의미망 지식베이스를 이용한 개념기반 정보검색기법의 성능연구

A Study on the Performance of the Concept-Based Information Retrieval Model Using a Domain Knowledge Base

노영희(Young-Hee Noh)*, 정영미(Young-Mee Chung)**

목 차

초 록

개념기반 검색모형의 주요 요소는 개념확장 알고리즘과 의미망 구조의 지식베이스이다. 따라서 본 논문에서는 가장 효과적인 개념기반 검색모형을 제시하기 위하여 개념확장 알고리즘을 비교하는 실험과 지식베이스의 성능을 비교하기 위한 실험을 수행하였다. 실험대상 문헌집단의 크기에 따라 1차 실험과 2차 실험으로 나누고 각 실험은 다시 개념확장 알고리즘 비교실험과 지식베이스 비교실험으로 구분된다. 알고리즘 비교실험에서는 순차적 bnb 알고리즘, 병렬적 bnb 알고리즘, 그리고 홉필드 넷 알고리즘을 비교 평가하였다. 또한 세 개의 개념확장 알고리즘을 통계적 기법인 불논리 검색기법의 단점을 보완하기 위해 등장한 P-norm 검색모형과도 비교 평가하였다. 지식베이스 비교실험에서는 다양한 방법으로 구축된 지식베이스 즉, 문헌기반 지식베이스, 시소러스기반 지식베이스, 통합형 지식베이스, 그리고 동의어 처리형 지식베이스를 비교 평가하였다.

ABSTRACT

The major elements of a concept-based retrieval model are a concept exploration algorithm and a knowledge base structured as a semantic network. This study performed two experiments comparing three concept exploration algorithms and then four knowledge bases in order to find the most effective concept-based retrieval model. Experiments are divided into two groups according to the size of a text collection. Each experiment is subdivided into algorithm-comparing experiment and knowledge base-comparing experiment. In the algorithm-comparing experiment, the performances of the sequential bnb algorithm, the parallel bnb algorithm, and the Hopfield net algorithm were evaluated and compared with the P-norm retrieval model which is an extended the Boolean retrieval model. In the knowledge base-comparing experiment, the performances of various knowledge bases were measured using a selected algorithm which had been identified as the most effective in the first experiment. The four types of knowledge bases developed for this experiment are document-based, thesaurus-based, integrated, and synonym processed.

키워드 : 개념기반 정보검색, 의미망 지식베이스, bnb 알고리즘, 홉필드 넷 알고리즘

* 이화여대 국제정보센터 실장
** 연세대학교 문헌정보학과 교수
■ 논문 접수일 : 2000년 7월 3일

1 서 론

대부분의 상업적인 정보검색시스템은 여전히 전통적인 도치색인파일과 불논리 검색 기법에 의존하고 있다. 통계적 정보검색 기법 중 특히 효과적인 확률검색 기법은 검색성능을 향상시키기 위해 그 공식을 다양하게 변형하며 사용되어 왔음에도 불구하고 일반화되지 못하고 있다(Maron, & Kuhns 1960; Bookstein, & Swanson 1975). 확률검색은 적합할 확률과 부적합할 확률을 기반으로 하고 있다는 점과 용어의 독립성이 보장되어야 한다는 점이 문제점으로 지적되고 있다

최근에 본격적으로 연구되기 시작한 개념기반 정보검색 모형은 기존의 통계적 검색모형의 단점을 보완할 수 있는 차세대 검색모형으로 간주되고 있다. 개념기반 검색모형은 일반적으로 시스템에 입력되는 문헌 데이터베이스로부터 지식베이스를 자동으로 구축하고, 이 지식베이스를 대상으로 개념확장을 수행한 후 문헌 데이터베이스로부터 관련 정보를 검색한다. 따라서 개념기반 검색모형의 성능을 좌우하는 주요 요소는 지식베이스와 개념확장 알고리즘이라고 할 수 있다.

의미망 구조의 지식베이스를 기반으로 개념확장을 수행하는 알고리즘을 개념확장 알고리즘이라 하며, 개념확장 알고리즘으로는 bnb 알고리즘(branch-and-bound expansion activation algorithm)과 홉필드 넷 알고리즘(Hopfield net algorithm)이 사용되고 있다.

bnb 알고리즘은 적용되는 지식베이스에 따라 크게 경험적(heuristic) bnb 알고리즘과 순차적(sequencial) bnb 알고리즘으로 구분할 수 있는데, 전자는 주로 전통적인 시소러스에 적용되고 용어간의 관계 정의에 따라 개념확장을 수행한다. 후자는 의미망 구조의 문헌기반 지식베이스에 적용되어 지식베이스 내 용어간의 의미 거리에 따라 개념확장을 수행한다. 홉필드 넷 알고리즘은 신경망 구조의 지식베이스에 적용되는 개념확장 알고리즘이다.

본 연구에서는 인간 전문가의 주제영역지식을 요구하지 않으면서도 유용한 지식베이스를 구축할 수 있는 방안을 모색하고자 하며, 이러한 지식베이스를 이용하여 개념확장을 할 수 있는 효과적인 개념기반 정보검색 모형을 제시하고자 한다. 이를 위해 위에서 설명한 순차적 bnb 알고리즘, 순차적 bnb와 개념확장 방식이 다른 병렬적 bnb 알고리즘, 그리고 홉필드 넷 알고리즘의 검색성능을 비교 분석하였다.

개념기반 검색모형의 검색성능은 개념확장 대상이 되는 지식베이스에 따라서도 달라질 것이다. 따라서 본 연구에서는 효과적인 지식베이스 구축을 통하여 검색성능을 향상시키고자 하였다. 이를 위해 개념확장 대상이 되는 의미망 구조의 지식베이스를 다양한 방법으로 구축하였는데, 실험대상 지식베이스는 문헌기반 지식베이스, 시소러스기반 지식베이스, 통합형 지식베이스, 그리고 동의어 처리형 지식베이스이다.

2 개념기반 검색모형

2.1 bnb 알고리즘을 이용한 개념확장

지식베이스를 의미망 구조로 구축하였을 때 개념확장의 방법론이 검색성능을 결정한다 의미망 구조로 지식베이스를 잘 구축하였다 하더라도 개념확장 모형이 부적합하다면 좋은 검색

효율을 기대하기는 어렵다.

최근에 의미망에 대한 효율적인 추론 알고리즘 개발의 중요성이 강조되기 시작했는데 소와(Sowa 1991)는 개념확장 방법으로서 두 용어간의 의미 거리를 채택했으며 효율적인 추론 알고리즘을 개발할 것을 제안했다.

의미망 구조의 지식베이스에 대한 추론과정을 정보검색에서는 확장 활성화(spreading activation)라고 하며, 이 알고리즘은 인공지능 기반 시스템에 채택된 상태공간 항해(state space traversal)의 변형이다. 추론은 링크를 따라 진행되고, 초기 노드와 연결된 노드를 따라 항해하며, 일반적으로 보다 짧은 경로가 긴 경로보다 선호된다.

깊이우선탐색(DFS: depth-first-search), 너비우선탐색(BFS: breadth-first-search), bnb(branch-and-bound) 확장 활성화 탐색, 그리고 A* 탐색과 같은 전통적인 탐색 기법은 상태공간 항해에 사용되어 왔다(Winston 1984). 이 중에서 A* 탐색과 bnb 확장 활성화 탐색은 의미망상에서 매우 적합한 용어들을 중심으로 확장해 가기 때문에 DFS나 BFS보다 많이 사용되어 왔다.

이 가운데 개념확장 방법으로서 최적의 기법으로 평가되고 있는 bnb 알고리즘은 확장대상이 되는 지식베이스가 용어간의 BT, NT, RT 관계가 명확한 시소러스인 경우와 용어간의 관계가 의미값으로만 표현된 지식베이스인 경우에 각각 다르게 적용된다. 전자를 경험적 bnb 알고리즘이라 하고 후자를 순차적 또는 병렬적 bnb 알고리즘이라 한다.

2.1.1 순차적 bnb 알고리즘

의미망 구조의 지식베이스에 적용되는 bnb 확장 활성화 탐색은 개념확장이 진행되는 동안 최단 경로를 찾기 위한 방법이며, 이용자가 제공한 용어에서 개념확장이 시작된다(Chen, & Dhar 1991). 이용자가 제공한 초기 탐색어에는 1의 가중치가 부여되며, 다음으로 이 용어들과 직접적으로 관련이 있는 이웃한 용어들을 탐색한다. 확장된 용어의 가중치는 이용자가 입력한 용어와의 링크 가중치를 기반으로 산출된다. 순차적 bnb 알고리즘의 개념확장 과정을 단계적으로 기술해 보면 다음과 같다.

(1) 이용자가 탐색문을 입력한다. 이용자가 입력한 초기 탐색어 집합이 {S1, S2, ... Sm}일 때, 의미망에 나타난 용어들 중 초기 탐색어와 일치하는 용어는 1의 가중치를 갖는다.

$$\mu_i(0) = x_i,\ 0 \le i \le n-1$$

$\mu_i(t)$는 t번 반복한 후의 노드 i의 가중치이다. 초기 탐색어에 할당된 노드의 가중치는 1이다.

(2) 순차적 bnb 알고리즘은 내림차순으로 우선순위 대기행렬인 $Q_{priority}$를 생성한다. 최초의 우선순위 대기행렬은 아래와 같다.

$$Q_{priority} = \{ S1, S2, \ldots Sm \}$$

또한, 출력 대기행렬인 Q_{output}을 생성해야 하는데, 이는 확장이 반복되는 동안 활성화 노드를 저장하기 위해서이다.

$$Q_{output} = \{\ \}$$

(3) 반복이 계속되는 동안, bnb 알고리즘은 $Q_{priority}$에서 가장 높은 가중치의 노드들을 제거하고 그들의 이웃 노드들을 활성화시키며, 다음

공식에 의해 이웃 노드들의 가중치를 산출한다.

$$\mu_j(t+1) = \mu_j(t) \times t_{ij}$$

위 식에서 $\mu_j(t+1)$는 bnb 알고리즘에 의해 확장될 새로운 노드의 가중치이고, $\mu_j(t)$는 확장 전 노드의 가중치이며 t_{ij}는 확장 전 노드와 확장될 새로운 노드의 유사도 가중치 즉, 링크 가중치이다. 새롭게 활성화될 노드의 가중치는 활성화될 노드와 활성화되기 전 노드간의 링크 가중치에 의존한다.

(4) 활성화되었던 노드는 출력 대기행렬, Q_{output}에 저장된다. 계산이 끝난 후 모든 활성 노드들은 가중치순으로 정렬되어 $Q_{priority}$에 저장된다.

(5) 두 개의 다른 노드로부터 도달되는 노드의 가중치는 두 노드간의 유사도 가중치를 합하여 산출할 수 있다. 이와 같이 의미망에서 두 개의 다른 시작 노드에 의해 도달되어질 수 있는 하나의 노드에 보다 높은 가중치를 할당하는 기법은 기타 다른 확장 활성화 탐색에 채택되어 왔다(Shoval 1985; Cohen, & Kjeldsen 1987; Chen, & Dhar 1991).

