碩士學位論文

텍스트마이닝 기술을 이용한 효율적인 검색 알고리즘 연구

濟州大學校 大學院

컴 퓨 터 工 學 科

金 帝 錫

2005年 12月

텍스트마이닝 기술을 이용한 효율적인 검색 알고리즘 연구

指導敎授　金　壯　亨

金　帝　錫

이 論文을 工學 碩士學位 論文으로 提出함.

2005年　12月

金帝錫의 工學 碩士學位 論文을 認准함.

審査委員長　　印

委　　員　　印

委　　員　　印

濟州大學校 大學院

2005年 12月

A Study of an Efficient Retrieval Algorithm Using a Text Mining

Je-Seok Kim
(Supervised by professor Jang-Hyung Kim)

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING

DEPARTMENT OF COMPUTER ENGINEERING
GRADUATE SCHOOL
CHEJU NATIONAL UNIVERSITY

2005. 12.

목 차

[그림 목차]

[표 목차]

Summary

Nowadays search engines have to sort through a prodigious amount of information and have become systems that find useful information among the unwanted information base. Therefore, in order to effectively find useful information in the information search environment we need to emphasize technology that will allow us to extract useful information from non-standard data.

In order to solve these problems our research has, through text mining technology, used learning methods from text documents in order to come up with ways of seeking new information. By doing so we have classified various original text mining documents into one text and compared them with a text, which had classified all the related documents. Also, we have used these classified documents so that when a user requests a keyword of a wanted document, the systems notices the keyboard and makes decisions concerning the related groups based on the classified groups. Therefore, after the documents are searched only in the designated groups, the words seldom used are discarded and the search results are offered to the user. In various anonymous documents, this method is used in order to find the exact document the user wants. Furthermore, when searching for information, by eliminating the unnecessary words from the start, using original text mining technology, and extracting the most suitable search words; we are able to rank the useful information, after measuring its value in order to come up with the results tailored to the user's requests, at the top of the page. By doing so we hope to come up with the most ideal algorism architecture research that can search for the most useful information possible, which would be our final goal for this study.

Ⅰ. 서 론

정보화 사회에서는 인터넷과 이에 기반 한 E-business는 기존 산업의 전부분에 걸쳐 효율성과 생산성 증대를 위한 전략적인 도구로 그 중요성이 지속적으로 증대되고 있으며, 효율적 정보검색(information retrieval)은 각종 의사결정에 매우 중요하며 그 결과에 따라 개인이나 기업, 그리고 국가의 성패가 달라질 수 있다. 새로운 기업정보 자료들이 끊임없이 등록되고, 지난 자료들이 갱신, 수정되는 등 전 세계에 있는 수많은 기업에서 다양한 지식자산(Knowledge Asset)들이 지속적으로 생성, 저장, 재사용하는 정보 중 20%만이 활용성이 높은 정형 데이터로 구성되어 있고, 나머지 80%는 워드프로세서, e-mail, 프리젠테이션, 스프레드시트, PDF와 같은 복합문서와 인터넷 페이지등의 비정형 텍스트 형태로 구성되어 있다.

또한 일반적인 웹에서의 검색과는 달리 대부분의 기업에서는 회사 내에 데이터 서버를 두고 무수히 많은 텍스트 문서를 서로 공유하고 있다.

서버에서 접속하여 문서를 참조하는 사용자가 원하는 문서를 찾기 위해서는 폴더를 이용하여 문서의 종류와 날짜, 내용 등을 간단히 표기 할 수 있으나, 이 방법에는 한계가 있다. 문서의 수가 많아질수록 셀 수 없이 많은 폴더와 파일들이 쌓여가고, 사용자가 원하는 문서를 검색하기 위해서는 무수히 많은 폴더와 파일들을 열어보고 닫고 해야만 할 것이다.

이 방법은 효과적인 업무 처리에 있어서 상당히 비 효율적일 뿐 만 아니라, 문서를 찾기 위해 여러 사람이 서버에 접속하여 여러 문서 파일들을 열어보고 닫고 할 경우, 네트워크의 트래픽 또한 야기 시키게 될 것이다. 그래서 최근 기업에서 잠재적인 정보를 발견해 내기 위해 많이 사용하는 데이타마이닝 기술중 비구조적인 텍스트 문서로부터 정보를 찾아 지식을 발견하는 것이 텍스트마이닝이다. 그러나 텍스트마이닝은 정형화된 데이터를 위한 일반 데이터

마이닝에 비하여 정보추출 능력이나 정확성 등이 많이 떨어지는 경향이 있다

최근 기업에서 유용하고 잠재적인 정보를 발견해내기 위해 많이 사용하는 데이터마이닝 기술은 정형화된 형태의 데이터를 주대상으로 하고 있다. 그러나 대규모 텍스트 데이터들은 구조적인 형태로 분석하기가 쉽지 않고, 대부분 자연어로 쓰여진 문장 형태이기 때문에 함축된 정보를 추출하기가 쉽지 않다. 이러한 비구조적인 텍스트문서로부터 정보를 찾아 패턴을 학습데이터로 수행하여 새로운 지식을 발견하는것이 텍스트마이닝이다. 또한 기존의 텍스트마이닝은 자연어로 구성된 비구조적인 텍스트 안에서 패턴 또는 관계를 추출하여 지식을 발견하는 것으로 주로 텍스트의 자동 분류작업이나 새로운 지식을 생성하는 작업에 활용되고 있다. 기존의 텍스마이닝은 여러문서들을 하나씩 미리 분류해 놓은 문서와 비교하면서 관련문서를 분류해 놓은 문서와 비교하면서 관련문서를 분류하고 이렇게 그룹이 지어진 문서들을 가지고 사용자가 원하는 문서의 키워드를 요청하게 되면 시스템이 키워드를 보고 분야별로 그룹을 만들어둔 곳 중 연관그룹을 판단하여, 해당그룹에서만 문서를 검색한 후 문서에서 자주 나오는 불용어를 제거 후 검색결과를 사용자에게 제공하게 된다.

본 연구에서는 이러한 문제점을 해결하기 위하여 검색시 불필요한 불용어를 최우선적으로 제거하여 최적의 검색어를 추출하여 사용자 요구사항에 맞는 결과를 얻을 수 있게 가중치를 부여 유용한 정보를 상단에 랭킹시키므로써 가장 유용한 정보를 검색할 수 있는 최적의 알고리즘 아키텍처 연구하는데 최종의 목표를 두고 있다.

Ⅱ. 텍스트마이닝

1. 텍스트마이닝 개요

데이터 마이닝 이전 연구들은 대부분, 트랜잰션과 데이터 웨어 하우스 데이터 같은 구조적 데이터에 초점을 두어왔다. 그러나, 현실적으로, 유용한 정보의 실질적인 부분들은 텍스트 데이터베이스(text database, 혹은 document database)에 저장되어있다. 이 데이터베이스들은 뉴스 주제, 연구 논문, 책 디지털 도서관, 이메일 메시지, 웹페이지 같은 다양한 소스들로부터 수집된 방대한 양의 문서들이다. 텍스트 데이터베이스는 전자 출판물, 이메일, CD-ROM, WWW(이것 또한 방대하게 연결된 동적인 텍스트 데이터베이스이다)과 같은 전자 형태로 얻을수 있는 정보들의 양이 증가함에 따라 빠르게 성장하고 있는 추세이다.

전통적인 정보 검색 기술들은 많은 양의 텍스트 데이터 증가에 적당하지 않다. 일반적으로, 많은 가용 문서들의 단지 작은 부분만이 개인이나 사용자에게 적절한 내용이다. 무엇이 문서들에 있을 수 있는지 모르고서는, 데이터로부터 유용한 정보를 분석하고 추출하기 위한 효과적인 질의를 정형화하는 것은 어렵다. 사용자들은 다른 문서들을 비교하고, 문서들의 중요도와 관련도에 따라 등급을 매기고, 혹은 다중 문서들의 경향과 패턴을 발견하는 도구들이 필요하다. 따라서, 텍스트 마이닝은 데이터 마이닝에서 점차 인기있고, 필수 적인 주제가 되고 있다[1] 오늘날 대부분 텍스트 데이터베이스에 저장된 데이터는 완전하게 구조적이지도 않고 또한 완전하게 비구조적인 형태인 반구조적인 데이터(semistructured data)이다. 이런 비정형 데이터에서 이루어진 문서들 중에서 학습을 통해서 새로운 정보를 찾아내는 기술로써 불특정 다수의 문서 내에서 사용자가 의도한 문서를 정확하게 찾아내는 방법으로써 오늘 텍스트 마이닝 프로그램은 학계와 기업에 주로 많이 사용되고 있다.

2. 텍스트 데이터 분석과 정보검색

정보검색과 데이터베이스시스템은 각각 다른 종류의 데이터를 다루기 때문에 동시성 제어, 회복, 트랜잭션 처리, 수정과 같이 정보검색 시스템에서 보통 존재하지 않은 데이터베이스 시스템 문제들이 있다. 또한 구조적이지 않은 문서, 키워드에 기반한 유사 검색, 관련성 개념들과 같은 전통적 데이터베이스 시스템들에서는 다루어지지 않는 일반적인 정보 검색 문제들이 있다.

1) 텍스트 검색의 기본척도

텍스트 검색의 질을 평가 하기 위한 척도는 정밀도(Precision)와 응답도(Recall)로 두 가지 척도들이 있다.

