사인 유사도 측정법에 의해 계산한 후, 유사도의 역순으로 랭크를 매기고 이를 사용자에게 제공한다.

3.5 사용자 프로파일을 이용한 문서 요약

텍스트 요약 내용을 결정하는데 주로 사용되는 기법에는 구조적 방법(structural method), 개념적 방법(conceptual method), 통계적 방법(statistical method), 통사론적 방법(syntactic method) 등이 있으며, 크게 추출 기반 접근법(extraction-based approach)과 이해 기반 접근법(understanding-based approach)으로 구분된다[3].

본 논문의 시스템은 이러한 기법중 추출 기반 접근법에 의해 문서 요약을 수행한다. 이는 분야에 무관하게 사용할 수 있고, 이해 기반 접근법에 비해 어떤 지식베이스의 구축도 선행될 필요가 없기 때문에 구현이 쉽고 간단하다는 장점 때문이었다[3].

이에 따른 일반적인 요약 알고리즘은 다음과 같은 순서로 진행이 된다[11]. 먼저 선처리(preprocessing) 단계에서 단어를 추출한 뒤, 모든 단어가 스코어를 내는데 적격인 것은 아니므로, 정해진 기준에 만족하는지를 판단하여 적합한 단어를 선택한다. 다음으로 정렬(ranking) 단계에서는 단어에 스코어를 부여하는 단어 랭킹, 문장에 스코어를 부여하는 문장 랭킹을 수행하여, 높은 랭킹의 문장을 문서의 요약으로서 추출한다.

본 시스템에서는 원래의 문서로부터 사용자의 프로파일과 관련하여 중요하고 대표적인 문장을 사용자가 정한 요약 문서 길이의 비율에 따라 제공하는 방법으로 동적인 요약을 수행한다. 이를 위해 각 문장에 대해 문서를 대표하는 정도를 <그림 9>와 같이 계산하고, 이 값이 큰 문장의 순으로 사용자 선택 비율만큼을 사용자에게 요약으로 제공한다.

문단 대신 다중 문장을 요약으로 제공하는 방법을 택한 이유는, 문단을 요약으로 제공하는 경우 불필요한 문장이 삽입되는 단점과 문단을 구분해 내기 어렵다는 단점 때문이었다[19].

정보 여과를 위해 문서 모델링 단계에서 구축된 <용어, 용어의 가중치> 외에 요약을 위한 추가 정보로는 문장 번호, 문장의 위치와 길이 등의 휴리스틱 정보를 사용하였으며, 이를 문서 모델링 과정에서 색인시에 기록하도록 하였다. 이러한 추가 정보는 요약을 위한 문장의 스코어 계산에 사용자 프로파일과 함께 사용된다.

$$D_i = \{S_{i1}, S_{i2}, S_{i3}, \ldots, S_{in}\}$$
$$S_{ij} = \{t_{ij1}, t_{ij2}, t_{ij3}, \ldots, t_{ijn}\}$$

<그림 8> 문서의 구성

<그림 8>은 보통의 문서 구성을 나타낸다. 즉, 문서 D_i는 문장 S_{i1}, S_{i2}, S_{i3} 등의 집합으로 이루어지고, 각 문장은 용어 t_{ij1}, t_{ij2}, t_{ij3} 등의 집합으로 이루어진다.

이에 따라 문장별 문서의 대표도를 나타내는 스코어 계산식은 다음의 <그림 9>와 같다.

$$ref(S_{ij}) = \frac{\sum_{k=1}^{t} w_{q_k} \times w_{s_{ijk}}}{\sqrt{\sum_{k=1}^{t} (w_{q_k})^2 \times \sum_{k=1}^{t} (w_{s_{ijk}})^2}} \times h$$

h : 문장 길이 등 휴리스틱 정보에 따른 모수

<그림 9> 문장별 대표도 스코어 계산식

즉, 각 문장에 대해 사용자 프로파일과의 유사도

측정을 하고, 문서 모델링 과정에서 색인시에 기록해 둔 문장 길이 등의 추가 정보를 사용하여 문장의 대표도 스코어를 계산한다.