2.1.2 병렬적 bnb 알고리즘

병렬적 bnb 알고리즘은 이용자가 처음 입력한 탐색어들에 대하여 개념확장을 수행한 후 확장된 모든 용어들에 대하여 병렬적으로 개념확장을 수행한다. 이 때 이용자가 제시한 기준치 이상의 확장용어 가중치를 갖는 모든 용어들에 대하여 확장이 이루어진다. 순차적 bnb 알고리즘은 확장된 용어들을 가중치순으로 정렬한 후 가중치가 가장 높은 용어에 대하여 확장을 하는 반면에 병렬적 bnb 알고리즘은 확장된 용어들 중 이용자가 요구한 확장용어의 가중치보다 높은 가중치를 갖는 모든 노드에 대하여 병렬적으로 개념확장을 한다. 이와 같이 개념확장이 병렬적으로 발생하기 때문에 여러 번 정렬작업을 수행할 필요가 없다는 장점이 있다.

병렬적 bnb 개념확장 알고리즘은 순차적 bnb 알고리즘과 유사한 과정을 거치게 되는데, 그 과정을 구체적으로 기술하면 다음과 같다.

(1) 이용자의 탐색문을 받아들인다. 이용자가 입력한 초기 탐색어 집합이 {S1, S2, ... Sm}일 때, 의미망에 나타난 용어들 중 초기 탐색어와 일치하는 용어는 1의 가중치를 갖는다.

(2) 대기행렬을 생성한다. 이용자가 입력한 초기 탐색어들과 확장된 용어들을 임시로 저장하기 위해 초기 대기행렬(input queue), Q_{input}과 출력 대기행렬(ouput queue), Q_{output}을 생성해야 하는데, 이는 확장이 반복되는 동안 활성 노드를 저장하기 위해서이다.

(3) 반복이 계속되는 동안 병렬적 bnb 알고리즘은 Q_{input}에 있는 모든 노드들의 이웃 노드들을 활성화시키며, 앞의 이웃노드 가중치 산출 공식에 의해 이웃 노드들의 가중치를 산출한다.

(4) 앞에서 설명했듯이 새롭게 활성화될 노드의 가중치는 확장 노드 가중치와 확장 노드 및 이웃 노드간의 링크 가중치에 의존한다. 마지막으로 활성화되었던 노드는 출력 대기행렬, Q_{output}에 저장된다. 계산이 끝난 후 모든 확장 노드들은 가중치순으로 정렬되어 Q_{input}에 저장된다. 두 개의 다른 시작 노드로부터 도달되어지는 하나의 노드 가중치는 순차적 bnb 알고리즘에서와 같이 두 용어간의 유사도 가중치를 합하여 산출한다

한편, 병렬적 bnb 알고리즘에 의한 개념확장

을 중지하는 시점을 결정하는 중지 조건은 순차적 bnb 알고리즘과 동일하다. 순차적 bnb 알고리즘에서와 같이 병렬적 bnb 알고리즘은 이용자로부터 확장될 용어의 수(p)와 확장된 용어가 가져야 할 가중치(W_p)를 제공받고, W_p 이상의 모든 용어에 대하여 조건 p를 만족할 때까지 병렬적으로 활성화가 이루어진다.

2.2 홉필드 넷 알고리즘을 이용한 개념확장

홉필드 넷 알고리즘(Hopfield 1982; Tank, & Hopfield 1987)은 가중치가 부여된 단일층(single-layered)의 망에서 개념을 추론해 나가는 전통적인 방법으로서 bnb 알고리즘의 대안으로 고려되고 있다. 이 알고리즘은 병렬 완화적 탐색(parallel relaxation search)을 수행한다. 즉, 탐색이 진행되는 동안 노드들은 병렬적으로 활성화되며 활성화된 노드의 가중치는 모든 초기 탐색어로부터의 링크 가중치들을 참조한다. 홉필드 넷 알고리즘은 병렬탐색을 수행하고 수렴적(convergent) 속성 및 단일층의 구조를 갖기 때문에 원래 신경망 탐색에 적용되어 왔다. 홉필드 넷 알고리즘은 다양한 분류업무나 최적화 업무에 사용되어 왔고(Lippmann 1987; Simpson 1990), 최근에는 블랙보드기반 검색시스템에 사용되었다(Chen et al. 1993). 그러나 대규모 지식베이스망에 적용함에 있어 이 알고리즘과 다른 전통적인 탐색 기법들을 비교하는 실험은 수행된 바 없다.

홉필드 넷 알고리즘에 의해 이용자의 부정확한 탐색문이 의미망으로 표현된 지식베이스상에서 구체화될 수 있다. 즉 초기 탐색문이 지식베이스상의 특정 노드와 연관되면, 홉필드 넷 알고리즘에 의해 이웃 노드로 활성화되며 활성화된 노드는 초기 탐색어들과 연결된 링크의 가중치들을 반영한다. 또한 변형함수(SIGMOID 함수, f_s)를 적용하여 새롭게 활성화될 노드를 결정한다. 이 과정은 출력 노드(node outputs)가 다음의 반복탐색에서 더 이상 변화가 없을 때 멈추게 된다. 홉필드 넷 알고리즘에 의한 개념확장이 성공적으로 완료된 후 출력 노드는 초기 탐색문을 가장 잘 표현하는 용어집합이 될 것이다.

홉필드 넷 알고리즘에 대해 구체적으로 살펴보면 다음과 같다.

(1) 이용자가 입력한 탐색문을 초기화한다. 초기 탐색어 집합 {S1, S2, ... Sm}은 이용자에 의해 입력되며, 초기 탐색어와 일치되는 의미망상의 각 노드는 1의 가중치를 갖도록 초기화된다. 또한 이용자는 bnb 탐색에서와 같이 최대 확장 탐색어 수(p)를 제공해야 한다.

$$\mu_i(0) = x_i,\ 0 \le i \le n-1$$

$\mu_i(t)$는 t번 반복한 후의 노드 i의 가중치이며 0과 1사이의 값을 갖는다. 모든 초기 탐색어에 할당된 가중치는 1이다.

(2) 초기 탐색문은 활성화되어 초기 탐색어들과 연관된 이웃 노드들의 가중치가 다음 공식에 의해 산출된다. t_{ij}는 노드 i에서 노드 j까지의 "접합(synaptic)" 가중치 즉, 링크 가중치를 표현한다.

$$\mu_j(t+1) = f_s\left[\sum_{i=0}^{n-1} t_{ij}\,\mu_i(t)\right] \quad 0 \le j \le n-1$$

위 식에서 f_s는 아래 식에서 보여지는 바와 같이 SIGMOID 변형함수이다(Knight 1990; Dalton, & Deshmane 1991).

$$f_s(net_j) = \frac{1}{1+exp\left[\frac{-(net_j-\theta_j)}{\theta_0}\right]}$$

net_j는 $\Sigma_{i=0}^{n-1} t_{ij}\,\mu_i(t)$이고, θ_j는 기준치이며 θ_0는 SIGMOID 함수의 형태를 수정하는데 사용된다. 이 공식은 홉필드 넷 알고리즘의 병렬 완화적 속성을 보여 주고 있다. 각각의 반복단계에서 모든 노드는 동시에 활성화된다. 가중치 계산식인 $\Sigma_{i=0}^{n-1} t_{ij}\,\mu_i(t)$ 또한 홉필드 넷 알고리즘의 독특한 특성이다. 병렬 활성화(parallel activation)를 기반으로 새롭게 활성화된 노드의 가중치는 이웃 노드와 이웃 노드들의 접합 가중치의 합을 기반으로 하여 새롭게 산출될 수 있다.

(3) 위 과정은 두 번의 반복 과정간에 산출되는 용어의 수에 변화가 없을 때까지 반복된다. 이것은 다음 식에 의해 확인될 수 있다.

$$\sum_{j=0}^{n-1} | \mu_j(t+1) - \mu_j(t) | \leq \varepsilon$$

위 공식에서 ε은 허용할 수 있는 최대 오류값이다. 최종 탐색문은 초기 탐색어들에 대해 적합한 용어집합을 포함한다. 마지막으로 활성화된 노드의 수가 이용자가 기대하는 최종 탐색어의 수인 p보다 많을 경우, 이 중지 조건 알고리즘은 마지막으로 활성화된 노드들 가운데 상위 p개의 용어를 선정할 수 있게 한다.

(4) 활성화된 노드의 수가 p보다 적으면, 시스템은 보다 많은 활성화 노드들을 발견해 내기 위해 위 과정을 반복한다.

3 개념기반 검색모형의 실험 설계

3.1 실험의 개요

본 연구에서 개발될 개념기반 정보검색시스템은 크게 두 부분으로 구성되어 있다. 검색대상이 되는 지식베이스를 의미망 구조로 구축하는 부분과 이 지식베이스를 대상으로 개념기반 정보검색을 수행하는 부분이다. 본 실험에서의 기본적인 지식베이스는 문헌 데이터베이스로부터 자동으로 구축된 문헌기반 지식베이스이다.

본 실험은 2단계로 진행되며 1차 실험과 2차 실험으로 구성된다. 각 실험은 다시 개념확장 알고리즘 비교실험과 지식베이스 비교실험으로 구분된다. 1차 실험과 2차 실험의 구성요소를 비교하여 보면 표 1과 같다.

〈표 1〉 1차 실험과 2차 실험의 구성 요소

		1차실험	2차실험
실험문헌유형		정기간행물기사	신문기사
기사 건수		1,023	5,839
탐색문 수		30	30
용어수/링크수	문헌기반 KB	5,505/108 388	19,261/240,842
	시소러스 기반 KB	12,686/23,342	12,686/23,342
	통합형 KB	17,465/131,740	29,968/264,196
	동의어 처리형 KB	10,992/114,864	24,215/247,322

개념확장 알고리즘 비교 실험에서는 순차적 bnb 알고리즘, 병렬적 bnb 알고리즘, 그리고 홉필드 넷 알고리즘의 검색성능을 비교 평가하였다. 즉 각각의 알고리즘을 사용하여 개념확장을 수행하고, 확장된 용어를 가지고 정보검색을

수행한다. 세 개의 개념확장 알고리즘을 비교하기 위해서 검색된 문헌집합을 처음 입력된 질문에 대하여 적합한 문헌과 부적합한 문헌으로 구분하며, 이 때 적합문헌과 부적합문헌의 판정은 처음 질문을 작성한 주제전문가 집단이 수행한다. 이 실험을 위해 사용된 지식베이스는 문헌 데이터베이스로부터 자동으로 구축된 문헌기반 지식베이스이다. 또한 각 개념확장 알고리즘 검색성능은 불논리 검색 기법인 P-norm 검색모형과 비교된다.

지식베이스별 성능비교 실험에서는 다양한 방법으로 구축한 지식베이스를 대상으로 앞의 개념확장 알고리즘 비교 실험에서 가장 높은 성능을 보여준 알고리즘을 적용하여 검색성능을 비교 평가하였다. 첫 번째 지식베이스는 문헌으로부터 구축된 문헌기반 지식베이스이다. 두 번째 지식베이스는 전통적인 시소러스 내의 어의적 관계를 의미값으로 표현하여 구축한 지식베이스이다. 세 번째 지식베이스는 문헌기반 지식베이스와 전통적인 시소러스를 통합하여 구축한 통합형 지식베이스다. 네 번째 지식베이스는 전통적인 시소러스 내에 나타난 동의어 관계만을 문헌기반 지식베이스에 통합시킨 동의어 처리형 지식베이스이다.