가) 정밀도(Precision): 이것은 질의와 검색문서들간의 사실적인 (“올바르다”라 응답) 관련된 것을 퍼센티지로 나타낸다.

$$\text{정밀도(Precision)} = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Retrieved\}|}$$

나)응답도(Recall): 이것은 질의와 관련된 실제로 검색된 문서들의 퍼센티지이다.

$$\text{응답도(Recall)} = \frac{|\{Relevant\} \cap \{Retrieved\}|}{|\{Relevant\}|}$$

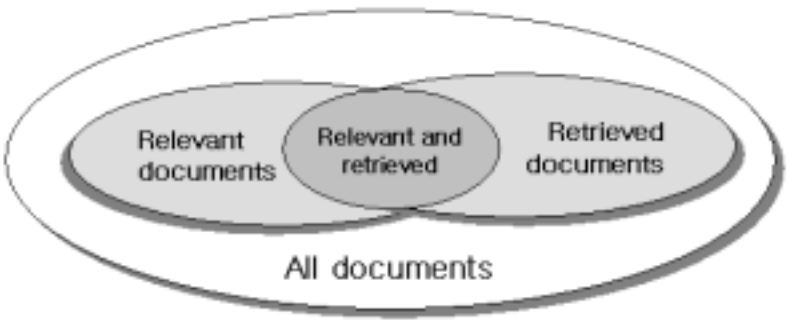

Fig. 1 Precision and Retrieved Relations

2) 키워드 기반과 유사도 기반 검색

대부분의 정보 검색 시스템들은 키워드 기반이나 유사도 기반 검색을 제공한다. 키워드 기반 정보검색(keyword-based information retrieval)에서, 문서들은 키워드나 키워드의 집합으로 인식되는 문자열로 표현되어진다. 사용자는 “car and repair shops", 혹은 "tea or coffee"과 같은 키워드나 키워드의 집합으로 구성되는 표현식을 제출한다. 좋은 검색시스템은 질의어들의 검색시 동의어를 고려해서 검색한다 예를 들어 car가 주어졌을때, automobile, vehicle과 같은 동의어도 검색시 고려되어야한다. 키워드 검색시 두가지 어려움이 있는 데 첫번째가 동의어문제를 문서 어디에서나 적용 똑같이 적용 되지 않는다. 두번째 다중의미문제(polysemy problem)로 예를 들어 system과 같은 키워드도 다른 문맥에서 다른 의미로 사용 될 수 있다.

유사도 기반검색(similarity-based retrieval)은 공통 키워드들의 집합에 기반해 유사한 문서들을 발견해낸다. 그 결과는 관련성 키워드의 근접성, 키워드 빈도수 등에 기반한 관련도(degree of relevance)에 기반 하여야 한다. 많은 경우 키워드들의 집합에서의 관계도, 예를 들어 데이터 마이닝과 데이터 분석 사이의 거리와 같은 척도를 정확히 제공하는 것은 어렵다.

텍스트 검색 시스템은 종종 관련없음(irrelevant)으로 간주되는 단어들의 집합이다. 이러한 것들을 “불용어”라 불리우며 예를 들어 a, the, of, for, with등 자주 나타날수 있더라도 검색과 전혀 관련 없는 불용어라 불린다.

여러 다른 단어들의 그룹이 같은 어근을 공유할 수 있다. 텍스트 검색시스템은 그룹에서 단어들이 다른 단어와 작은 구문 변화만 있는 단어들의 그룹들과 동일시하는 것이 필요하고 그룹 당 공통된 어근을 수집할 필요가 있다. 예를 들어 drug, drugged, drugs들의 그룹은 drug이라는 공통된 어근을 공유하여, 같은 단어의 다른 산출물들로 볼 수 있다.

비슷한 문서들이 유사한 관련 용어 빈도수들을 갖는다고 기대하기 때문에, 우리는 빈도수 표에서 유사한 관련 용어 발생에 기초하여 질의에 대한 문서들 사이, 또는 문서들의 집합에서의 유사성을 측정할수 있다.

문서 유사성을 측정하기 위해 많은 척도가 제안되어 왔다 대표적인 코사인 척도(cosine measure)로 두 문서 벡터들을 v_1, v_2 라고 했을때 코사인 유사성은 다음과

같이 정의한다.

$$\mathrm{sim}(v_1 \cdot v_2) \frac{v_1 \cdot v}{|v_1||v_2|}$$

3) 잠재의미 인덱싱(Latent Semantic indexing)

잠재의미 인덱싱 방법은 용어 빈도수 행렬의 크키를 줄이기 위해 행렬 이론에서 잘 알려진 기술인 특이값 분해(singular value decomposion, SVD)를 사용한다. T용어들과 D문서들로 표현되어진 T*D용어 빈도수 행렬이 주어지면,SVD방식은 큰 수집 문서들에 대해 약 몇 백(예, 200)의 값을 갖는 K에 대하여, K*K사이즈로 행렬을 줄인다. 정보손실의 양을 줄이기 위해, 단지 빈도수 행렬의 덜 중요한 부분들이 빠지게 된다.

4) 기타 텍스트 검색 인덱싱 기법

역 인덱싱(inverted index)와 시그니쳐 파일(signature files)을 포함한 몇개의 알려진 텍스트 검색 인덱싱 기법들이 있다.

역 인덱싱(inverted index)는 문서표(document table)와 용어표(term_table)라는 두개의 해시 인덱스 테이블 혹은 B+tree 인덱스 표를 유지하는 인덱스 구조이다.

시그니쳐 파일(signature files)은 데이터베이스에서 각 눈서에 대한 기호 기록을 저장하는 파일이다. 각 기호는 용어를 나타내는 B비트의 고정 사이즈를 가진다. 단순한 부호화 기법은 다음과 같다. 문서 기호의 각 비트는 0으로 초기화 되어진다 그것을 표현하는 용어를 문서에 나타낸다면, 비트는 1로 도니다. 기호 S_2에 지정된 각 비트가 또한 S_1에 지정된다면 기호 S_1은 다른 기호 S_2로 매치된다.

3. 키워드 기반 연관과 문서분류

분석은 함께 빈번하게 발생하는 키워드 집합이나 용어들에서 수집하고 그때 그들 사이의 연관성이나 상호관계를 발견한다.

텍스트 데이터베이스에서 대부분 분석들처럼, 연관 분석은 먼저 텍스트를 구문분석,어근분석,불용어 제거 등의 순서로 전처리를 하고, 연관 마이닝 알고리즘을 가동한다. 문서 데이터베이스에서, 각 문서는 트랙잭션으로 볼수 있고, 문서에서 키워드들의 집합은 트랙젝션에서 항목의 집합으로 생각할 수 있다. 즉 데이터베이스는 다음 형식이다.

{ document_id,a_set_of_keyword }

빈번하게 발생하는 연속 키워드들이나 근접하게 위치한 키워들은 용어(term)나 구(phtase)를 형성할 수 있다는 것을 주목해야 할 것이다. 연관 마이닝 처리는 영역 독립 용어 혹은 구에서 [Standford, University] 또는 [U.S., president, Bill, Cliton]와 같은 복합연관(Compound associations)의 감지나, [dollars, shares, exchange, total, commission, stake, securities]와 같은 비복합연관(noncompound associations)의 감지를 쉽게 한다. 이러한 연관에 기초한 마이닝은 “용어 수준 연관 마이닝”(개개의 단어들을 마이닝하는 것에 비하여)으로 불러진다.

1) 문서 분류 분석

자동 문서 분류는 대량의 거대한 온라인 문서들의 존재와 함께 중요한 텍스트 마이닝 작업으로, 어려운 작업이지만 문서검색과 부순차적 분석을 실행하기 위해 클래스로부터 그런 문서들을 자동적으로 구성할수 있게하는 작업은 필수적이다.

“어떻게 자동 문서 분류를 실행할 수 있는가?” 일반적인 절차는 다음과 같다.

첫째, 미리 분류된 문서들의 집합을 시험 집합으로 택한다. 그리고 시험집합은 분류체계를 유도하기 위해 분석되어진다. 그런 분류체계는 종종 시험 과정에서 세분화될 필요가 있다. 그렇게 생성된 분류체계는 다른 온라인 서류들의 분류에 사용되어 질 수 있다.

이 과정은 관계 데이터의 분류와 비슷하게 보인다. 그러나 기본적인 차이는 있다. 관

계 데이터는 잘 구조화되어 있다. 각 투플(tuple)은 속성-값 쌍의 집합으로 정의된다. 예를 들어 {sunny, warm, dry, not_windy, play_tennis} 투플(tuple)에서 "sunny"라는 값은 weather_outlook속성과 상응하고, "warm"은 temperature속성과 상응한다. 분류 분석은 어떤 속성-값 쌍들의 집합이 사람들이 테니스들 치러가는지 여부를 결정하는 데 있어서 가장 큰 차별 요소를 결정한다.

문서분류를 위한 효율적인 방식은 빈번하게 발생하는 텍스트 패턴들에 관련 되어진 집합들에 기초에 문서들을 분류하는 연관 기반 분류를 탐구하는 것이다. 그런 연관 기반 분류 방법은 다음과 같이 진행한다. 첫째, 키워드와 용어들은 정보검색과 단순한 연관 분석 기법에 의해서 추출될 수 있다. 둘째, 키워드들과 용어들어 개념 계층들은 WordNet이나 전문가 지식에 의존하거나, 키워드 분류시스템과 같은 유용한 용어 클래스들로부터 문서들의 한 클래스를 최대한 구별할 수 있는 연관 용어들의 집합들을 찾기 위해 적용할 수 있다. 이것은 각 문서 클래스들에 연관된 연관 규칙들의 집합을 생성한다. 그런 분류 규칙들은 그것들의 발생빈도수와 판별력에 기초해 정렬될 수 있고, 새로운 문서들을 분류하기 위해 사용될 수 있다.

4. 텍스트마이닝 응용 기술 활용

4.1 텍스트 마이닝 (Web Text Mining)

웹의 엄청난 테스트와 문헌들은 이제 적절한 관리와 조직화가 되어야 한다. 웹의 내용은 대부분 비 구조적인 형태를 갖고 있으며 웹 페이지나 문헌들로 되어 있다. 비즈니스 데이터 마이닝을 위한 응용프로그램이 어느 정도 성숙기에 있는 반면에 웹 데이터를 마이닝하고 관리하는 단계는 시작이라고 볼 수 있다. 최근 통계에 의하면 웹 페이지의 90% 이상이 전자적으로 비 구조적인 형태를 취하고 있으며 이러한 추세는 더욱 증가할 것으로 예측한다. 또한 인터넷의 내용은 점점 더 증가가 할 것이다.

인터넷과 인트라넷을 위한 검색엔진이 어느 정도 내용의 접근을 제공하고 있으나 정보 검색의 망라성과 관련이 없는 검색으로 소요되는 시간적 문제들은 해결하지 못하고 있다. 물론 구조적인 데이터에 데이터 마이닝 기술이 잘 활용되고 있지만 비 구조적인 데이터 분석을 위해서 사용되는 텍스트 혹은 지식 마이닝 도구는 이제 막 소개되고 있다.