계산된 문장별 스코어에 의해 요약으로 제공할 문장을 선택하는 방법에는 순위, 임계값, 비율 등에 의한 방법이 있는데[19], 이 중 비율에 의한 선택으로 사용자가 입력한 비율만큼이 요약으로 제공되도록 하였다. 즉, 문장의 스코어가 높은 순서부터 일정 비율만큼의 문장을 요약으로 제공한다.

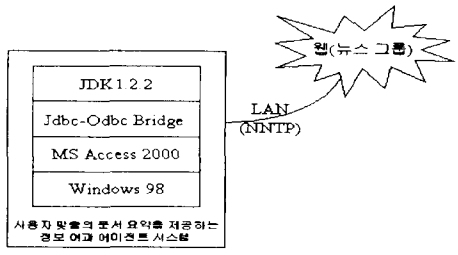
4. 구현 및 실험

4.1 시스템 개발 환경

본 시스템은 윈도우98(Windows98) 운영체제 환경에서 Microsoft Access 2000 데이터베이스를 사용하여 구현하였다. 또한 jdk1.2.2의 자바(Java)를 구현 언어로 사용하였다. 자바는 인텔리전트 에이전트의 자율성(autonomy), 지능(intelligence), 이동성(mobility) 등의 속성을 지원하는데 뛰어나기 때문이다[17]. 데이터베이스와 자바의 연동은 Jdbc-Odbc Bridge를 이용하였으며, 네트워크를 사용할 수 있는 랜(LAN)환경에서 구현하였다.

실험 데이터로는 뉴스 그룹의 기사를 이용하였는데, Net News Transport Protocol(NNTP)을 사용하여 읽어온 뉴스 그룹 기사에 대해 스코어를 부여하여 사용자가 원치 않는 뉴스 기사를 여과해내도록 하였다.

이러한 시스템 환경을 그림으로 나타낸 것이 <그림 10>이다.



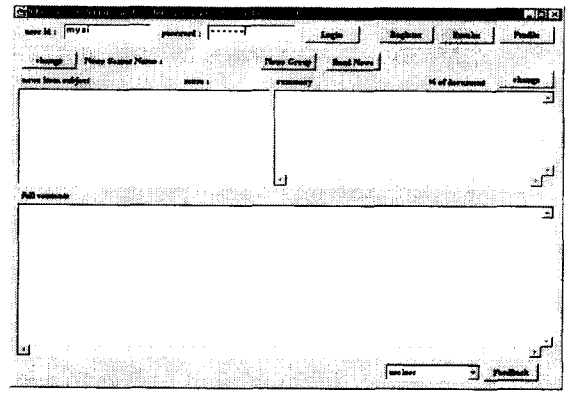
<그림 10> 시스템 개발 환경

이렇게 자바 환경에서 구현된 GUI(Graphical User Interface)를 통하여 사용자는 특정 서버에 접속하고, 관련 문서 및 요약을 제공받게 된다. 또한 관심 단어 입력, 피드백 부여, 문서 요약 비율 선택, 뉴스 서버 및 뉴스 그룹 결정 등의 시스템 상태를 구성할 수 있다.

4.2 구현 화면

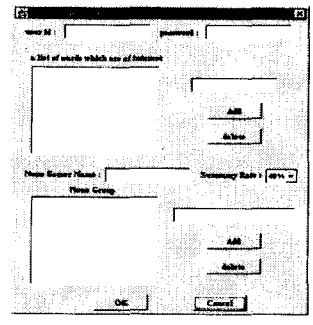
본 시스템은 3.1절에서 제시한 <그림 1>의 시스템의 구조로 이루어져 있으며, 3.1.2절에서 제시한 시스템의 수행 과정에 따라 그 수행이 이루어지게 된다.