3.2 유형별 지식베이스 구축

3.2.1 문헌기반 지식베이스

네 개의 지식베이스 중 기본 지식베이스가 되는 것은 문헌기반 지식베이스로서, 실험 문헌집단의 각 문헌에 출현한 용어를 자동으로 추출하고 추출된 용어들의 가중치를 산출한다. 또한 용어간의 유사도를 분석하여 의미망 구조의 지식베이스를 최종적으로 구축하게 되는데 그 과정을 구체적으로 살펴보면 다음과 같다.

먼저, 통계적 기법에 의해 용어와 용어간의 유사도를 산출하고, 이를 기반으로 의미망 구조의 지식베이스를 구축할 수 있다. 즉, 문헌으로부터 지식베이스를 구축하기 위해 각 문헌으로부터 용어를 추출하고 용어의 가중치를 산출한 다음 용어의 문헌 내 동시출현빈도를 기반으로 유사도를 산출하여 의미망으로 표현한다.

특정 문헌에 출현한 용어의 가중치를 산출하기 위한 공식은 다양하지만, 본 연구에서는 단어빈도와 역문헌 빈도를 각각 최대값으로 나누어 표준화시킨 공식을 사용하였다(Salton, Fox, & Wu 1983).

$$w_{ik} = \frac{tf_{ik}}{\max(tf_{ih})} \times \frac{idf_k}{\max(idf_h)}$$

tf_{ik} = 용어 k가 특정 문헌 i에서 출현한 빈도

$\max(tf_{ih})$ = 특정 문헌에서 가장 높은 출현빈도를 갖는 단어의 빈도

idf_k = 용어 k의 역문헌 빈도

$\max(idf_h)$ = 전체 문헌 데이터베이스에서 가장 높은 출현빈도를 갖는 단어의 역문헌 빈도

한편, 가중치가 부여된 두 용어간의 의미관계를 생성하기 위해서는 용어간의 유사도가 산출되어야 한다. 개념기반 검색을 위한 의미망 구조의 지식베이스를 구축하는데 사용되는 유사계수도 다양하지만, 본 연구에서는 코싸인 유사계수를 사용하여 용어간의 유사도를 산출하였다.

$$W(T_h, T_k) = \frac{\sum_{i=1}^{n} d_{ij} \times d_{ik}}{\sqrt{\sum_{i=1}^{n} d_{ij}^2 \times \sum_{i=1}^{n} d_{ik}^2}}$$

위 공식에서 $W(T_j, T_k)$는 용어 T_j와 용어 T_k간의 유사도 가중치를 나타내고, d_{ij}는 문헌 i에 출현한 용어 T_j의 가중치이며($0 \le d_{ij} \le 1$), d_{ik}는 문헌 i에 출현한 용어 T_k의 가중치이다($0 \le d_{ik} \le 1$).

위와 같이 용어의 추출 및 용어간의 유사도 산출과정을 거친 용어를 기반으로 의미망 구조의 지식베이스를 구축하였다. 의미망 구조의 지식베이스가 구축될 때, 용어간의 링크 가중치가 0.3 미만인 용어들에 대하여는 연결링크를 생성하지 않았는데 그 이유는 0.3 미만의 링크 가중치를 갖는 용어로 개념확장이 이루어질 경우 검색성능이 낮아질 것으로 판단되었기 때문이다.

3.2.2 시소러스기반 지식베이스

본 연구에서 사용된 실험데이터는 경제학 분야 정기간행물 기사의 초록 및 경제신문기사로서 시소러스기반 지식베이스 구축을 위해 한국경제신문사의 〈경제신문 시소러스〉를 활용하였다.

이 시소러스는 용어와 용어간의 관계를 이미 가지고 있기 때문에 의미망 구조로 쉽게 표현될 수 있다. 이 실험에서는 초기 지식베이스를 구축하기 위해 경제신문 시소러스에 나타난 모든 관계에 값을 부여하여 의미망으로 표현하였다. 즉, 관계값이 링크 가중치로 사용되는 것이다. 일반적으로 관계값의 범위는 0부터 1까지이며 용어간의 의미적 관계가 동의어 수준일 때 1의 값을 부여한다. 본 실험에서는 USE/UF에는 가장 높은 관계값 즉, 1의 값을 부여하였으며, BT/NT/RT 관계에는 각각 0.3/0.6/0.1의 값을 부여하였다.

3.2.3 통합형 지식베이스

문헌기반 지식베이스는 시스템에 입력된 문헌을 분석하여 그 문헌들에서 발견된 용어와 용어간의 관계만을 가지고 구축한 지식베이스이다. 이 지식베이스는 주제전문가에 의해 명확하게 정의된 용어간의 관계정보가 반영되지 않고 오로지 통계적으로 산출된 정보 즉, 두 용어의 동시출현빈도만을 기반으로 구축되었기 때문에 다소 부정확한 의미망이 구축될 수 있다.

시소러스와 문헌의 내용을 모두 이용하여 지식베이스를 구축하기 위해 앞에서 구축한 시소러스기반 지식베이스를 초기 지식베이스로 사용하고, 문헌기반 지식베이스를 초기 지식베이스에 통합하여 통합형 지식베이스를 구축하였다. 용어간의 링크 가중치는 초기 지식베이스가 가지는 용어간의 관계값과 문헌으로부터 산출된 용어간의 유사도 값을 합하여 2로 나눈 값이 된다.

3.2.4 동의어 처리형 지식베이스

이 지식베이스는 전통적인 시소러스에 나타난 USE/UF 관계만을 문헌기반 지식베이스에 통합하여 구축한 지식베이스이다. 문헌기반 지식베이스에 동의어 관계에 있는 용어를 적절히 표현하기 위해 USE/UF 관계에 관계값으로 1을 부여하였다 통합형 지식베이스를 구축할 때와 마찬가지로 다음과 같이 세가지 경우가 발생한다. 첫째, 두 용어가 문헌집합에도 나타나고 시소러스에서도 발견되는 경우이다 이 경우 두 용어간의 유사도 값을 무조건 시소러스에 나타난 값 즉, 1로 대체하도록 하였다. 둘째, 시소러스 내의 동의어 관계에 있는 두 용어 중 하나만 문헌기반 지식베이스에서 발견되고 나머지 하나는 발견되지 않는 경우, 이 용어를 지식베이스에 추가하고 용어간의 유사도 값은 1

로 하였다. 셋째, 동의어 관계에 있는 두 용어가 문헌기반 지식베이스에서 발견되지 않는 경우 두 용어가 새로운 연관성을 가지고 지식베이스에 추가되도록 하였다.

4 1차 실험 및 실험결과 분석

4.1 개념확장 알고리즘의 성능 비교

4.1.1 검색 조건의 변화에 따른 성능 분석

본 실험에서는 검색 조건을 최대 확장용어의 수와 확장될 용어의 최저 가중치를 다양하게 조합하여 12가지 조건에서 검색실험을 수행하였다. 즉 최대 확장용어의 수(p)를 8개, 12개, 16개로 변화시키고 확장될 용어의 최저 가중치(W_p)를 0.2, 0.3, 0.4, 0.5로 변화시켜 가면서 검색결과가 어떻게 달라지는지를 실험하였다. 또한 검색문헌 수를 10건, 20건, 그리고 30건으로 하였을 때의 재현율과 정확률을 평가하였다.

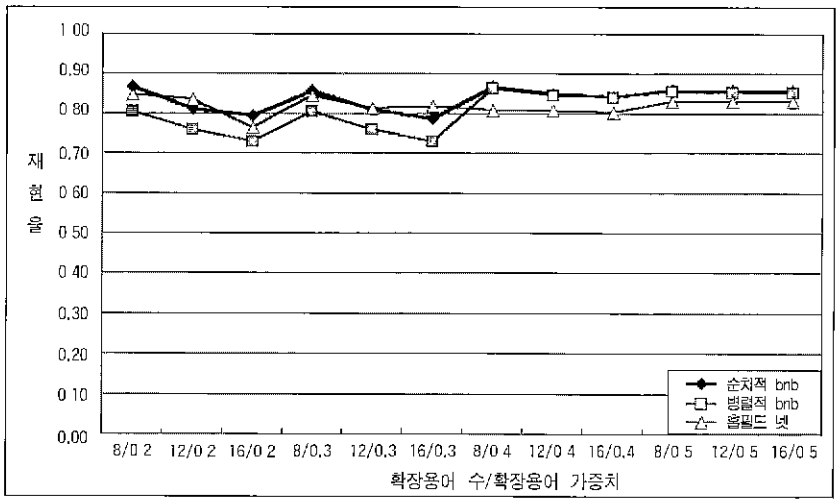
사용된 검색 조건별로 실험결과를 살펴보면 다음과 같다. 탐색문 30개에 대하여 수행된 개념기반 검색결과 중 검색문헌 수를 상위 10건으로 제한하여 검색효율을 측정한 결과, 세 알고리즘간에는 큰 성능차이가 발생하지 않았으나, 홉필드 넷 알고리즘은 다른 2개의 알고리즘에 비해 검색성능이 비교적 높게 나타났다. 홉필드 넷 알고리즘은 특히 확장용어 가중치가 낮은 지점 즉, 0.2와 0.3인 경우에 검색효율이 좋았으며 확장용어 가중치가 0.4 이상인 조건에서는 bnb 알고리즘과 같거나 약간 낮게 나타났다.

검색문헌 수를 20건으로 하였을 때는 순차적 bnb 알고리즘이 가장 높은 검색성능을 보여 주었다. 홉필드 넷 알고리즘은 확장용어 가중치를 0.3 이하로 하였을 때 병렬적 bnb 알고리즘보다 검색성능이 높지만 확장용어 가중치가 0.4 이상인 조건에서는 병렬적 bnb 알고리즘이 더 높은 검색성능을 보여 주었다. 세 알고리즘 모두 0.4 이하 일 때는 확장용어 수를 8개로 하였을 때 높은 검색성능을 보여 주지만 0.5를 넘으면 확장용어 수의 증가에도 불구하고 성능차이가 크게 발생하지 않는 것을 알 수 있다.

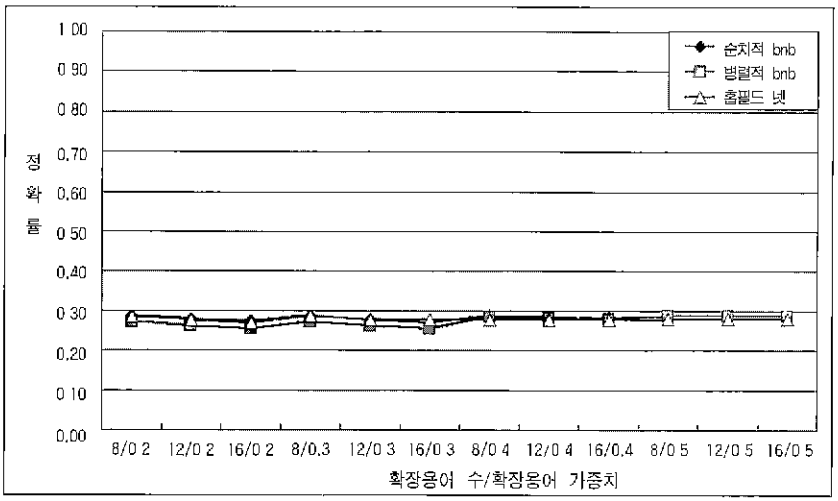
검색문헌 수를 30건으로 하여 검색성능을 측정한 결과 알고리즘별 재현율 차이가 뚜렷해지는 반면 정확률의 성능은 비슷해지는 것을 알 수 있다. 검색문헌 수를 30건으로 하였을 때에도 순차적 bnb 알고리즘의 성능이 가장 높게 나타났으며 확장용어 수가 적을수록 검색성능이 높았다. 병렬적 bnb 알고리즘은 확장용어 가중치가 0.3 이하인 지점에서는 홉필드 넷 알고리즘보다 낮지만 0.4 이상에서는 홉필드 넷 알고리즘보다 검색성능이 높게 나타났다. 그림 1과 2는 검색문헌 수를 30건으로 했을 때의 재현율과 정확률이다.