텍스트 마이닝 기술을 웹 내용에 적용함으로서 웹의 정보에 접근하기 위한 표준화된 측정도구를 개발할 수 있을 것이다. 웹 데이터를 데이터 마이닝 모델인 분류와 클러스터링을 사용함으로서 색인과정을 자동화할 수 있다. 색인 과정을 자동화함으로서 소유권을 문제를 극복할 수 있으며 웹의 크기를 조정할 수 있다. 또한 지능적인 마이닝 도구들은 인터넷과 인트라넷의 풍부한 비 구조적인 데이터를 잘 다룰 수 있을 것이다.

4.2 지식관리시스템 (Knowledge Management System)

일반적으로 데이타(data)는 단순한 사실(fact)을 의미하며, 정보(information)는 이러한 데이타들의 결합을 통해 어떠한 의미가 부여된것 이라고 이야기합니다. 그리고 지식(knowledge)이란 이러한 정보들 중에서 일반화, 즉 모든 사람들이 공유하고 사용할 수 있는 것이라고 말할 수 있습니다. 따라서 지식관리시스템을 단순히 말한다면, 기업내의 조직 구성원들의 다양한 개인적경험(주로 업무와 관련) 중에서 다른 이들도 사용할 수 있는, 즉 일반화될 수 있는 경험들을 다른 이들이 활용할 수 있는 형태로 변화하여 공유할 수 있도록 지원하는 시스템이라고 말할 수 있습니다

1) Sovereign Hill

Sovereign Hill Software는 문서 저장소를 위해 지식 개발과 관계 마이닝 기술을

특수화하고 있다. Sovereign Hill은 메사츄세츠 대학의 CIIR (the Center for Intelligent Information Retrieval)에서 개발한 InQuery 검색엔진의 개념 추출과 분석 도구로 주목받고 있다. 이러한 도구들은 Dataware II와 같이 텍스트 데이터베이스에 있는 기존의 구성을 이용하지만 이들 또한 추가적인 주요 개념들을 자동적으로 확인함으로 색인 작업을 추가한다. 데이터 안의 널리 보급된 개념들은 기본 개념의 관계 속에서 인접성과 출현 빈도수를 분석하여 이를 토대로 자료를 개념화하며, 개념 추출 도구로 개념 추출 도수에 의해서 자동적으로 추출된 관련 개념을 순위 리스트와 함께 검색결과을 제시한다. 이렇게 함으로서 이용자는 관련 개념을 사용하여 원 검색결과를 통해서 드릴 다운할 수 있다. 이 방식은 이용자로 하여금 방대한 데이터베이스에서 간단하고 광범위한 검색을 가능하게 하며 많은 검색 결과에 대하여 순위에 따라 정렬하고 개념별로 정리된 검색결과를 제공한다.

최근에 Dataware는 Sovereign Hill Software, Inc를 매입함으로서 자사의 Dataware II와 Sovereign Hill사의 InQuery 개념 추출 방식과 관계형 마이닝 기술을 통합하여 차세대 지식 마이닝 상품을 생산하고자 한다.

2) Relevance Technologies,

Relavance Technology Inc.은 지능적인 마이닝 솔루션을 전문적으로 개발하는 회사로 1996년 설립되었다. 1998년 Documentum Inc. 에 의해 인수합병되어, 문서관리 솔루션 및 데이터마이닝 슈트(suite)개발에 역점을 두고 있다. 이에 1999년 중반 웹 어플리케이션환경의 데이터 마이닝 기술을 도입한 문서관리솔루션을 개발 보급하고 있다. 이것은 문서 저장소를 위한 정보 마이능 기능분만 아니라 파일 서버, 인트라넷, 전자메일 시스템, 웹과 같은 내외부의 정보화 함께 인터페이스를 제공한다.

Relevance 생산물들은 EIC (the Enterprise Intelligence Center)로 구성되어 있다. 이것은 구조적/비구조적 정보를 마이닝할 수 있으며 필요하다면 특수한 경영 프로세스를 위하여 배포하기도 한다. 즉 이용자들이 적합하고 이전에 알지 못했던 정보을 쉽게 찾을 수 있도록 지능적인 도움을 제공한다.

Relevance EIC는 다른 텍스트 마이닝 상품과 달리 비구조적인 정보의 검색엔진으로의 역할만 하는 것이 아니라 특수한 경영 요구에 맞는 주문된 응용프로그램이다. 다른 마이닝 상품들이 넓은 영역의 개념을 제시하지만 EIC는 특별한 경영 기능의 개념에 초점을 맞추기 때문에 특수한 경영 요구에 중점을 둔 질문에 해답을 줄 수 있다.

이러한 정보 마이닝을 가능하게 하는 기술이 Relevance SMA (the Semantic Modeling Architectur)이다. 이것은 정보를 검색하기 위하여 자연어 처리 기술과 통

계적인 클러스터링, 의미론적 모델을 혼합하여 구성되고 있다. 이 모델은 6가지 요소로 이루어져 있다: 정보원, 정보 관리자, 경영 프로화일, 가상 웨어하우스, 응용 서비스, 이용자 인터페이스. 6가지 요소중에서 Relevance EIC의 정보 마이닝의 열쇄는 " 경영 프로화일"에 달려 있다. 이 도구는 시소러스와 유사하며 경영 관련 개념이 특수한 통제어와 함께 미리 정해진다. 일단 정보원에 있는 개념이 확인되면 적합한 정보를 추출하기 위하여 경영 프로화일은 경영 관련 개념을 모델에 적용한다.

4.3 검색엔진 (Search Engine)

웹 검색엔진에 텍스트 마이닝의 응용이 대두되고 있으며 이것은 좀 더 정확하고 일관성 있는 검색으로 특별하거나 망라적인 결과를 제공할 것이다. 텍스트 마이닝 기법을 활용하고 있는 검색엔진으로 노던 라이트 (Northern Light), 데이터웨어 (Dataware), 익사이트 (Excite(Magellan)), 인화인드(InFind), 구글(Google) 등을 들 수 있다. 여기서는 Northern Light, InFind, 그리고 Google을 소개한다.

1) Northern Light

Northern Light는 1995년 개발된 검색엔진으로서 1400억 이상의 웹페이지와 5,400여 개의 원문데이터베이스를 제공하고 있다. 5,400여 개의 전문 저널, 도서, 잡지, 뉴스와이어 및 참고정보원을 제공하는 특별한 집서를 구성하고 있다. 다른 검색엔진에서는 잘 제공하지 못하는 자료들을 제공한다. 이 엔진은 비교적 학술적 자료 검색에 유용하다. 보통 웹 상에서 무료로 정보를 검색할 수 있으나, 원문을 원할 때는 $1-$4 정도의 이용요금을 지불하게 된다

다른 검색엔진과는 달리 Northern Light는 웹과 특별한 집서의 정보원를 조직화하기 위하여 데이터 마이닝을 사용하여 질이 높은 정보를 제공한다. 다시 설명하면, Northern Light는 전문 사서에 의해 만들어진 폴더 (folder)에 각 정보를 배열하는 분류 데이터 마이닝 모델을 사용한 검색엔진이라 하겠다. 네가지 유형의 폴더는 주제, 문서형식, 소스, 언어별로 분류되며. 대략 20,000이상의 광범위한 계층적 관계어와 200,000-300,000에 이르는 첨가어로 수록된다. 이러한 색인은 사람에 의해 수작업으로 생성되지만, 데이터베이스에 있는 논문들은 컴퓨터에 의해 자동적으로 색인된다.

2) InFind

데이터 마이닝 기법을 검색엔진에 활용한 또 다른 예로 InFind를 들 수 있다. 인터넷 상에서 가장 잘 알려진 여섯 개 엔진 (Altavista, Excite, Infoseek, yahoo!, Webcrawler, Lycos)들을 동시에 검색하는 메타검색엔진인 InFind의 가장 큰 특징 중에 하나는 클러스터링이라고 볼 수 있다. InFind는 각각의 검색엔진에서 검색한 결과물들을 제목의 간략한 리스트를 제시하며 URL없이 유사한 아이템을 모아서 출력한

다. 다른 검색엔진들은 질의어와의 유사도에 따라 검색 결과물들을 나열하고 있으나, InFind는 모든 적합한 결과를 모아서 관련 있는 아이템끼리 모아 그룹화 한다. 이러한 클러스터링 작업은 많은 결과물을 이해하기 쉽게 해주고 이용자로 하여금 자신이 찾고자 하는 정보에 어느 것이 관련이 있고, 어느 것이 관련성이 적은가를 빨리 파악하게 해 준다. 예를 들면 'The David Letterman show'에 대해서 찾고자 할 경우 InFind는 CBS' official site에서 검색한 페이지와 팬으로부터의 페이지를 따로 분류하여 빨리 볼 수 있게 해 준다.

3) Google

Google은 googol에서 따온 이름으로 10의 100제곱 혹은 천문학적인 숫자를 의미한다. 즉 이 검색엔진의 궁극적 목표는 전 세계의 모든 망라적인 정보를 조직화하여 접근하고 유용하게 사용할 수 있도록 하는데 있다고 한다. Google은 1998년 Stanford 대학 박사과정에 있던 Larry Page와 Sergey Brin가 데이터 마이닝과 웹의 링크 구조에 기초하여 만들었다.

Google은 1999년 9월 21일 베타 단계를 마치고 라이브 버전으로 시작한 새내기 검색엔진으로 아직은 비교적 간단한 기능을 제공하고 있으나 반응은 좋은 편이다. Google의 주요 특징은 관련성피드백(relevance feedback)검색, PageRank라는 사이트 순위 선정 방법, 그리고 다국어 검색 서비스를 들 수 있다(이수연, 김성희 2000). Google은 정확도 측정을 문서에서 발생하는 키워드 정보에 의존하지 않고 해당문서를 링크하고 있는 다른 문서 수에 의해 결정한다. 또한 "PageRank" 기술은 수많은 웹페이지가 서로 하이퍼링크로 이어져 있다는 웹의 기본 구조와 인기 있는 사이트는 링크가 많이 되고 있다는 가정에 기초를 두고 사이트 순위를 매기는 것이다. Google의 다국어 검색 서비스는 이용자가 원하는 언어의 페이지를 선택해서 검색할 수 있도록 하여 지역화(localization)를 함께 제공한다.

Google에서만 찾아 볼 수 있는 특징으로 Link to cached page를 검색결과 페이지에 제공한다. 이것은 이미 없어진 사이트나 서버/네트워크의 일시장애나 중단 등으로 접속이 잘 되지 않는 상황을 어느 정도 막아주기도 한다.