본 시스템의 메인 화면은 <그림 11>과 같다. 이 화면을 통해 이미 등록된 사용자라면 user id와 password를 입력함으로써 login이 이루어진다.



<그림 11> 시스템 메인 화면

초기 사용자는 Register 버튼을 통해 자신의 정보를 입력해야 시스템 사용이 가능한데, 사용자 등록 화면은 <그림 12>와 같다.



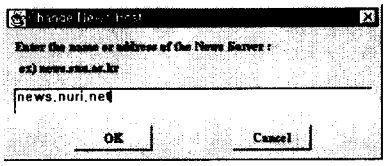
<그림 12> 사용자 등록 화면

이러한 사용자 등록 화면을 이용하여 사용자는 id, password, 관심 단어 리스트, 뉴스 서버, 뉴스 그룹, 문서 요약 비율을 자신에 맞게 입력한다.

이를 통해 여러 분야에 걸친 관심 정보 여과를 수행할 수도 있는데, 즉, 컴퓨터중 AI, 컴퓨터 프로그램밍 언어중 JAVA, 취미중 여행 정보 등과 같이 자신의 관심 분야별 정보만을 다루는 고유의 id와 password를 구성하고 이를 이용하여 정보 여과 에이전트를 효과적으로 이용할 수 있다. 이것은 특히 자신의 관심 분야의 domain specific reference를 구축하는데 도움을 주게 되며, 매번 문서를 새로 읽어올 때마다 그 내용이 업데이트된다.

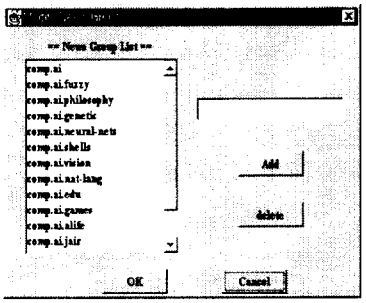
Revoke 버튼을 통해 사용자는 id, password를 포함한 등록 정보를 삭제할 수 있고, Profile 버튼은 사용자 기호 프로파일의 내용을 수정할 수 있도록 한다.

등록된 뉴스 서버 이름은 다음 <그림 13>과 같이 뉴스 서버 변경 화면을 통해 변경할 수 있다.



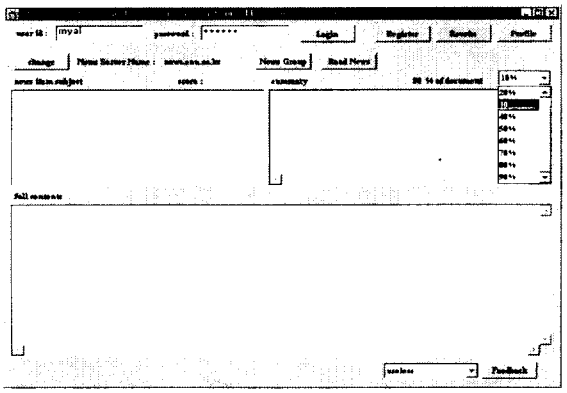
<그림 13> 뉴스 서버 변경 화면

<그림 14>의 뉴스 그룹 변경 화면은 등록된 뉴스 그룹을 확인해 보거나, 변경이 가능하도록 한다.



<그림 14> 뉴스 그룹 변경 화면

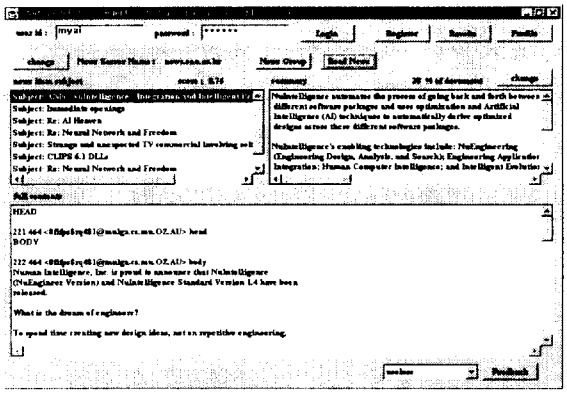
문서 요약 비율 또한 다음 <그림 15>와 같이 선택을 통해 변경할 수 있다.