세 알고리즘의 공통적인 특징은 첫째, 확장용어 가중치가 높을 경우 확장용어 수를 8개에서 16개로 조정하여도 성능이 크게 높아지거나 낮아지지 않는다는 것이다. 둘째, 확장용어 수가 많고 확장용어 가중치가 낮은 경우에 매우 낮은 검색효율을 나타냈다. 세 알고리즘 모두 확장용어 수와 확장용어 가중치를 16개와 0.2로 하였을 때 대체적으로 검색성능이 낮게 나타났다.

표 2는 다양한 개념확장 조건에 의해 산출된 각 값들의 평균을 검색문헌 수별로 비교한 것이다. 검색문헌 수를 10건, 20건, 30건으로 한 모든 경우에 정확률에 대한 평균은 거의 차이



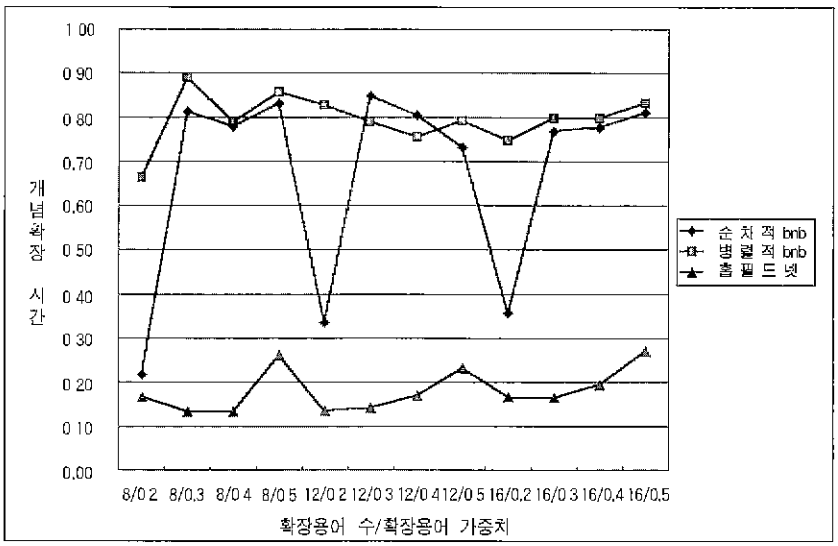
〈그림 1〉 개념확장 알고리즘의 재현율 비교(검색결과 30건)



〈그림 2〉 개념확장 알고리즘의 정확률 비교(검색결과 30건)

〈표 2〉 개념확장 알고리즘의 검색효율 평균 비교

성능 / 검색 문헌수	재현율			정확률		
	순차적 bnb	병렬적 bnb	홉필드 넷	순차적 bnb	병렬적 bnb	홉필드 넷
10건	0.5498	0.5454	0.5620	0.5039	0.4958	0.5047
20건	0.7445	0.7223	0.7300	0.3577	0.3483	0.3509
30건	0.8384	0.8046	0.8185	0.2824	0.2734	0.2783
평균	0.7109	0.6907	0.7035	0.3813	0.3725	0.3779



〈그림 3〉 알고리즘별 개념확장 시간 비교

가 없는 것으로 나타났다. 재현율에 있어서도 큰 차이는 없지만 검색문헌 수를 10건으로 하였을 때에는 홉필드 넷 알고리즘이 다른 알고리즘보다 높은 검색성능을 보여 주며, 검색문헌 수를 20건과 30건으로 올렸을 경우에는 순차적 bnb 알고리즘의 검색성능이 가장 높게 나타났다.

즉, 검색문헌 수를 10건으로 하였을 경우에는 홉필드 넷 알고리즘의 재현율은 0.5620으로 순차적 bnb 알고리즘 및 병렬적 bnb 알고리즘보다 각각 2.2%, 3.0% 높게 나타났다 반면에 검색문헌 수를 20건이나 30건으로 하였을 때에는 순차적 bnb 알고리즘의 재현율이 가장 높게 나타났으며 병렬적 bnb 알고리즘 보다는 각각 3.0%, 3.7%, 그리고 홉필드 넷 알고리즘 보다는 각각 1.9%, 2.4% 높게 나타났다.

4.1.2 개념확장 시간의 비교

본 실험에서는 세 개의 개념확장 알고리즘의 개념확장 시간을 각각 비교 분석하였다. 그림 3은 탐색문 30개에 대하여 개념확장을 하는데 소요된 시간들의 평균으로서 검색문헌 수를 30건으로 하여 확장용어 수와 확장용어 가중치를 조합한 12개의 조건에서 개념확장 시간을 비교한 것이다. 그림에서 보듯이 홉필드 넷 알고리즘은 모든 조건에서 가장 빠르게 개념확장이 이루어진 것을 알 수 있다. 홉필드 넷 알고리즘의 개념확장 시간은 0.1초에서 0.3초이다.

반면에 병렬적 bnb 알고리즘은 0.6초에서 0.9초가 개념확장에 소요되며 개념확장이 가장 느리게 진행되었다. 순차적 bnb 알고리즘은 0.2초에서 0.9초까지 다양한 개념확장 시간을 보여 주고 있다. 즉, 확장용어 가중치가 0.2인 경우에는 비교적 개념확장 속도가 빠르나 0.3 이상에서는 속도가 급격히 느려진다. 이와 같은 개념확장 시간차는 개념확장을 위해 수행된 연산과정의 차이 때문에 발생한 것으로 분석된다.

4.2 개념기반 검색과 P-norm 검색의 성능 비교

P-norm 검색모형은 불논리 검색, 퍼지 집합 검색, 벡터 공간 모형들을 결합하여 일반화시킨 모형으로, 탐색어에 가중치가 부여된 불논리 탐색문과 색인어에 가중치가 부여된 문헌 벡터간의 유사도를 반영하는 일반화된 거리 함수(generalized distance function)에 기반을 두고 있다(Salton, Fox, & Wu 1983).

문헌 D와 질문 Q간의 유사도를 산출하는 P-norm 검색모형은 각 논리 연산자에 따라 다른 공식이 적용되며, 본 연구에서는 실험을 위해 작성된 탐색문의 성격상 각 탐색어가 AND로 조합된 것과 동일하기 때문에 개념 조합에 AND 연산자만을 사용하여 다음 공식을 적용하였다.

$$SIM(Q \text{ AND } p,\ D) = 1 - \sqrt[p]{\frac{a_1^p(1-d_{A_1})^p + a_2^p(1-d_{A_2})^p + \cdots + a_n^p(1-d_{A_n})^p}{a_1^p + a_2^p + \cdots + a_n^p}}$$

위의 공식에서 A_n은 문헌에 출현한 색인어이고 a_n은 탐색문에 출현한 탐색어이다. 본 실험에서는 탐색어에 1의 가중치를 부여하였으며, 색인어 가중치로는 역문헌빈도에 단어빈도를 곱한 3.2.1의 용어가중치 공식을 사용하였다.

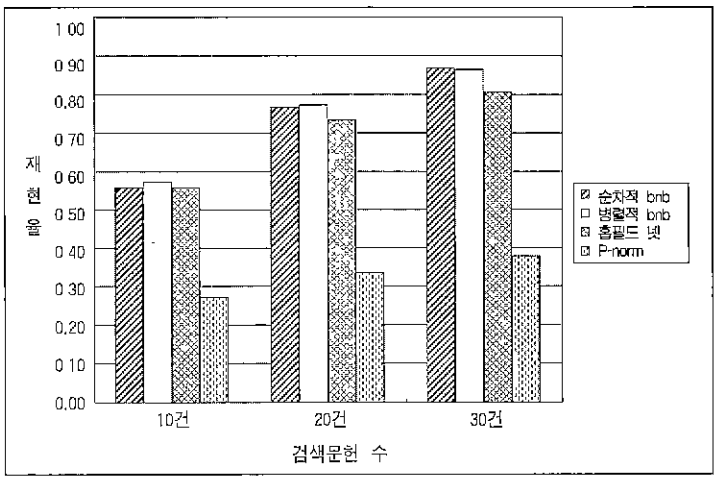
P-norm 검색모형에서 p 파라미터 값은 검색 성능에 영향을 미치는 중요한 요소 중의 하나이다. 설톤 등(Salton, Fox, & Wu 1983)은 모든 연산기호에 동일한 p 값을 적용할 경우, 가중치로 이진값을 사용할 때에 적합한 p 값은 2~5 사이이며 가중치로 실제값을 사용할 때는 p 값은 좀 더 낮은 1~2 사이가 가장 효과가 있다고 밝히고 있다. 본 연구에서는 대부분의 실험에서 비교적 높은 성능을 보이는 것으로 밝혀진 2를 p 값으로 부여하였다.

개념기반 검색모형의 검색성능을 P-norm 검색모형과 비교하기 위해서 위에서 실험한 다양한 조건 중 확장용어 수가 8개이고 확장용어 가중치가 0.4인 조건을 취하였다. 그림 4와 그림 5는 확장용어 수와 확장용어 가중치가 각각 8개와 0.4인 경우의 개념기반 검색을 10건, 20건, 30건에서 P-norm 검색모형과 비교한 것이다.

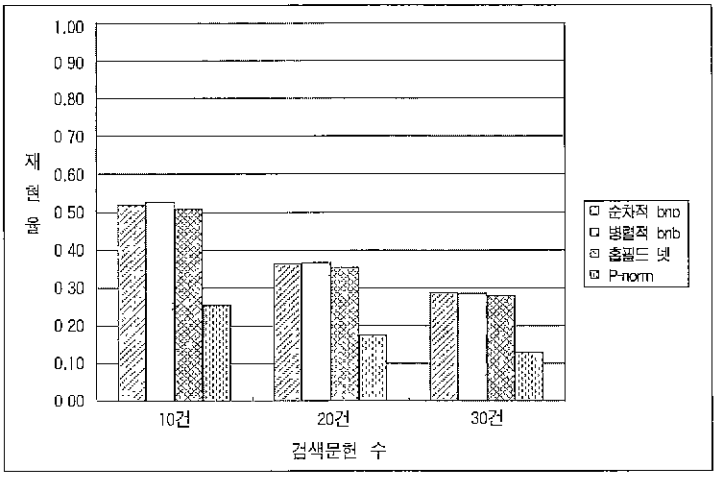
개념확장 알고리즘은 10건, 20건, 30건의 모든 경우에 0.5 이상의 높은 재현율 및 정확률을 보여 주고 있는 반면, P-norm 검색은 재현율과 정확률이 각각 0.27과 0.25로 개념확장 알고리즘이 P-norm 검색보다 거의 200% 정도 높은 성능을 보여 주고 있다.