5. 기존 텍스트마이닝 기술을 이용한 검색

5-1 기존의 문서 검색 기법

1960년대 이대로, 데이터베이스와 정보기술은 원시적인 파일 처리 시스템(file processing system)으로부터 정교하고 강력한 데이터베이스 시스템으로 체계적으로 진화하여 왔다. 1970년대 이후, 데이터베이스 시스템에 대한 연구와 개발은 초기의 계층형 데이터베이스 시스템(hierarchical database system)과 네트워크형 데이터베이스시스템으로부터 데이터를 관계 테이블 구조에 저장하는 관계 데이터베이스 시스템, 데이터 모델링 도구 그리고 인덱싱 및 데이터 조직기법의 발달로 진보하였다. 사용자는 질의어, 사용자 인터페이스 최적회된 질의처리(query processing) 그리고 트랜잭션 관리를 통하여 편리하고 융통성 있게 데이터에 접근할 수 있게 되었다. 오늘날 데이터의 분산, 다양화 그리고 공유와 관련된 문제들이 광범위하게 연구되었다. 또한 이질(heterogeneous)데이터베이스 시스템과 WWW와 같은 인터넷 기반의 범세계적 정보시스템등이 태동하였고 정보 산업의 중추적 역활을 맡고 있다.

이러한 발전 속에서 데이타베이스에서 가장 유용한 정보를 찾는 텍스트마이닝은 문서의 요약, 문서분류, 문서군집, 특성추출 등 데이타마이닝 관점에서 문서로부터 구조화된 정보를 추출하여 데이타베이스화 시키거나 규칙을 찾아내는 것이 가장 일반적인 Web 상에서 문서를 찾아주거나 사용자에게 profile을 생성 및 분석한다.

오늘날 많이 알려진 구글의 개발자들이 쓴 논문을 보면 기본적으로는 Pattern matching 기법을 사용하였다. 그러나 다른 검색 엔진과 다른 가장 주목해야할 부분은 바로 페이지랭크(Page Rank)를 검색된 문서의 순위 결정에 사용 하였다는 것 이다.

대개 어떤 페이지의 인용 횟수(백 링크:back link)를 세는 방식으로 이뤄진다. 어떤 페이지가 얼마나 많이 인용(참조)되고 있는가를 셈으로써 그 페이지의 중요성이나 품질을 추정해볼 수 있는 것이다. 페이지랭크는 이 아이디어를 더욱 확장해서 단순히 모든 링크를 세는 것에서 한 발 더 나아가 그 링크가 어떤 페이지로부터 왔는지를 차별화했고, 링크를 하고 있는 페이지로부터 외부로 나가는 총 링크 개수로 노멀라이징(normalizing)했다.

그러나 구글 검색엔진은 텍스트마이닝 기법을 적용했지만 기존의 검색엔진의 문제점을 그대로 갖고 있다.

1) 기존의 검색엔진의 문제점

가) 단순 패턴을 통한 과도한 검색 문제

사용자가 검색한 키워드가 동음어이거나 중간 부분이 일치할경우에는 과도한 검색결과를 출력하는 문제이다 예를 들어 “핑클”이라고 검색했을때 “서핑클럽”등 어절을 통한 의미분석 결과를 추가하여 띄어쓰기로 표현 되는 검색 문서의 누락문제가 발생하기도 한다.

나) 검색시 형태소 분석 실패로 인한 검색결과 오류 문제

외국어 및 다양한 언어로 이루어진 키워드 검색시 해당문서를 못찾아 내거나 엉뚱한 정보를 처리할수가 있다. 이러한 문제는 검색시 정보간 철자 및 띄어쓰기 오류로 인한 검색 실패하거나 신조어나 고유명사등이 지속적으로 등록이 되지 못하거나 형태소 분석기의 성능 저하로 인한 검색 결과문제

다) 검색결과의 과다로 인한 정확성 문제

등록된 정봐 많을수로 동의어의 결과 처리로 인한 검색량이 증가로 사용자가 검색결과를 모두 검토하기 어려워 정보검색을 포기하는 경우가 발생하기도 하고 연관성이 없는 내용이 다량으로 포함되므로 검색시간이 많이 소요된다.

라) 그 외 검색속도와 결과표현문제 및 랭킹시스템 문제등

오늘날 복합문서등 다양하게 검색처리능력이 향상이 되었지만 결과표현이 접근성이 떨어지거나 다양한 포맷으로 인한 유기적인 연동이 많이 부족함.

다양성 있는 검색결과를 표현하는 랭킹은 고품질의 문서필터를 시급히 개발하는게 주요 목표이고 고객의 요구에 적절히 대응할수가 있다.

5-2 기존 검색엔진 검색기법

정보 관리 및 검색시스템(EDMS)에서 키워드를 사용자가 입력후 문서 분류후 검색결과를 키워드와 상관도에 의해 랭킹 처리한다.
(예:“세계에서 제일 장수한 사람”) 단 여기서 문장검색(Full Text) 검색시 문구와 문구사이에는 반드시 띄어 쓰기를 하여야 하며 그렇지 못할경우 에는 검색결과의 정확도가 매우 떨어진다.

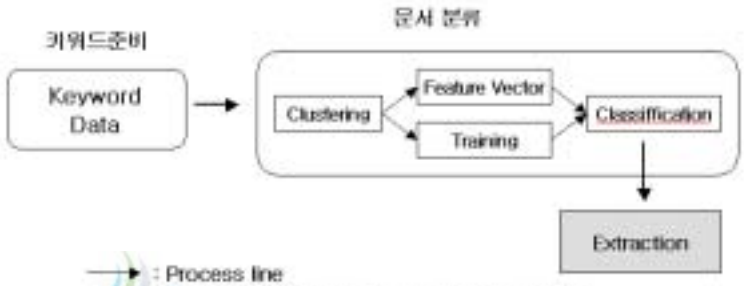

Fig.2 기존 검색엔진 흐름도

핵심 부분에 대한 코딩 소스부분

```
if a>1 or a1>1 then
   search=replace(search,"not","-")
   search=replace(search,"-","-")
   search=replace(search,"'","&quot;")
   split_search=split(search,"-")
   scount=cint(ubound(split_search)+1)
   for i=1 to scount
     k=i-1
     if i>1 then

        text1=text1&" and keyword not like '%"&split_search(k)&"%'"
        else
```

```
        text1="keyword like '%"&split_search(k)&"%'"
      end if
    next

    text1="("&text1&")"
    stext=text1
    stext="("&stext&")"

 sql= "select * from jser where  "&stext&" order by cool,tit"
rrs.open sql,con

   elseif b>1 or b1>1 then
   search=replace(search,"or","+")
     search=replace(search,"+","+")
   search=replace(search,"'","&quot;")
    split_search=split(search,"+")
   scount=cint(ubound(split_search)+1)
   for i=1 to scount
```


배열을 통한 사용자 키워드에 대한 AND, OR, NOT 검색 가능 그 외 특수문자는 치환문장을 이용한 검색 결과 처리

Fig.3 search result

6. 텍스트마이닝을 이용한 불용어 추출의 필요성

현재 통용되고 있는 일반적인 텍스트마이닝 방법은 원하는 정보의 키워드(key word)를 몇 개 입력한 후, 이 키워드를 지니고 있는 문서를 찾는 방법이다. 이 방법은 데이타베이스에 있는 문서의 양이 적을 때는 유용하지만, 데이타베이스 내용의 증가와 더불어 너무 많은 문서를 찾아 주는 문제점이 노출되고 있다. 예를 들어 평범한 단어를 신문기사 데이타베이스에서 키워드로 검색하면 수만 건의 기사가 검색되어, 검색된 기사의 정보 가치가 없는 것이 현실이다. 최근에는 효율적인 텍스트마이닝을 위해 여러 가지 방법이 고안되고 있는 데, 대표적인 방법이 한 두 개의 키워드 검색보다 자연어로 되어 있는 전문(全文: full text)을 이용하여 검색하는 방법이다. 예를 들어 한 논문을 연구할 때 이 논문 전체를 질문으로 하여 '이 논문의 내용과 관련이 있는 모든 정보를 찾으시오'라고 하는 방법이다. 이 방법은 각 문서 안에 있는 텍스트의 연관성을 이용하여 주어진 질문에 가장 적합한 문서들을 검색하여 주는 기능을 제공한다. 최근에는 Asparoukhov & Krzanowski (2001)는 대표적으로 많이 이용되는 13가지의 판별분석모형을 의학실험에서 나타나는 이항데이타에 적용해 비교하는 논문을 발표하였다. 그 결과는 데이타 양의 많고 적음이나 변수의 수가 많고 적음에 따라 각 모형의 장단점이 있어 어느 한가지 방법이 모든 데이타에 대해 우위에 있지는 못한 것으로 나타났다. 텍스트마이닝은 대개 많은 변수와 대량의 데이타가 있는 경우가 대부분으로 아직 만족스러운 통계적 판별분석 모형이 출현하고 있지 못한 것이 현실이다(Lee & Kantor (1991), (1998), 이정진(2000)). 선진국에서는 정보화사회 특히 전자도서관(digital library)의 현실화를 위해 텍스트마이닝 분야에 많은 연구가 진행되고 있지만, 우리 나라에서는 이러한 연구가 거의 이루어지고 있지 못하다. 텍스트마이닝은 해당 언어의 구조연구와도 밀접한 관계가 있어 향후 우리 나라의 정보회사회 발전을 위해서 이 분야의 집중적인 연구가 시급히 요구된다.

텍스트마이닝은 비구조적인 테이터안에서 테이터끼리의 관계와 패턴을 추출하여 그 내용을 자동 분류하고 새로운 지식을 생성하여 작업에 활용되고 있다. 오늘날 대부분의 정보들의 확실히 구조가 잡히지 않은 텍스트 형태로 존재하므로 그 내용을 정확히 파악하기 위해서는 내용끼리의 연관 관계와 패턴을 파악하여 정확한 정보의 추출과 불필요한 정보를 제거하여 보다 요구자의 필요한 정보를 손쉽게 검색함으로써 능

률적인 일처리를 할 수 있다.