<그림 15> 문서 요약 비율 변경 화면

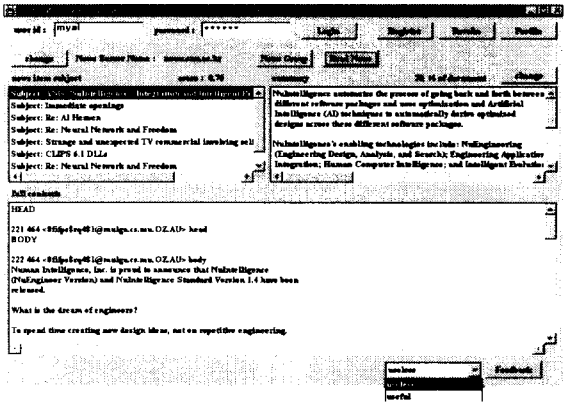
이렇듯 자신에 맞는 시스템 상태를 구성한 후, Read News 버튼을 누르게 되면 사용자가 등록한 뉴스 서버의 뉴스 그룹으로부터 새로운 뉴스를 읽어오게 된다. 이렇게 가져온 뉴스에 대해 색인을 수행하여 <용어, 용어의 가중치> 등의 정보를 생성해 내게 되는데, 색인시 숫자와 특수 control 문자는 제외하여 처리하였고, 대문자 역시 고려하지 않았다. 이러한 색인 및 유사도 측정, 요약 등은 읽어오는 순간에 동적으로 처리되도록 하였다.

이러한 과정을 통해 구축된 문서 모델은 사용자 프로파일과의 유사도 측정을 통해 유사도가 높은 순으로 사용자에게 그 스코어와 함께 제공된다. 이때 각 문장별 유사도에 의해 사용자가 선택한 비율만큼이 <그림 16>과 같이 사용자 요약으로 제공된다.



<그림 16> 여과된 뉴스 기사 및 요약 화면

사용자는 이렇게 자신에게 제공된 문서의 관련성 여부를 <그림 17>과 같이 피드백으로 입력하게 되고, 이의 학습을 통해 사용자 프로파일이 사용자의 기호 변화를 반영하도록 업데이트된다.



<그림 17> 사용자 관련성 피드백 입력 화면

5. 결론 및 향후 과제

본 논문에서는 검색 엔진에서의 낮은 정확율의 단점을 보완하여 사용자를 정보 과잉 상황에서의 불필요한 정보로부터 보호하기 위해, 사용자의 프로파일을 기반으로 하여 정보를 개인화된 요약과 함께 제공하는 정보 여과 에이전트(information filtering agent)인 '사용자 맞춤의 문서 요약을 제공하는 정보 여과 에이전트 시스템'을 제안하였다.

제안한 시스템에서는 학습된 사용자의 개인 정보에 따라 여과된 문서를 개인화된 자동 요약과 함께 제공함으로써, 사용자에게 정보 여과로 제공된 문서를 다시 쉽게 여과할 수 있도록 하는 최상의 정보 여과 효과를 제공하는 장점이 있다.

뿐만 아니라, 자동요약에서의 몇 가지 문제점을 정보 여과 에이전트와 함께 구현함으로써 보완할 수 있다. 우선, 모든 사람의 요구에 맞는 요약문을 작성할 수는 없다는 문제점을 사용자 프로파일에

기반한 요약문을 제공함으로써 보완할 수 있다. 다음으로 요약문 생성에서의 최적 길이에 대한 문제를 들 수 있는데, 정보 여과 에이전트를 통한 사용자와의 상호작용에 의해 요약문으로 제공할 문서의 길이를 비율로 입력받음으로써 이 문제 역시 보완할 수 있다.