검색문헌 수를 10건으로 했을 경우만의 재현율을 보면 순차적 bnb, 병렬적 bnb, 그리고 홉필드 넷 알고리즘은 P-norm 검색모형의 0.2705에 비해 각각 206%, 211%, 206% 높게 나타나고 있다. 정확률에 있어서도 P-norm 검색의 0.2533에 비해 각각 204%, 208%, 200% 높게 나타나고 있다.

이와 같이 P-norm 검색모형의 재현율이 개념기반 검색모형의 재현율보다 낮은 이유는 적합문헌이면서도 탐색문의 용어와 정확히 일치하는 색인어를 갖지 않는 문헌이 데이터베이스 내에 다수 존재하지만 P-norm 검색모형은 개념확장을 하지 않음으로써 이러한 문헌들을 검색해 낼 수 없기 때문인 것으로 분석된다.



〈그림 4〉 개념기반 검색모형과 P-norm 검색모형의 재현율 비교



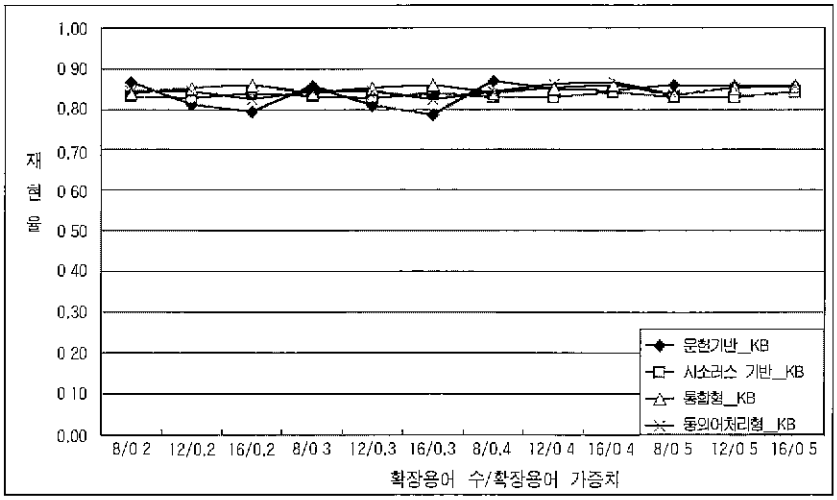
〈그림 5〉 개념기반 검색모형과 P-norm 검색모형의 정확률 비교

4.3 지식베이스 유형별 성능 비교

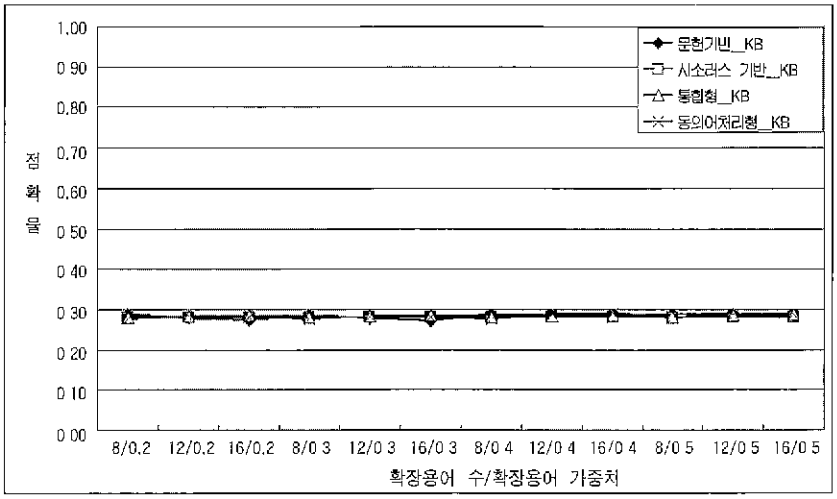
4.3.1 검색 조건의 변화에 따른 성능 비교

다양한 방법으로 구축된 의미망 구조의 지식베이스를 대상으로 한 순차적 bnb 알고리즘에 대한 검색성능 평가도 개념확장 알고리즘 비교실험에서 사용하였던 탐색문 30개에 대하여 수행하였다. 검색 조건을 최대 확장용어의 수(p)와 확장될 용어의 최저 가중치(W_p)를 12가지 경우로 조합하여 검색실험을 수행하였으며, 또한 검색결과 문헌을 상위 10건, 20건, 30건에서 각각 분석하였다.

네 개의 지식베이스에 순차적 bnb 알고리즘을 적용하여 수행된 개념기반 검색결과 중 상위 10건까지의 검색성능을 분석한 결과, 네 개 지식베이스의 정확률은 거의 비슷하였고 재현



〈그림 6〉 지식베이스별 개념확장 알고리즘의 재현율 비교(검색결과 30건)



〈그림 7〉 지식베이스별 개념확장 알고리즘의 정확률 비교(검색결과 30건)

율에 있어서도 큰 차이를 보이지 않았다. 확장용어 수를 8개로 하였을 때에 네 개의 지식베이스는 거의 비슷한 성능을 보여 주고 있으나 확장용어 수를 늘리면 재현율의 차이가 커졌다. 확장용어 가중치가 0.4 이상이 되면 확장용어 수의 변화에도 불구하고 재현율은 거의 비슷해진다. 문헌기반 지식베이스는 낮은 확장용어 가중치에서 확장용어 수의 영향을 가장 많이 받는 것으로 나타났고, 통합형 지식베이스는 확장용어 수의 영향을 거의 받지 않는 것으로 나타났다.

개념기반 검색결과 중 상위 20건까지의 재현율 및 정확률을 분석한 결과, 검색문헌 수를 10건으로 하였을 때와 마찬가지로 네 개 지식베이스의 정확률은 거의 비슷하다는 것을 알 수 있다. 재현율에 있어서는 평균적으로 볼 때

통합형 지식베이스의 성능이 비교적 높지만, 재현율이 가장 낮은 시소러스기반 지식베이스에 비해 3.4% 정도의 매우 적은 차이를 보이고 있다. 문헌기반 지식베이스는 확장용어 수를 많게 할 때보다 적게 하였을 때 재현율이 더 높게 나타났으나 시소러스기반 지식베이스와 통합형 지식베이스는 확장용어 수를 적게 할 때보다 많게 하였을 때 오히려 재현율이 높게 나타났다.

검색문헌 수를 30건으로 하였을 경우에도 정확률은 네 개의 지식베이스간에 거의 차이가 없었고 재현율도 거의 비슷하였다. 문헌기반 지식베이스는 확장용어 수를 8개로 하였을 때 비교적 높은 검색성능을 보여 주었고 확장용어 수를 16개로 하였을 때는 확장용어 가중치의 변화에 따라 다른 세 개의 지식베이스보다 낮거나 비슷한 검색성능을 보여 주었다. 시소러스기반 지식베이스와 통합형 지식베이스는 확장용어 가중치보다 확장용어 수의 영향을 더 많이 받는 것으로 나타났다. 동의어 처리형 지식베이스는 확장용어 가중치가 0.2나 0.3인 경우, 확장용어 수를 8개로 하였을 때가 12개나 16개로 하였을 때보다 재현율이 더 높게 나타났으나 확장용어 가중치를 0.4나 0.5로 하였을 경우에는 확장용어 수를 늘릴수록 성능이 조금씩 향상되었다(그림 6, 그림 7 참조).

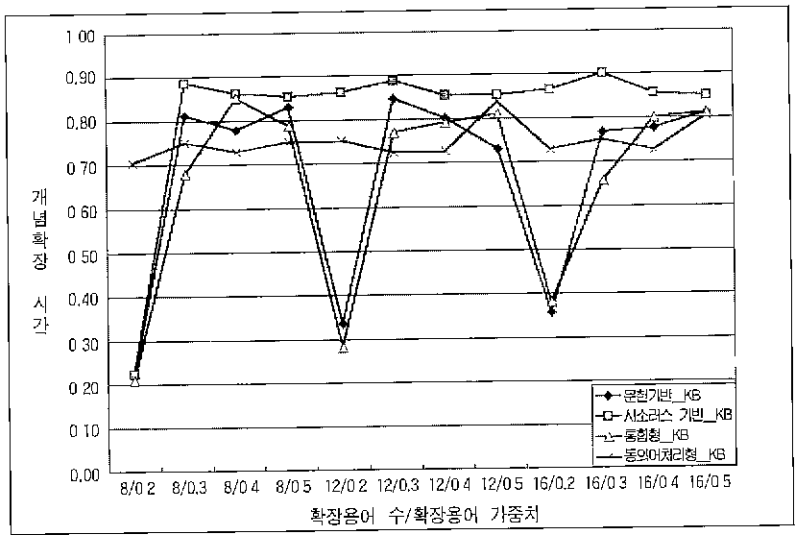
표 3은 다양한 개념확장 조건에 의해 산출된 각 재현율과 정확률의 평균을 검색문헌 수별로 비교한 것이다. 검색문헌 수를 10건, 20건, 30건으로 한 모든 경우에 정확률에 대한 평균은 거의 차이가 없는 것으로 나타났다. 재현율에 있어서는 검색문헌 수를 10건, 20건, 30건으로 한 모든 경우에 통합형 지식베이스의 성능이 약간 더 높게 나타났지만 성능 차이가 크게 발생하지는 않는 것으로 보인다.

검색문헌 수를 10건으로 하였을 때의 재현율을 보면 통합형 지식베이스가 비교적 높게 나타났고 다음으로 시소러스기반 지식베이스, 동의어 처리형 지식베이스순이며, 문헌기반 지식베이스가 가장 낮은 검색성능을 보여 주었다. 검색문헌 수를 20건과 30건으로 한 경우에는 통합형 지식베이스의 재현율과 정확률이 가장 높게 나타났고 다음으로 동의어 처리형 지식베이스, 문헌기반 지식베이스순으로 나타났고 시소러스기반 지식베이스의 성능이 가장 낮게 나타났다.

문헌기반 지식베이스와 시소러스기반 지식베

〈표 3〉 지식베이스별 검색효율 평균 비교

성능 / 검색 문헌수	재현율				정확률			
	문헌 기반 KB	시소러 스기반 KB	통합형 KB	동의어 처리형 KB	문헌 기반 KB	시소러 스기반 KB	통합형 KB	동의어 처리형 KB
10건	0.5498	0.5666	0.5746	0.5614	0.5039	0.4911	0 5038	0.4938
20건	0.7445	0.7316	0.7570	0.7569	0.3577	0.3465	0.3595	0.3584
30건	0.8384	0 8331	0.8493	0.8446	0.2824	0.2805	0.2830	0.2818
평균	0.7109	0.7104	0.7269	0.7209	0.3813	0.3727	0.3821	0.378



〈그림 8〉 지식베이스별 개념확장 시간 비교

이스이 성능이 비교적 낮게 나타난 이유는 문헌기반 지식베이스의 경우 용어의 문헌내 동시출현빈도에 의해 용어간의 유사도를 산출하여 구축되었기 때문에 이 지식베이스를 대상으로 개념확장을 한 경우 의미적으로 관련 없는 용어가 최종 탐색문에 포함되기 때문이며, 시소러스기반 지식베이스의 경우는 이용자의 초기 탐색어가 어의적으로 매우 관련성이 높은 용어로 개념확장이 이루어졌다 할지라도 확장된 용어가 실제 문헌 집단에 출현하지 않는 경우가 많기 때문인 것으로 분석된다.