따라서 기존의 텍스트마이닝 기술체계는 주로 특성추출(feature extraction)를 통하여 특성백터(feature vector)를 생성하는게 특징인데, 여러번 나온 단어를 중요도가 높은 단어로 간주하고 가중치를 부여하여 정보와 지식을 발견하고 그 내용을 분류화, 군집화를 시켜 새로운 지식을 생성하는데 그 정확성이 접속사나 관사, 형용사등 우리말의 특성에 따라 검색결과가 다소 떨어진다. 그래서 본 연구에서는 정보를 찾고자하는 요구자의 검색어에서 접속사, 관사, 형용사등 불필요한 불용어를 제거하여 보다 정확한 검색어를 추출하고 기존의 텍스트 마이닝 기술을 이용하여 검색어의 연관관계가 높은 중요도에 가중치를 두어 상단에 랭킹함으로써 최적의 결과를 사용자에게 제공함과 동시에 능률적인 업무처리가 향상이 될 것이다. 그러므로 최적의 검색어를 추출하는 것은 기존의 불필요한 정보나 과다한 정보 검색결과로 정보 검색의 비효율을 개선하고 사용자가 요구한 키워드가 동음이어나 단어 중간부분이 같을 때의 오검색을 최소로 줄여주므로써 사용자 편익을 증진 시킬 수 있다.

Ⅲ. 텍스트마이닝 이용한 효율적인 알고리즘 설계

텍스트마이닝에서 가장 일반적으로 사용하는 기법은 특성벡터(feature vector)를 이용하는 것이다. 이 방법은 특성추출(feature extraction)과정을 통하여 텍스트에 대한 특성벡터를 생성하게 된다. 따라서, 텍스트 분석 기반이 되는 것이 바로 특성추출에의한 특성벡터이며 이의 통계수치는 각 분석기법들의 근거가 되는 것이다.

텍스트마이닝 기술체계는 자연어처리, 정보추출, 시각화, 데이터베이스 그리고 기계학습의 분야를 포함하고 있다.

이중에서 비/반정형 데이터에 대하여 자연언어처리(Natual Language Processing)기술과 문서 처리 기술을 적용하여 유용한 정보를 추출, 가공하는 것을 목적으로 하는 기술이다. 본 연구에서는 최적어 검색결과를 최단시간에 사용자에게 제공함으로써 보다 능률적인 업무처리와 다양한 분야에 빠르게 적용할수 있는 최적의 알고리즘을 이끌어 내는게 주요 목표이다.

그러기 위해서는 기존의 검색기법의 처리 보다 효율적인 방법을 연구 함으로써 텍스트로 이루어진 문서들중 학습을 통한 새로운 정보를 불특정 다수의 문서 내에서 사용자가 의도한 문서를 보다 정확히 찾아내기 위해서 키워드에서 먼저 불용어를 제거하여 검색결과의 특성추출를 사용자 의도와 가장 가까운 정보와 최단시간에 처리하는 알고리즘을 연구 하는 게 주요 목적이다.

먼저 가장 핵심부분인 검색 전 키워드에서 불용어 제거후 연관단어 검색과 기존 방법의 차이점과 실제 프로그램 구현한 후 얼마나 효과적으로 문서를 검색 할 수 있는지 실험을 통하여 테스트를 하였다. 또한 기존 상업적으로 성공한 구글 검색엔진에서 사용하는 검색시스템과 새로운 알고리즘을 적용한 검색결과의 정확성을 살펴보았다.

1. 효율적인 텍스트마이닝 알고리즘 처리

기존방법에서는 불용어를 검색후 처리함으로써 검색 결과가 상당히 많았으며 연관없는 단어가 포함된 문서가 검색 되므로써 검색 결과의 정확도가 많이 떨어졌다. 그러나 제안한 알고리즘에서는 요구자가 입력한 키워드에서 불필요한 불용어를 불용어 사전(불용어가 저장된 데이터베이스) 이용하여 제거하고 정보가 저장된 데이터에서 키워드에 합당한 분류를 찾은 후 해당 그룹에서 검색, 추출하여 요구자에게 출력 결과를 나타낸다.

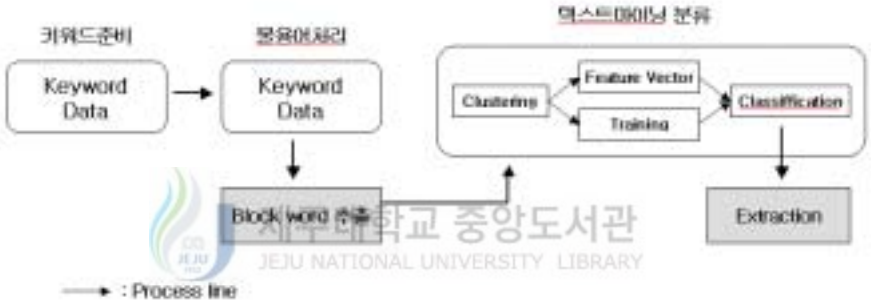

Fig.4 The Proposed flow diagram

2-1 문서검색 효율적 검색기법

정확한 검색결과를 사용자에게 제공 하려며는 검색어에서 핵심 키워드를 추출하는 것이 가장 우선시 되어야 한다. 설사 사용자 불필요한 불용어나 어순이 틀린 내용을 키워드로 입력하드래도 키워드에서 핵심 단어를 추출 하여 AND 검색 후 가장 관련이 높은 검색내용 즉 핵심단어가 가장 많이 포함된 내용을 상단에 랭킹 시키므로써 보다 효율적인 검색 결과를 도출할 수가 있을 것이다. 문서 내의 모든 단어를 숫자로 변환하기 위해서는 우선 단어 사전이 필요하며, 검색의 효율성을 위해 불용어 사전이 필요하다. 단어 사전, 불용어 사전 모두 순차적으로 Index number를 붙인 후, 검색 대상 문서를 전부 Index number로 바꾸어서 주기억장치에 저장한다.

Fig.5 효율적 검색기법 흐름도

1) 문서내에 얼마나 많은 해당단어의 출현빈도에 따라 문서분류

문서내에서 해당되는 단어의 포인트를 출현빈도에 따라 포인터로 연결되어 있다.

<Document>

1	-->	Dictionary/Appear	-->	…	-->	N
2	-->	Dictionary/Appear	-->	…	-->	N
:	:	:	-->	:	-->	:
N	-->	Dictionary/Appear	-->	…	-->	N

Fig.6 해당단어의 출현빈도에 따른 문서분류

2-2 불용어 판별기법

키워드로 입력된 데이터에서 저장된 불용어사전에서 불필요한 데이터를 키워드에서 제거한다. 단 여기서 단어와 접속사, 형용사의 중복 구분를 접미에 처리되는가에 유무에 따라 제거여부를 판단하는 알고리즘을 적용 후 키워드에서 유효한 단어를 추출한다. 가장 핵심 적인 내용은 일단 입력된 내용중에서 띄어쓰기로 입력된 단어중에서 접미에 관사, 형용사, 접속사를 불용사전 즉 불용어가 저장된 데이타베이스에서 검색 일치하는 내용이 접미에 나타나면 그 내용만 삭제하여 최적의 키워드만 추출하는 게 주요 내용이다.

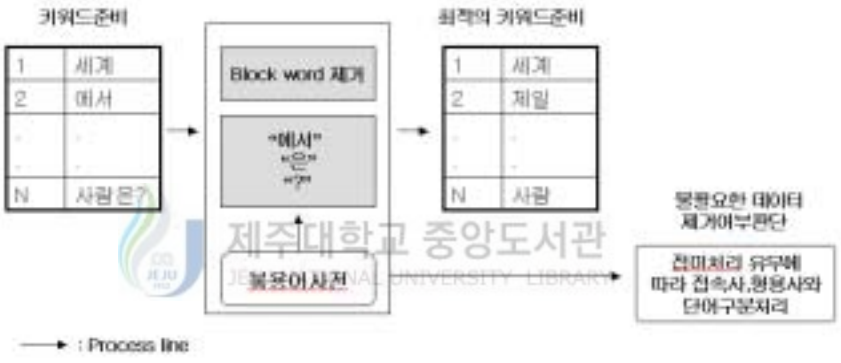

Fig.7 불용어 판별기법 흐름도

1) 세부적인 내용

전체 문서 중에서 너무 많이 출현 하게 될 경우, 문서 검색시 그다지 큰 도움을 주지 못하기 때문이다. 검색이라 함은 무수히 많은 문서들 중에서 특정 몇 개만을 찾아내야 하기 때문에 너무 많은 문서에서 발견이 된다면, 이는 특정 문서를 구분 지어 주는데 크게 도움이 되지 못하기 때문이다. 본 프로그램에서는 불용어를 판별하기 위한 공식으로 다음과 같은 방법을 사용한다.

기존 검색시..Block_Word = (전체 문서중 연관단어 출현 횟수/전체 문서 개수)*100

새로운 검색시... Block_Word: 키워드 중간에 관사, 형용사, 접속사 등

Fig.8 불용어 처리 순서 흐름도

2) 불용어 추출순서

a. 키워드는 띄어쓰기를 기준으로 입력 했을 때 처리 가능

b. 키워드에서 띄어쓰기로 된 구문을 split 문법을 이용하여 공백을 다른 문자로 치환하여 단어를 구분한다.

예) "세계에서 제일 키 큰 사람은?"

① 공백을 "-" 치환한다 꿭 세계에서-제일-키-큰-사람은?

② "-" 구분된 내용을 split배열함수을 이용하여 카운트한다 꿭. 5개 구분

③ 반복문을 적용하여 5개로 구문된 내용을 나눔꿭꿭꿭... 세계에서

④ "세계에서" 단어에서 "에서"라는 조사를 불용어 사전에 검색 후 해당되면 삭제

⑤ 이렇게 5번 반복하여 최종 내용만 AND 문으로 결합시킨다.

⑥ 결합된 문장을 최종적으로 AND 검색한다.세계&제일&키&큰&사람

※ 단 불용어 사전에서 제거 시 단어 중간에 조사가 포함이 되면 삭제 불가능하도록 하고 반드시 단어 끝에 위치하는 부분만 삭제되도록 한다.