이를 위해 본 시스템에서는 사용자와의 상호작용을 통해 사용자의 기호를 학습하게 되는데, 사용 횟수가 증가할수록 사용자의 기호와 매치되는 좀 더 개인화된 문서 및 요약을 제공할 수 있게 된다.

위와 관련하여 본 시스템의 단점은 사용자가 자신의 기호에 맞는 만족할만한 정보를 제공받기 위해서는 어느 정도의 시간이 필요하다는 점이다. 우선 용어의 가중치 계산을 위해 사용하는 TF×IDF 방법의 경우, IDF를 통해 만족할만한 intra-cluster 유사도를 측정(문서 전체 집합 안에서 용어의 출현 특성을 측정)하기 위해서는 전체 문서 집합이 어느 정도 확보되어야 하는데 이를 위한 시간이 필요하며, 또한 사용자 개인의 기호가 충분히 반영된 사용자 프로파일을 구축하기 위해서도 학습을 위한 시간이 필요하기 때문이다. 따라서 이의 해결 방안에 대한 연구가 필요하다.

참고문헌

[1] Marko Balabanovic, Yoav Shoham, Yeogirl Yun, "An Adaptive Agent for Automated Web Browsing", Stanford University Digital Library Project Working Paper, 1995.

[2] http://www.inxight.com/

[3] James Liu, Yan Wu, Lina Zhou, "A Hybrid Method for Abstracting Newspaper Articles", Journal of the Americal society for information science 50(13), pp. 1234-1245, 1999.

[4] Dunja Mladenic, J. Stefan Institute, "Text-Learning and Related Intelligent Agents: A Survey", IEEE Intelligent Systems, pp. 44-54, Jul./Aug. 1999.

[5] H. A. Proper, P. D. Bruza, "What is Information Discovery About?", journal of the American Society for information science, pp. 737-750, Jul. 1999.

[6] R. Chandrasekar, B. Srinivas, "GLEAN: Using Syntactic Information in Document Filtering", Information Processing & Management, Vol. 34, No. 5, pp. 623-640, 1998.

[7] Alper K. Caglayan, Colin G. Harrison, "Agent Sourcebook", New York : John Wiley & Sons, Inc., 1997.

[8] http://www.cs.cmu.edu/~yiming/courses/11741/index.html

[9] Joon Ho Lee, Hyun Yang Cho, Hyouk Ro Park, "n-Gram-based indexing for Korean text retrieval", Information Processing and Management 35, pp. 427-441, 1999.

[10] Gerard Salton, Amit Signhal, Mandar Mitra, Chris Buckley, "Automatic Text Structuring and Summarization", Information Processing & Management, Vol. 33, No. 2, pp. 193-207, 1997.

[11] http://www-4.ibm.com/software/data/iminer/fortext/summarize/summarize.html

[12] Francis Crimmins and Alan F. Smeaton, "TetraFusion: Information Discovery on the Internet", IEEE intelligent systems, pp. 55-62, Jul./Aug. 1999.

[13] Anandeep S. Pannu & Katia Sycara in CMU, "A Learning Personal Agent for Text Filtering and Notification"

[14] http://www.sims.berkeley.edu/~hearst/

[15] "Intelligent Information Agents : Agent-Based Information Discovery and Management on the Internet", Springer, 1999.

[16] Ricardo A. Baeza-Yates, Berthier Ribeiro-Neto, "Modern Information Retrieval", Addison Wesley Longman, Inc., 1999.

[17] Joseph P. Bigus, Jennifer Bigus, "Constructing Intelligent Agents with Java", John Wiley & Sons, Inc., 1998.

[18] 장동현, 맹성현, "자동 요약 시스템", 한국정보과학회지, 제15권, 제10호, pp. 42-49, 1997.

[19] 조태호, "키워드 가중치에 의한 텍스트 요약에서의 다중 문장 선택", 1999년 한국정보처리학회 춘계 학술발표논문집 제6권 제1호, pp. 649-652.