반면에, 통합형 지식베이스 및 동의어 처리형 지식베이스의 성능이 비교적 높게 나타난 것은 문헌기반 지식베이스를 대상으로 하였을 때 매우 낮은 유사도를 갖는 용어로 확장될 수 있는 가능성을 배제하고 시소러스기반 지식베이스의 통합으로 인하여 어의적으로 관련있는 용어로 확장하게 함으로써 초기 탐색어와 비교적 높은 유사도를 갖는 용어로 개념확장이 이루어지게 하기 때문인 것으로 보인다

4.3.2 개념확장 시간의 비교

본 실험에서는 순차적 bnb 알고리즘을 이용하여 네 개의 지식베이스를 대상으로 개념확장을 하였을 때의 개념확장 시간을 각각 비교 분석하였다. 그림 8은 탐색문 30개에 대하여 개념확장을 하는 데 소요된 시간들의 평균으로서 그림에서 보듯이 시소러스기반 지식베이스를 대상으로 개념확장을 하였을 때 시간이 가장 오래 걸리는 것으로 나타났다. 그 이유는 이 지식베이스를 기반으로 개념확장을 할 경우 확장용어 수와 확장용어 가중치 등 개념확장 조건에 적합한 용어를 찾기 위해 초기 탐색어로부터 먼 단계까지 개념확장이 수행되었기 때문인 것으로 분석된다.

동의어 처리형 지식베이스는 확장용어 수와 확장용어 가중치가 각각 8개와 0.2인 조건에서는 다른 지식베이스보다 개념확장 시간이 길지만 확장용어 가중치가 0.3 이상에서는 개념확장이 비교적 빠르게 진행되었음을 알 수 있다.

한편, 문헌기반 지식베이스와 통합형 지식베

이스는 확장용어 가중치가 0.2인 조건에서는 확장 조건을 만족하는 용어를 비교적 빨리 발견한다는 것을 알 수 있고 확장용어 가중치가 0.3 이상인 조건에서는 확장용어 수의 변화와 무관하게 개념확장에 시간이 오래 걸렸음을 알 수 있다.

5 2차 실험 및 실험 결과 분석

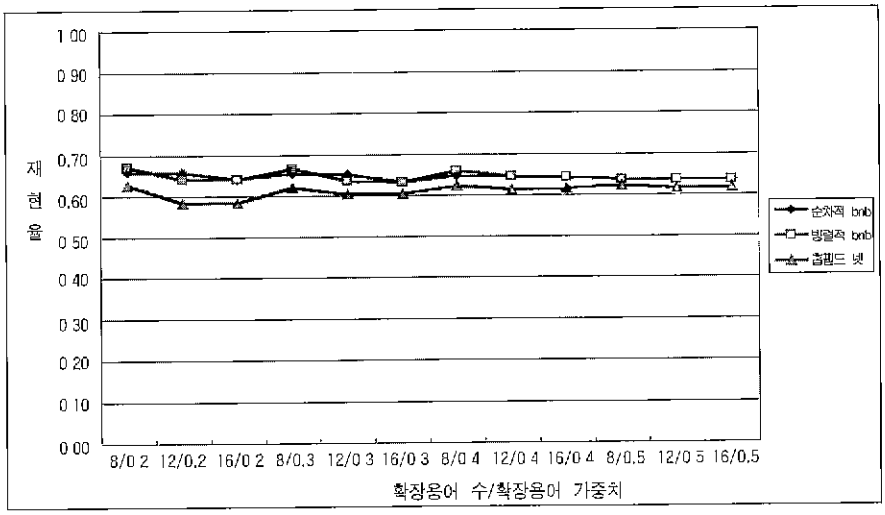
2차 실험도 1차 실험에서와 같이 개념확장 알고리즘 비교실험과 지식베이스 비교실험으로 구분되며 검색 조건을 최대 확장용어의 수와 확장될 용어의 최저 가중치를 다양하게 조합하여 12가지 조건에서 검색실험을 수행하였다. 즉 최대 확장용어의 수(p)를 8개, 12개, 16개로 변화시키고 확장될 용어의 최저 가중치(W_p)를 0.2, 0.3, 0.4, 0.5로 변화시켜 가면서 검색결과가 어떻게 달라지는지를 실험하였다. 또한 검색문헌 수를 10건, 20건, 30건, 40건, 그리고 50건으로 하였을 때의 재현율과 정확률을 평가하였다. 1차실험과 달리 검색문헌 수를 50건까지로 하여 분석한 이유는 실험대상 문헌수가 6,000여건으로 1차 실험보다 약 6배정도 많기 때문이다.

5.1 개념확장 알고리즘의 성능 비교

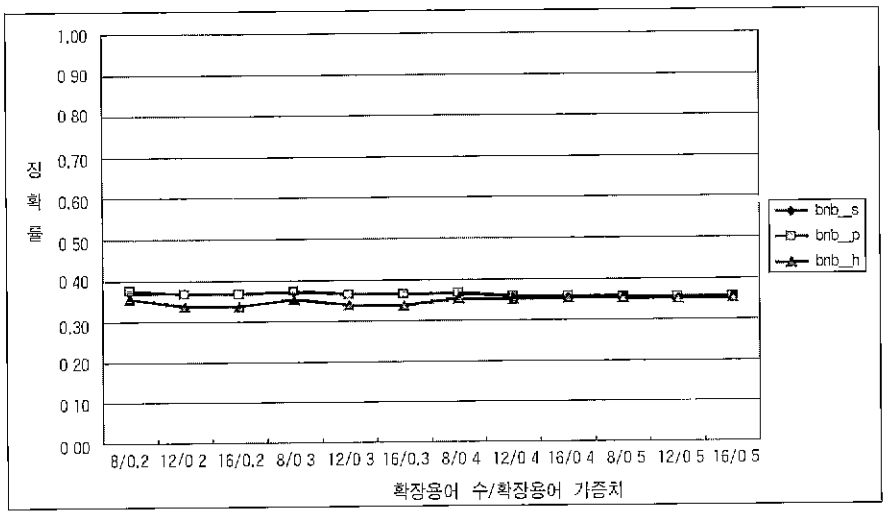
탐색문 30개에 대하여 수행된 검색결과 중 상위 10건까지의 검색효율을 분석한 결과 세 알고리즘간에 큰 성능차이는 발생하지 않는 것으로 나타났다. 병렬적 bnb 알고리즘과 순차적 bnb 알고리즘은 유사한 성능을 보여 주었으나 홉필드 넷 알고리즘은 비교적 낮은 성능을 보여 주었다. 순차적 bnb 알고리즘 및 병렬적 bnb 알고리즘은 낮은 확장용어 가중치에서 비교적 높은 성능을 보여 주었고 홉필드 넷 알고리즘은 변수조정의 영향을 받지 않고 비교적 일정한 성능을 보여 주었다. 세 알고리즘의 공통적인 특징은 높은 확장용어 가중치에서는 확장용어 수의 변화에 영향을 받지 않으며 전반적으로 확장용어 수가 적을수록 높은 성능을 보여 준 점이다.

검색결과 중 상위 20건까지의 성능비교에서도 큰 성능차이가 발생하지 않았다. 평균적으로 보았을 때 병렬적 bnb, 순차적 bnb, 홉필드 넷 알고리즘순의 성능차이를 보이고 있는데 1차 실험에서 순차적 bnb 알고리즘이 병렬적 bnb 알고리즘보다 높은 성능을 보여 주고 있는 점과는 차이를 보이고 있다. 순차적 bnb 알고리즘은 확장용어 가중치가 높은 지점에서 비교적 높은 성능을 보이고 병렬적 bnb 알고리즘은 확장용어 가중치가 0.4인 지점에서 가장 높은 성능을 보여 주었으며, 대체적으로 확장용어 수가 적을수록 높은 성능을 보여 주었다. 홉필드 넷 알고리즘은 확장용어 가중치가 높고 확장용어 수가 적을수록 높은 성능을 보여 주었다. 세 알고리즘의 공통적인 특징은 확장용어 가중치가 0.5로 되면 확장용어 수의 변화에도 불구하고 성능차이가 거의 발생하지 않는다는 점이다.

검색결과 중 상위 30건까지의 성능을 비교한 결과 홉필드 넷 알고리즘이 가장 낮은 성능을 보여 주었고 병렬적 bnb 알고리즘과 순차적 bnb 알고리즘은 유사한 성능을 나타냈으며, 확장용어 수와 확장용어 가중치가 각각 8과 0.4인 지점을 기점으로 해서 성능이 향상됨을 알 수 있다. 홉필드 넷 알고리즘은 낮은 가중치에서 낮은 성능을 나타냈고 다른 조건에서는 성능이 비교적 안정적이었다.



〈그림 9〉 개념확장 알고리즘의 재현율 비교(검색결과 50건)



〈그림 10〉 개념확장 알고리즘의 정확률 비교(검색결과 50건)

검색결과 중 상위 40건까지의 검색성능을 분석한 결과 홉필드 넷 알고리즘은 가장 낮은 성능을 보이고 순차적 bnb 알고리즘과 병렬적 bnb 알고리즘은 유사한 성능을 보여 주었다. 특히 병렬적 bnb 알고리즘은 확장용어 가중치가 0.4인 조건에서 가장 높은 성능을 보여 주었고 홉필드 넷 알고리즘은 낮은 가중치에서 낮은 성능을 나타냈다.

검색결과 중 상위 50건까지의 성능을 분석한 결과 재현율에 있어서 알고리즘간의 성능차이가 뚜렷해지는 것으로 나타났다. 순차적 bnb 알고리즘과 병렬적 bnb 알고리즘의 성능은 거의 비슷하지만 홉필드 넷 알고리즘은 가장 낮은 성능을 보여 주었다 세 알고리즘의 공통적

〈표 4〉 개념확장 알고리즘의 검색효율 평균 비교

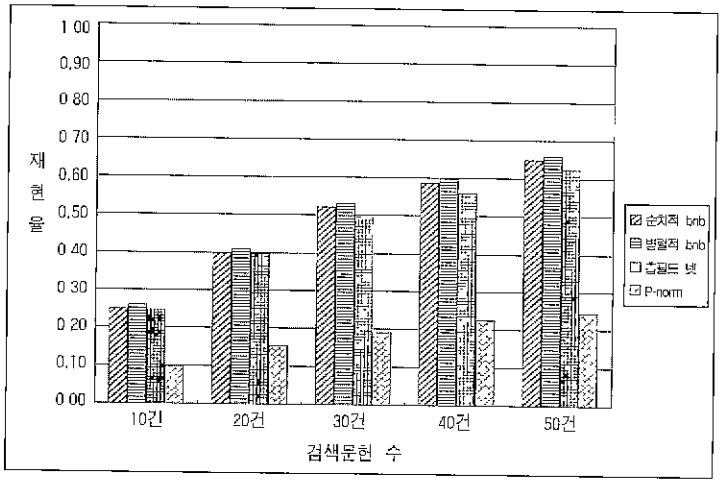
성능 / 검색 문헌수	재현율			정확률		
	순차적 bnb	병렬적 bnb	홉필드 넷	순차적 bnb	병렬적 bnb	홉필드 넷
10건	0.2501	0.2527	0.2410	0.5947	0.5977	0.5772
20건	0.3969	0.4005	0.3895	0.5104	0.5134	0 5032
30건	0.5091	0.5092	0.4879	0 4521	0.4522	0.4398
40건	0.5825	0 5829	0.5481	0.4006	0 4010	0.3794
50건	0.6459	0.6456	0.6112	0.3635	0.3637	0.3481
평균	0.4769	0.4781	0.4555	0 4642	0.4656	0.4495

인 특징은 확장용어 가중치가 동일한 경우 확장용어 수가 적을수록 높은 성능을 보여 주지만 확장용어 가중치가 0.5 이상이 되면 확장용어 수의 변화에도 불구하고 성능차이가 발생하지 않는다는 점이다(그림 9, 그림 10 참조).