3) 소스 표현

```
if a>1 or a1>1 then
 search=replace(search," ","-")
 split_search=split(search,"-")
```

→ ① ② 공백 치환 후 Split 1차 배열 함수

```
 scount=cint(ubound(split_search)+1)
 for i=1 to scount
 k=i-1
 if i>1 then
 text1=text1&" and keyword not like '%"&split_search(k)&"%'"
 else
 text1="keyword like '%"&split_search(k)&"%'"
 end if
 next
 sql1= "Delete * from Block where  Block="&stext&"
 rrs.open sql1,con
```

→ ③ ④ 나누어진 구문에서 불용어사전을 이용한 제거

```
 text1="("&text1&")"   stext=text1    stext="("&stext&")"
 sql= "select * from jser where  "&stext&" order by cool,tit"
 rrs.open sql,con
```

→ ⑤ ⑥ 최종AND문으로 검색

※ 키워드가 반드시 띄어쓰기로 입력이 되어야 하구 단어 중 접미사가 “아, 와, 가”등 불용어 사전에 저장된 단어가 단어에서 강제적으로 삭제됨 그러나 단어 중간에 들어 있는 내용은 불용어로 처리하지 않음.

2-3 연관단어 판별기법

문서를 검색 후 얼마나 빈번하게 핵심키워드 내용과 같이 나타나는지를 알아봄으로써 연관단어 인지를 판단할 수가 있다. 즉 전체 문서에서 사용자가 입력한 검색어가 출현하는 빈도가 많은 단어를 연관단어라고 결정할 수가 있다. 기존의 전체 대상을 1로 설정하고 전체부분에서 출현한 부분과 검색된 내용부분에서 출현한 부분과의 곱을 전체값 1에서 빼어주면 그만큼의 가중치값을 구할 수가 있다.

연관단어 = (전체값) 1 - (전체문서 내 연관단어 출현횟/전체문서개수)
* (검색문서 내 연관단어 출현개수/검색문서개수)

2-4 정확도 향상을 위한 가중치 기법

사용자가 검색을 할 때 검색 키워드가 있을 것이다. 검색 키워드가 만약 “학교”라고 했을때 찾고자하는 문서중에서 “학교”는 내용중에 하나정도 있고 “학생”이라는 내용이 상당히 많이 나타난다. 그러므로 “학교”와 “학생”은 연관단어라 볼수 있다 그래서 그 만큼의 가중치 값을 적용하여 검색시 최상단에 위치하도록 함으로써 보다 정확한 정보를 사용자에 제공하는게 가중치를 적용하는 주요 목표이다.

보통 문서내에 핵심 키워드값이 많이 포함된 문서를 상단에 랭킹시키지만 연관단어가 주로 포함된 문서중 가중치 즉 연관단어가 많이 포함된 내용 가중치 값이 큰 값을 갖고 있는 문서를 최우선적으로 표현함으로써 사용자의 핵심키워드와 연관단어를 통한 유사 내용를 부가적으로 더찾아 줌으로써 부가적인 정보 파악과 유사 검객결과와 지금의 지식 검색처럼 사용자의 의도와 연관이 있는 정보를 가중치를 이용한 검색기법으로 도출할 수가 있다.

3. 최적의 검색결과 추출

사용자가 키워드를 입력 후 키워드에 해당되는 최적의 검색결과를 얻는게 최우선이지만 오늘날 사용자가 사용한 키워드가 자기가 찾고자하는 의도 있는 키워드가 아닐수 있고 유사한 키워드를 사용할 수가 있다. 그래서 여기서는 그와 관련한 연관단어 판별기법을 이용한 키워드와 관련한 유사내용을 가중치를 주어 검색결과를 사용자가 의도하는 내용과 연관된 내용을 동시에 보여 주므로써 보다 최적의 검색 결과를 도출하는게 주요 연구이다. 오늘날 지식검색처럼 키워드와 관련된 지식을 따로 검색하여 상단에 보여주므로써 사용자에게 큰 호응을 얻었다. 그런 것처럼 키워드와 관련된 연관단어를 가중치를 부여하여 지식검색처럼 상단에 동시에 보여주므로써 최적의 검색결과을 추출하는데 큰 도움이 될것이라 본다.

크게 두가지로 검색부분을 나누고자 한다. 첫째 사용자가 입력한 키워드를 불용어를 추출하여 핵심 키워드를 추출하여 검색결과를 나타나는것이고 둘째 핵심 키워드를 추출하여 연관단어를 가중치를 두어 검색결과를 보다 다양하게 얻는게 주요 목표이다.

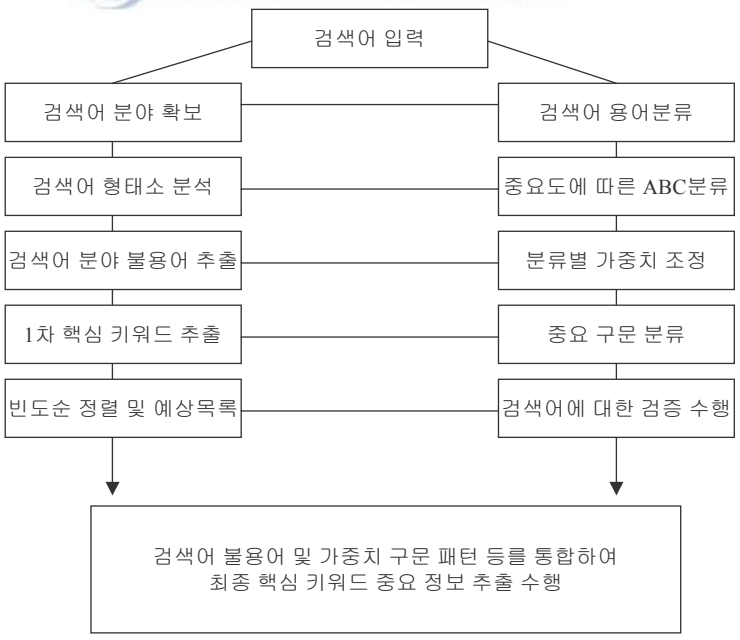

Fig.9 검색처리 흐름도

Ⅳ. 구현 및 실험 결과

본 논문에서 제안한 불용어 제거후 핵심 키워드를 추출하여 최적어 검색어를 추출하여 연관단어에 가중치를 부여하여 최적의 검색결과를 상단에 랭킹시키므로써 사용자 얻고자하는 검색결과를 도출시키는게 주요 목적이다.

1. 실험 환경

본 논문에서 구현한 불용어를 제거 기법을 이용하려며는 기존에 웹프로그램 언어로 ASP로 불용어가 저장된 데이타베이스에서 불러와서 SQL를 이용하여 접미부분에 적용된 불용어를 제거하는 방법을 이용하였다.

본 논문에서의 실험 환경은 CPU2.4Ghz, 메모리 512MB를 가지는 펜티엄 4 PC와 데이타베이스를 OFFICE에 기본적인 ACCESS 를 이용했고 Windows 2000SETVER OS 환경에서 실험을 하였다. 실험을 위한 시스템 구현 프로그래밍 언어로는 ASP(Active server page)를 사용하여 구현 하였다.

Table. 1 Experiment Environments

시스템 사양	Pentium Ⅳ processor, 512MB RAM
운영체제	Windows 2000SERVER
프로그램 언어	ASP
사용된 데이타베이스	ACCESS

2. 제안 알고리즘을 이용한 성능평가

실험을 위해서는 우선 검색을 원하는 문서들이 있어야 한다.
보다 객관적인 평가를 위해서 다양한 정보가 들어있는 문서를 무작위을 추출하여 데이타베이스에 저장하였고 주로 웹상에서 정보를 얻었으며 기존에 주로 사용하는 컴퓨터 관련 업체, 뉴스 기사, 생활정보등을 이용하였다.
문서들은 /data 이 저장하여 사용한다.
불용어 사전은 Block.db, 문서는 data.db 로 사용한다.

Table. 2 테스트에 사용된 데이터

<테스트에 사용된 데이터>

사용된 문서 파일 내용 개수	1,254개
불용어 사전 파일 내용 개수	85개
사용된 사전 파일 내용 개수	856개

① 실험1

불용어 제거하지 않고 단순단어 검색

② 실험2

불용어 제거하여 핵심키워드 적용후 단어 검색

③ 실험3

불용어 제거하여 핵심키워드 적용후 연관단어 가중치 검색

< 실험1 >

a. 검색하고자 하는 키워드: 자바를 이용한 채팅

b. 검색사용된 파일 : jser.mdb

c. 검색 속도체크: Clocklt 1.3 스톱워치 프로그램

검색시 검색결과의 처리 속도와 사용자가 원하는 문서의 정확도를 기준으로 조사하였다. 정확도는 키워드인 “자바를 이용한 채팅”에 해당되는 문서결과가 나왔는지 엉뚱한 결과가 나왔는지를 검색한다.

검색속도는 분당 초속도를 나타내고 연관성은 만족도를 1로 놓고 연관단어 갯수로 판단한다.

1) 검색속도와 연관단어 갯수

Table. 3 검색속도 및 연관단어 개수(실험1)

< 검색속도>

사용된 검색어	검색속도
“자바를 이용한 채팅”	0.97초

< 연관단어 갯수 >

순위	연관포함 갯수	검색된문서
1	8	nawana.com
2	7	webpluspack.com
3	4	dgsoft.co.kr
4	3	wingtip.co.kr

연관단어: 자바, 채팅, 게임, JAVA, 웹, 네트워크, 인터넷, 대화방, 회의, 서버, 클라이언트 라고 11개의 검색어를 기준으로 정하고 이중에서 연관단어가 포함된 갯수를 평가한다.

< 실험2 >

a. 검색하고자 하는 키워드: 자바를 이용한 채팅

b. 검색사용된 파일 : jser.mdb

c. 검색 속도체크: Clocklt 1.3 스톱워치 프로그램

검색시 불용어를 제거후 핵심 키워드만 찾아서 검색한다.

Table. 4 검색속도 및 연관단어 개수(실험2)

< 검색속도>

사용된 검색어	검색속도
“자바를 이용한 채팅”	1.02초

< 연관단어 갯수 >

순위	연관포함 갯수	검색된문서
1	10	dgsoft.co.kr
2	8	electricisland.com
3	7	wingtip.co.kr
4	6	okjsp.pe.kr

"자바를 이용한 채팅“이라 검색하면 **”자바“** + ”를“ + ”이용한“ + **”채팅“** 이라고 검색시 불용어 사전에 등록한 를 과 이용한를 제거하여 자바 채팅이라 검색한다.

< 실험3 >

a. 검색하고자 하는 키워드: 자바를 이용한 채팅

b. 검색사용된 파일 : jser.mdb

c. 검색 속도체크: Clocklt 1.3 스톱워치 프로그램

검색시 불용어를 제거후 가중치를 부여하여 검색한다.