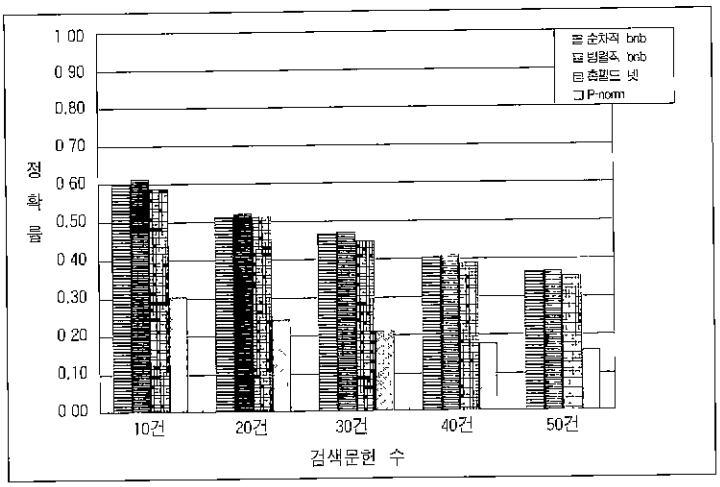
2차 실험을 종합적으로 평가해 보면 첫째, 병렬적 bnb, 순차적 bnb, 홉필드 넷 알고리즘 순의 성능차이가 발생한다. 둘째, 병렬적 bnb 알고리즘은 순차적 bnb 알고리즘과 거의 비슷하였으며 단지 재현율 및 정확률에 있어서 순차적 bnb 알고리즘이 병렬적 bnb 알고리즘보다 각각 0.25%, 0.30%정도 높게 나타났다. 셋째, 병렬적 bnb 알고리즘은 홉필드 넷 알고리즘과는 차이를 보이고 있으며 병렬적 bnb 알고리즘이 홉필드 넷 알고리즘보다 재현율과 정확률에 있어서 각각 4 9%, 3 6% 씩 높게 나타났다(표 4 참조).

5.2 개념기반 검색과 P-norm 검색의 성능 비교

1차 실험에서와 같이 개념기반 검색결과를



〈그림 11〉 개념기반 검색모형과 P-norm 검색모형의 재현율 비교



〈그림 12〉 개념기반 검색모형과 P-norm 검색모형의 정확률 비교

P-norm 검색모형과 비교분석하였으며, P-norm 검색을 위한 실험환경은 1차 실험과 동일하다. 비교 실험 조건은 확장용어 수가 8개이고 확장용어 가중치가 0.4인 조건에서 검색문헌 수가 각각 10건, 20건, 30건, 40건, 50건인 지점을 모두 비교하였다(그림 11, 그림 12 참조). 실험결과, 세 개의 알고리즘 모두 P-norm 검색모형보다 약 200% 정도 높은 성능을 보여 주었다. 예를 들어, 검색문헌 수를 30건으로 했을 경우를 보면, P-norm은 순차적 bnb 알고리즘보다 재현율과 정확률이 각각 216%, 158% 낮고, 병렬적 bnb 알고리즘보다 224%, 162% 낮으며, 홉필드 넷 알고리즘보다는 206%, 152% 낮게 나타났다.

5.3 지식베이스 유형별 성능 비교

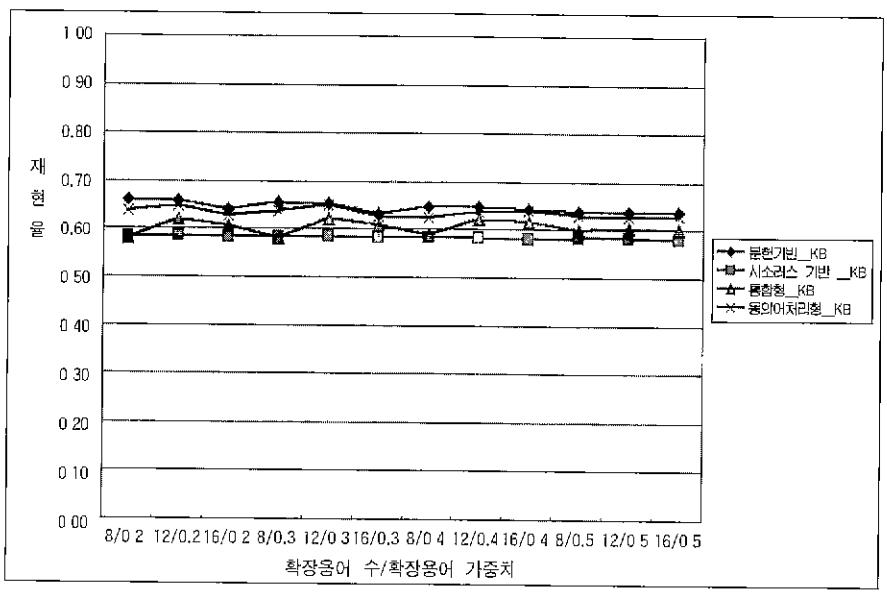
다양한 방법으로 구축된 지식베이스를 비교하기 위한 2차 실험에서의 비교실험조건도 1차 실험과 동일하다. 먼저, 검색결과 중 상위 10건까지의 재현율 및 정확률을 비교한 결과, 문헌기반 지식베이스가 가장 높은 성능을 보여 주었고 다음으로 동의어 처리형, 통합형, 시소러스기반 지식베이스순의 성능차이를 보여 주었다. 문헌기반 지식베이스는 재현율과 정확률에 있어 각각 시소러스기반 지식베이스보다 20.3%, 17.7% 정도 높게 나타났고, 통합형 지식베이스보다는 각각 14.5%, 14.9%, 동의어 처리형 지식베이스보다는 각각 7.1%, 6.4% 높게 나타났다. 문헌기반 지식베이스는 확장용어 가중치가 0.2와 0.3인 조건에서 비교적 높은 성능을 보이고 확장용어 수가 적을수록 높은 성능을 보여 주었다. 시소러스기반 지식베이스는 확장용어 가중치가 0.2와 0.3인 조건에서 비교적 높은 성능을 보이고 확장용어 수가 많을수록 높은 성능을 보여 주었다. 통합형 지식베이스는 확장용어 가중치가 0.4인 조건에서 가장 높은 성능을 보여 주었고, 동의어 처리형 지식베이스는 확장용어 가중치가 낮고 확장용어 수가 많을수록 높은 성능을 보여 주었다. 위 네 지식베이스의 공통적인 특징은 확장용어 가중치가 0.5 이상이 되면 확장용어 수

의 변화에도 불구하고 거의 성능차이가 발생하지 않는다는 것이다.

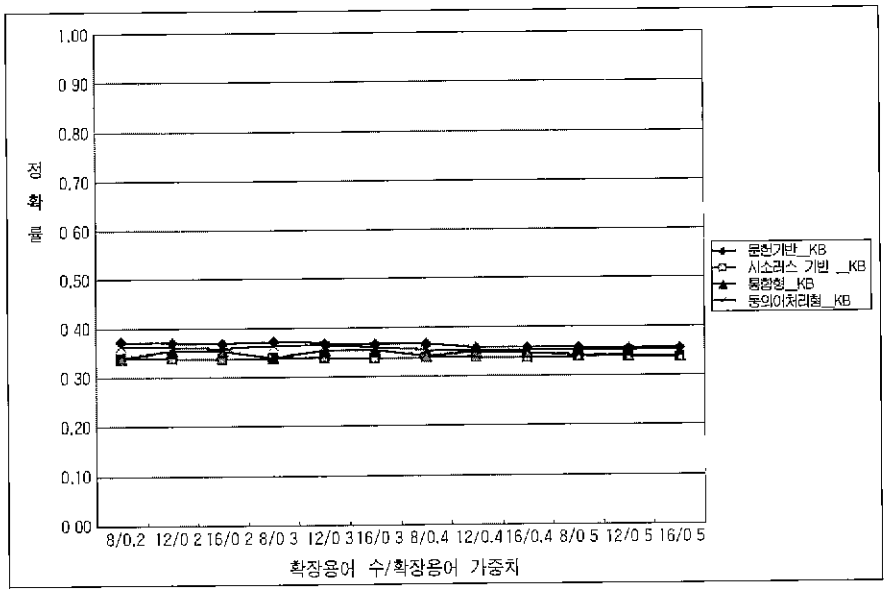
다음으로 검색결과 중 상위 20건까지의 재현율 및 정확률 비교에서 문헌기반 지식베이스가 가장 높은 성능을 보여 주었으며 다음으로 동의어 처리형, 시소러스기반, 통합형 지식베이스 순으로 나타났다. 그러나 재현율과 정확률에 있어 모두 근소한 성능차이가 발생했는데 재현율과 정확률에 있어 문헌기반 지식베이스는 평균적으로 시소러스기반 지식베이스보다 3.4%, 5.6%, 통합형 지식베이스보다 4.9%, 7.8%, 그리고 동의어 처리형 지식베이스보다 3.1%, 3.8% 높게 나타났을 뿐이다. 한편, 문헌기반 지식베이스와 동의어 처리형 지식베이스는 확장용어 가중치가 0.4와 0.5인 조건에서 비교적 높은 성능을 보여 주었고 시소러스기반 지식베이스와 통합형 지식베이스는 확장용어 가중치가 0.2 및 0.3의 낮은 조건에서 비교적 높은 성능을 보여 주었다.

검색결과 중 상위 30건까지의 재현율 및 정확률 비교에서 문헌기반 지식베이스가 가장 높은 성능을 보여 주었고 다음으로 동의어 처리형, 통합형, 시소러스기반 지식베이스순으로 나타났다. 문헌기반 지식베이스는 재현율과 정확률에 있어 시소러스기반 지식베이스보다 평균적으로 각각 10.9%, 8.7%, 그리고 통합형 지식베이스보다 8.9%, 7.4%, 동의어 처리형 지식베이스보다 3.0%씩 높게 나타났다.