Table. 5 검색속도 및 연관단어 개수/가중치(실험3)

< 검색속도>

사용된 검색어	검색속도
“자바를 이용한 채팅”	15.02초

< 연관단어 갯수 및 가중치 >

순위	연관포함 갯수	검색된문서
1	10	dgsoft.co.kr
2	8	electricisland.com
3	7	wingtip.co.kr
4	6	okjsp.pe.kr

순위	연관단어	가중치
1	자바	0.89002
2	채팅	0.88900
3	게임	0.18999
4	JAVA	0.17999
5	웹	0.15999

연관단어가 정해진 상태에서 비교 검색을 했을 때 포함된 내용이 직접 적용한 키워드가 많은 가중치가 적용이 되었으며, 보다 연관단어 높은 검색어를 가중치를 이용하여 찾을 수 있었다.

2-1. 불용어 제거후 검색 성능 비교

불용어 제거하지 않고 검색한 결과와 제거후 검색한 결과 정해진 연관단어가 얼마나 많이 포함하는지 비교하였고 검색속도도 비교하였다. 검색결과의 정확도는 사용자의 의도와 기준이 각자 다를 수 있으므로 객관적인 평가는 연관단어 포함수의 비교를 기준으로 하였으며 그만큼 연관단어가 많이 포함이 되며 관련된 정보도 많을 것이라는 객관적 기준으로 비교 하였다.

1. 속도 비교

키워드를 입력후 검색결과까지 표현되는 시간을 Clocklt 1.3 스톱워치 프로그램으로 측정하였다. 검색속도가 빠르면 그만큼 사용자에게 효율성을 더 줄 것이다.

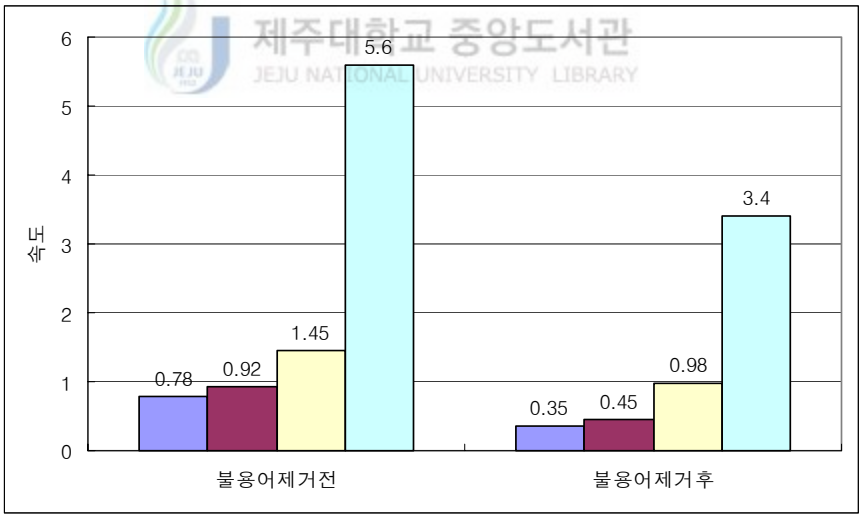

Fig.10 속도비교 흐름도

2. 연관단어 포함 비교

연관단어 포함 개수에 비교한다. 연관단어가 많이 포함되며 그 만큼 검색 키워드와 많이 관련된 내용이 검색될 것이라 본다.

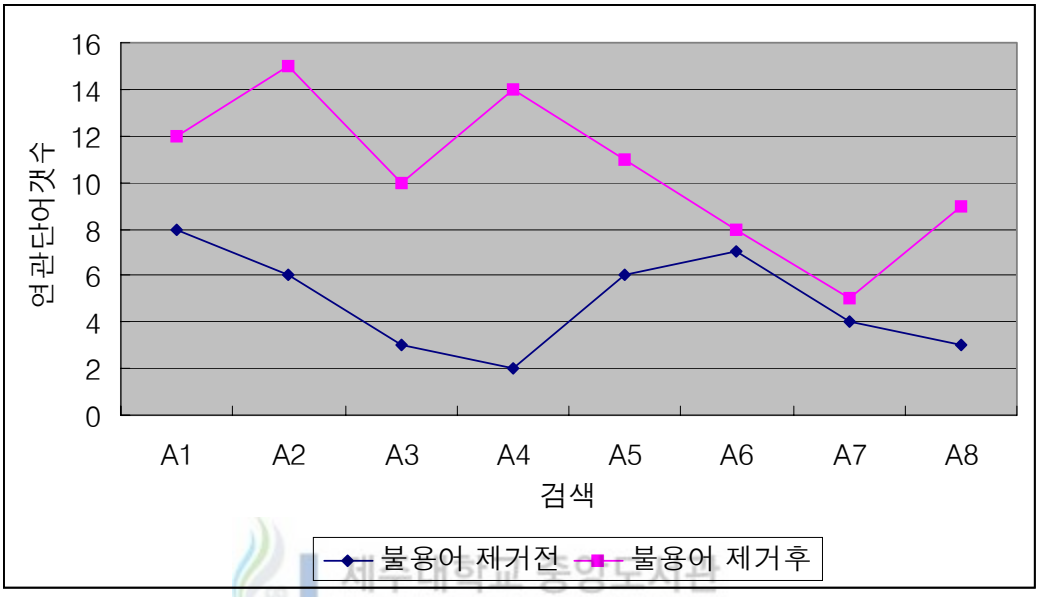

Fig.11 연관단어 포함개수 비교

2-2. 연관분석 및 형태소처리 실험

키워드가 검색 대상 문서에는 분명히 존재하는데 검색엔진이 해당 문서를 찾아내지를 못하는 문제가 종종 발생한다. 특히 외산 검색 솔루션인 경우, 한국어 처리 부분이 취약하기 때문에 이러한 문제가 많이 발생하고 있다.

이러한 문제점을 해결하기 위해서는 형태소 분석처리를 올바르게 처리 하는게 우선시 된다. 그러기 위해서는 띄어쓰기와 보정처리 기능을 이용한 기초적인 형태소 처리 기술을 적용하여 보다 검색 시 문제를 낮추어 보기로 하겠다.

1. 연관분석

웹 링크 구조들의 이러한 속성들은 연구자들에게 허브(hub)라는 불리는 웹페이지들의 또 다른 중요한 카테고리들을 고려하도록 유도 하였다.

신뢰할만한 링크의 집합을 제공하는 웹페이지 혹은 집합이다. 허브 페이지들은 스스로 돌출하지 않고, 그들을 링크하는 페이지 수도 적다. 그러나 그것들은 공통된 주제에 대한 두드러진 사이트들의 집단에 대한 링크를 제공한다.

HITS(Hyperlink-Induced Topic Search)는 허브를 사용하는 알고리즘이다

첫째, HITS는 인덱스 기반 검색엔진으로부터 예를 들어 처음 200페이지에서 모아진 질의 용어를 사용한다. 이 페이지들은 뿌리 집합(root set)을 구성한다.

둘째, 가중치 확산 단계를 시작한다. 이것은 허브나 신뢰사이트의 가중치를 결정하는 반복적 과정이다.

HITS 알고리즘은 주어진 검색 주제에 대한 거대한 허브 가중치와 거대한 신뢰 가중치를 가진 페이지들의 짧은 목록을 산출한다. 많은 실험들은 넓은 범위의 질의들에 대해 HITS가 놀랍게도 좋은 검색 결과들을 제공하는 것을 보이고 있다.

HITS 알고리즘에 기반한 시스템들은 Clever가 있고 비슷한 원리에 기반한 또 다른 시스템은 Google이 있다. 웹 링크들과 텍스트 문맥을 분석함으로서 그런 시스템들은 Alta vista와 같은 용어 인덱스-기반 엔진들에 의해 생성된 검색결과와 Yahoo와 같은 인간 존재론에 의해 생성된 검색 결과들보다 더 좋은 질의 검색 결과들을 얻어낼 수 있다고 보고되었다.

2. 형태소처리

형태소 분석이란 한국어 텍스트를 입력으로 하고 그것을 형태소 단위 - 사전의 표제어 단위로 분석하여 사전에 있는 정보(품사정보)화 함께 출력해 주는 것이다

가장 중요한 일은 구문 구조 분석기 이후의 처리기가 한 어절에 대한 정보를 사전에서 참조할 수 있도록 사전의 표제어(원형)를 정확하게 찾는 것이다.

1) 형태소처리 실험

불용어사전에는 품사 정보가 대분 포함되어있다 보통 접미부분에 품사가 포함이 되므로 핵심 키워드를 추출하기 위해서는 형태소를 제대로 알아야 한다, 그리고 형태소 처리형태를 이용한 불용어 제거 프로그램를 개발하여야 하므로 형태소에 입가한 불용어 제거 실험을 해보았다

2) 형태소 예

부사	빨리
부사 + 조사	빨리는
감탄사	아이구
명사	현대전자
관형사	새, 헌
명사 + 접미사 + 조사	사람들을
명사 + 조사	사람을
대명사 + 조사	그것을
대명사 + 접미사 + 조사	그것들을

보통 부사, 조사, 관형사, 접미사, 감탄사 등을 모두 제거하거나 아니면 대명사를 제외한 나머지 부분을 불용어 사전(데이타베이스)를 제거한 검색 결과를 비교한다.

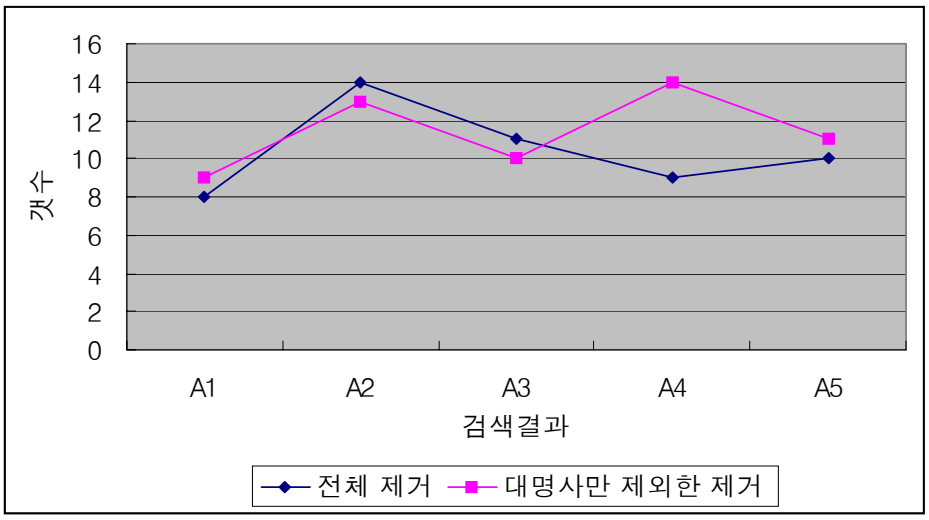

Fig.12 형태소 처리

4. 기존검색과 비교분석 평가

기존에 검색엔진과 비교분석을 할 수 있다 속도나 전체 분석은 객관적 평가는 어렵다. 같은 조건하에 결과가 다른 점을 파악해야 하므로 기존 검색과 비교분석은 불가능하고 단 같은 키워드를 입력 시 연관단어 결과 치 만을 비교분석하였다 기존 검색은 연관단어는 어느 정도 검색 했는가를 기준으로 하여 비교분석 하였다.

1) 연관 단어 검색 비교

기존 검색 네이버, 야후와 비교 검색 하였다 검색어를 입력 첫 번째 페이지만 비교분석하였다.

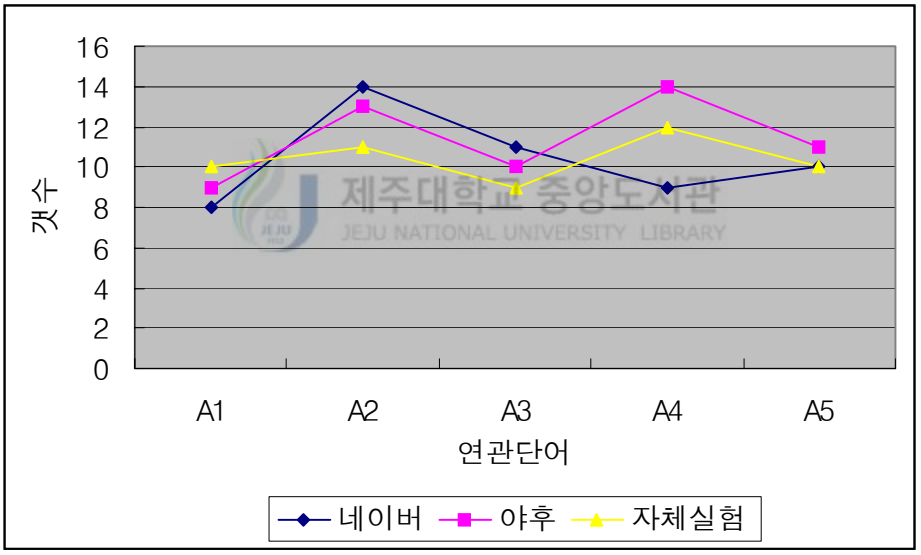

Fig.13 기존검색과 비교검색

정확한 비교 검색은 어렵지만 기존 검색과 별반 차이가 없었으며 기존의 검색엔진은 파트별로 검색해주므로 좀 많은 검색을 한다. 지식검색, 뉴스 검색, 포스트 검색등 한 개인 데이타베이스를 이용한 자체 실험과 단순 비교는 상당히 어려웠다 단 연관단어 갯수만 비교 검색만을 비교한다면 절대 떨어지지 않는 결과를 만들었다.

V. 결 론

비구조적 문서가 다량이 포함된 웹 문서 검색 시 현재 연구되어 온 텍스트마이닝 시스템이, 보다 오류율과 속도 개선과 더불어서 그동안 웹 중심의 자연어 텍스트문서를 자동 분석하기 위한 효율적 시스템에 새로운 접근이 이루어 졌으며 기존 검색 시스템의 여러 문제점과 한계들을 효과적으로 극복하고 사용자의 편익 증가 시킬 수 있다.

오늘날 검색엔진은 상당히 많고 여러분야로 나누어서 검색을 한다. 또한 검색된 정보를 통한 지식발전과 새로운 정보를 통한 부가가치가 향상이 개인 및 나아가 나라 발전에 도움을 줄 것이라 믿고 있다. 또한 현재 인터넷을 사용하는 인구의 대부분은 인테넷 에서 검색엔진을 통한 새로운 정보를 찾고 있으며 그 정보를 통한 지식을 서로 교환하여 질 높은 생활을 추구해 나갈수 있다.

그러기 위해서는 찾고자 하는 정보를 손쉽게 접근하고 더불어서 연관 있는 정보를 동시에 제공함으로써 보다 부가적인 비용과 새로운 정보교환이 손쉽게 할수 가 있으며 이 알고리즘을 통한 다양한 곳에 적용을 하여 보다 능률적인 작업환경을 제공해 줄 것이라고 믿는다.

또한 비구조적인 데이터들은 아직도 생산이 되어지고 있으며 무수히 많은 연관된 문서와 새로운 정보와 유기적으로 결합한다. 앞으로는 누가 정보를 많이 갖는냐가 문제가 아니라 보다 질 높은 정보를 갖느냐가 핵심이 될 것이라 본다. 그러기 위해서는 정확한 정보를 빠르고 연관성이 높은 정보를 제공함으로써 정보검색시 높은 서비스 만족감과 관련된 부가적인 비용을 줄여 효율적인 검색시스템 개발이 기대가 된다. 다만 앞으로는 시멘틱 웹기술을 이용한 지식검색 기능으로 한 단계 발전 할 수 있는 연구와 여전히 다량의 문서 검색 시 속도 개선 문제점을 있으므로 분산 환경시스템을 이용한 텍스트마이닝 연구가 진행이 되어야 할 것 이라 생각된다.

[참고문헌]

[1] 노나키 이쿠지로, Michael Polanyi, Delphi Group 1998.

[2]Yang, Y., "An Evaluation of Statistical Approaches to Text Categorization". Journal of Information Retrieval, 1999.

[3] Lee, H-Y, "Text Mining-Knowledge Discovery from Text", Trend in Knowledge Discovery from Databases, 29th June 1999.

[4] 구글 개발자들이 쓴 'The anatomy of large scale search engine' 논문

[5] 텍스트마이닝 기술을 적용한 대용량 온라인 문서데이타의 계층적 조직화 기법", 서울대 학위논문

[6] Soumen Chakrabarti , "Mining the Web Discovering knowledge from Hypertext Data" , MORGAN KAVFMANN PUBLISHERS

[7] Soumen Chakrabarti , "Mining the Web Discovering knowledge from Hypertext Data" , MORGAN KAVFMANN PUBLISHERS

[8] Dan Sullivan , "Document Warehousing and Text Mining Techniques for Improving Business Operations, Marketing and Sales" , WILEY COMPUTER PUBLISING

[9] Tom M. Mitchell(Carnegie Mellon University), "MACHINE LEARNING", The McGraw-Hill Companies, Inc

[10] Sergey Brin and Lawrence Page, “The anatomy of large scale search engine”

[11] Rosie Jone, Andrew McCallum, Kamal Nigam and Ellen Riloff, "Bootstrapping for Text Learning Tasks", IJCAI-99 Workshop on Text Mining Foundations, Techniques, and Applications

[12]Kanagasa R. and A-H. Tan, "Topic Detection, Tracking and Trend Analysis Using Self-Organizing Neural Networks". in Proceedings, Fifth Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'01), Hong Kong, pages 102-107, 2001.

[13] Tan, A-H, "Predictive Self-Organizing Networks for Text Categorization". in Proceedings, Fifth Pacific- Asia Conference on Knowledge Discovery and Data Mining (PAKDD'01), Hong Kong, pages 66-77, 2001.

[14] Lakshmi, V., A-H Tan, C-L Tan, "Web Structure Analysis for Information Mining. Accepted by ICDAR'01 Workshop on Web Document Analysis", Seattle, September 10-13, 2001.

감사의 글

힘차게 시작했던 2년간의 대학원 생활이 벌써 하루 하루 흘러 어는덧 아쉬운 끝으로 다가 왔습니다. 2년 전만 해도 대학원이라는 생활이 참으로 힘들거라는 두려움으로 시작했지만, 생활해 가면서 교수님들과 선배님들 그리고 같이 공부하고 연구하던 동기들이 있어 참으로 힘든 줄 모르고, 열심히 대학원 생활을 할 수 있었습니다.

이제 대학원 생활의 마지막 자리에 서면서, 언제나 관심을 가지고 끝가지 성심 성의껏 지도해 주신 김장형 지도 교수님과, 이렇게 좋은 결과를 맺을 수 있도록 조언해 주신 안기중 교수님, 곽호영 교수님, 송왕철 교수님, 변상용 교수님, 이상준 교수님, 김도현 교수님, 변영철 교수님께 먼저 감사의 마음을 전합니다.

또한 연구실에서 공부하는 동안 언제나 대학원 생활과 연구에 대해 조언을 아끼시지 않으시고, 관심을 가져주신 박사 강진석 선생님, 박사과정인 강영도 선생님, 김정효 선생님, 강길봉 선생님, 변태보 선생님, 고봉수 선생님, 양동호 선생님께도 진심으로 감사 드립니다.

그리고 같이 연구하고 생활하던 석사과정 임정홍 선생님과 멀티미디어 연구실이 여러 식구들과 대학원에 같이 들어와서 함께 열심히 공부했던 저의 동기들 및 학과 사무실에서 저희들을 위해 애써주신 정은경 조교 선생님께도 감사의 마음을 전하는 바입니다.

2년이란 짧은 대학원 생활동안 이처럼 감사의 마음을 전할 수 있는 멋진 교수님들과 좋은 선배님들 그리고 함께 했던 동기들이 있어 대학원 생활이 저희 기억속에서는 언제나 좋은 기억으로 남을 수 있을 것 같습니다.

끝으로 언제나 가슴 졸이면서 아들의 대학원 생활을 뒷바라지 하고, 뒤에서 항상 따뜻하게 맞아주고, 격려의 말씀을 아끼시지 않으시던 모든 분들에게 마지막으로 진심 어린 감사의 말씀을 드립니다.

마지막으로 언제나 이 모든 분들께 살아가면서 언제나 좋은 일만 있으시길 빌겠습니다. 감사합니다.