검색결과 중 상위 40건까지의 재현율 및 정확률을 비교하였을 때에도 문헌기반 지식베이스가 가장 높은 성능을 보여 주었고 다음으로 동의어 처리형, 통합형, 시소러스기반 지식베이스순으로 나타났다. 재현율에 있어서는 비교적 뚜렷한 성능차이를 보여준 반면 정확률은 약간의 성능차이가 발생했다. 즉, 문헌기반 지식베이스는 평균적으로 시소러스기반 지식베이스보다 10.6%, 통합형 지식베이스보다 7.4%, 동의어처리형 지식베이스보다 2.9% 높게 나타났



〈그림 13〉 지식베이스별 개념확장 알고리즘의 재현율 비교(검색결과 50건)



〈그림 14〉 지식베이스별 개념확장 알고리즘의 정확률 비교(검색결과 50건)

으며, 정확률은 평균적으로 시소러스기반 지식베이스보다 9.6%, 통합형 지식베이스보다 6.0%, 동의어 처리형 지식베이스보다 3.5% 높게 나타났다.

검색결과 중 상위 50건까지의 재현율 및 정확률 비교에서도 문헌기반 지식베이스가 가장 높은 성능을 보여 주었다. 그러나 변수조정에 의한 규칙성을 발견하기는 힘들었고, 다만, 시소러스기반 지식베이스는 낮은 확장용어 가중치에서 비교적 높은 성능을 보여 주었으며, 다른 지식베이스는 높은 확장용어 가중치에서 비교적 높은 성능을 보여 주었다(그림 13, 그림 14 참조).

지식베이스의 성능을 종합적으로 평가해 보

〈표 5〉 지식베이스별 검색효율 평균 비교

성능 / 검색 문헌수	재현율				정확률			
	문헌 기반 KB	시소러 스기반 KB	통합형 KB	동의어 처리형 KB	문헌 기반 KB	시소러 스기반 KB	통합형 KB	동의어 처리형 KB
10건	0.2501	0.2112	0.2184	0.2334	0.5947	0.5072	0.5175	0.5591
20건	0.3969	0.3837	0.3782	0.3848	0.5104	0.4833	0.4736	0.4919
30건	0.5091	0.4592	0.4706	0.4944	0.4521	0.4161	0.4208	0.4391
40건	0.5825	0.5266	0.5425	0.5659	0.4006	0.3656	0.3779	0.3872
50건	0.6459	0.5825	0.6035	0.6344	0.3635	0.3377	0.3451	0.3563
평균	0.4769	0.4326	0.4426	0.4625	0.4642	0.4219	0.4269	0.4467

면 지식베이스의 성능은 문헌기반, 동의어 처리형, 통합형, 시소러스기반 지식베이스순으로 나타났다. 표 5에서 보듯이 문헌기반 지식베이스의 재현율과 정확률은 평균적으로 시소러스기반 지식베이스보다 10.2%, 10.0%, 통합형 지식베이스보다는 7 7%, 8.7%, 그리고 동의어 처리형 지식베이스보다는 3.1%, 3.9%씩 높게 나타났다. 문헌기반 지식베이스의 성능을 평가할 때 평가할 검색문헌 수를 10건에서 50건까지 늘릴 경우 재현율은 58.7%, 28.7%, 14.4%, 10.9% 씩 향상되었고, 정확률은 16.5%, 12 9%, 12.9%, 10 2% 씩 낮아졌다.

6 요약 및 결론

개념기반 정보검색은 의미망 구조의 지식베이스를 기반으로 초기 탐색문에 대한 개념확장을 수행한 후 확장된 탐색문을 가지고 최종 탐색을 수행하는 검색 기법이다. 본 논문에서는 다양한 개념확장 검색모형을 실험하여 가장 우수한 개념확장 알고리즘을 발견하고 효율적인 개념확장 검색모형을 제시하고자 하였다. 이를 위해 실험대상 문헌 수를 1,000여건으로 한 1차 실험과 6,000여건으로 한 2차 실험을 하였으며, 각 실험은 다시 개념확장 알고리즘 비교실험과 지식베이스 비교실험으로 구분된다. 또한 개념확장 알고리즘을 통계적 기법인 불논리 검색 기법의 단점을 보완하기 위해 등장한 P-norm 검색모형과도 비교 평가하였다. 먼저, 1차 실험의 결과를 분석해 보면 아래와 같다.

첫째, 탐색문 30개에 대하여 수행된 개념기반 정보검색 실험에서 검색문헌 수를 상위 10건으로 제한하여 검색효율을 측정한 결과, 세 알고리즘의 검색성능이 큰 차이가 없었으나 홉필드 넷 알고리즘은 다른 두 개의 알고리즘에 비해 검색효율이 비교적 높게 나타났다. 홉필드 넷 알고리즘은 특히 확장용어 가중치가 낮은 지점에서 검색효율이 좋았다.

둘째, 검색문헌 수를 20건과 30건으로 하였을 때는 순차적 bnb 알고리즘이 가장 높은 검색성능을 보여 주었다. 홉필드 넷 알고리즘은 확장용어 가중치를 0.3 이하로 하였을 때 병렬적 bnb 알고리즘보다 검색성능이 높지만 확장용어 가중치가 0.4 이상인 경우에는 병렬적 bnb 알고리즘이 더 높은 검색성능을 보여 주었다.

셋째, 개념기반 검색모형은 P-norm 검색모형보다 재현율과 정확률에 있어 거의 200% 정도 높은 검색성능을 보여 주었다.

넷째, 지식베이스 비교실험에서 검색문헌 수를 10건, 20건, 30건으로 제한하여 실험한 결과, 네 개 지식베이스의 정확률은 거의 차이가 없었으며 재현율에 있어서도 큰 차이가 없었다. 다만, 평균으로 보았을 때 통합형 지식베이스가 다른 지식베이스에 비해 재현율에 있어서 상대적으로 높게 나타났다. 검색문헌 수를 10건으로 하였을 때는 문헌기반 지식베이스의 재현율이 가장 낮게 나타났고, 검색문헌 수를 20건과 30건으로 하였을 때에는 시소러스기반 지식베이스의 재현율이 가장 낮게 나타났다.

실험대상 문헌의 수가 6,000여건인 2차 실험의 결과는 1차 실험의 결과와 약간의 차이를 보이고 있는 것으로 나타났다. 먼저, 알고리즘 비교실험의 경우를 보면 각 실험에서 가장 높은 성능을 보여준 알고리즘이 1차 실험에서는 순차적 bnb 알고리즘인 반면 2차 실험에서는 병렬적 bnb 알고리즘인 것으로 나타났다. 또한 가장 높은 성능을 보여준 조건이 1차 실험에서

는 확장용어 가중치가 0.4인 지점인 반면 2차 실험에서는 확장용어 가중치가 낮은 조건에서도 비교적 높은 성능을 보여 주었다.

다음으로 지식베이스 비교실험에서는 검색성능의 순위에 있어서 변화가 있었다. 즉, 1차 실험에서는 검색성능의 순위가 통합형, 동의어 처리형, 문헌기반, 시소러스기반 지식베이스순이었으며 성능차이가 크게 발생하지는 않았다. 반면, 2차 실험에서의 성능은 문헌기반, 동의어 처리형, 통합형, 시소러스기반 지식베이스순으로 나타났다. 특히 지식베이스간 성능차이 분석에서 나타난 현상은 1차 실험에서는 지식베이스간 성능차이가 뚜렷하지 않고 파라미터 값에 의해 성능차이가 발생하는 반면, 2차 실험에서는 지식베이스간 성능차이가 뚜렷하며 파라미터 값에 의해 많은 영향을 받지 않는 것으로 나타났다.

두 실험에서 동일하게 나타난 현상은 P-norm 검색과의 비교에서 개념확장 알고리즘의 성능이 P-norm 검색보다 높은 성능을 보여 주었다는 점이다. 1차 실험에서는 개념확장 알고리즘이 P-norm 검색보다 재현율과 정확률 모두 약 200% 정도 높은 성능을 보여 주었고 2차 실험에서는 재현율과 정확률 각각 약 200%, 160%정도 높은 성능을 보여 주었다. 지식베이스 비교실험에서 동일하게 나타난 현상은 두 실험 모두에서 시소러스기반 지식베이스의 성능이 가장 낮게 나타났다는 것이다.

본 연구의 1차 실험과 2차 실험을 통해 발견한 사실은 다음과 같다. 첫째, 순차적 bnb 알고리즘과 확장 방식을 달리한 새로운 방식의 병렬적 bnb 알고리즘을 실험한 결과 병렬적 bnb 알고리즘이 순차적 bnb 알고리즘과 유사한 성능을 보여 준다는 점이다. 둘째, 본래 신경망 구조의 지식베이스에 적용되어 왔던 홉필드 넷 알고리즘을 의미망 구조의 지식베이스에 적용한 결과 홉필드 넷 알고리즘은 다른 bnb 알고리즘과 유사한 검색성능을 보여 주었다. 셋째, 1차 실험 후 시소러스기반 지식베이스의 성능은 지식베이스의 크기가 커지면 다른 알고리즘보다 높아질 것이라고 예측하였으나 문헌 데이터베이스의 크기를 6배로 하였음에도 불구하고 여전히 낮은 성능을 보여 주었다. 오히려 다양한 조건 변화에도 불구하고 모든 조건에서 다른 지식베이스보다 낮은 성능을 보여 주었고 따라서 시소러스가 개발된 분야이건 개발되지 않은 분야이건 문헌기반 지식베이스의 활용만으로도 높은 검색성능을 기대할 수 있음을 알 수 있다.

참 고 문 헌

신은자. 1998. 『피드백 정보를 이용한 불논리 검색 시스템의 성능 증진에 관한 실험적 연구』. 박사학위논문, 연세대학교 대학원, 문헌정보학과.

정영미 외. 1993. 『키워드 색인에 있어서의 한글 색인어의 선정을 위한 연구: 신문기사 색인 및 검색을 위한 시소러스 구성을 중심으로』. 서울: 한국경제신문사.

한국경제신문사. 1993. 『경제신문시소러스』. 서울: 한국경제신문사.

Bookstein, A., and D. R. Swanson. 1975. "Probabilistic models for automatic indexing". *Journal of the American Society for Information Science*, 26(1): 45-50.

Chen, H., and V. Dhar. 1991. "Cognitive process as a basis for intelligent retrieval systems design". *Information Processing & Management*, 27(5): 405-432.

Chen, H., K. J. Lynch, K. Basu, and T. D. Ng. 1993. "Generating, integrating, and activating thesauri for concept-based document retrieval". *IEEE EXPERT, Special Series on Artificial Intelligence in Text-based Information Systems*, 8(2): 25-34.

Chen, H., P. Hau, R. Orwig, L. Hoopes, and J. F. Nunamaker. 1994. "Automatic concept classification of text from electronic meetings". *Communications of the ACM*, 37(10): 56-73.

Cohen, P. R., and R. Kjeldsen. 1987. "Information retrieval by constrained spreading activation in semantic networks". *Information Processing & Management*, 23(4): 255-268.

Fogel, D. B. 1994. "An introduction to simulated evolutionary optimization". *IEEE Transactions on Neural Networks*, no. 5: 3-14.

Hopfield, J. J. 1982. "Neural network and physical systems with collective computational abilities". *Proceedings of the National Academy of Science*, USA, 78(8): 2554-2558.

Lippmann, R. P. 1987 "An introduction to computing with neural networks". *IEEE Acoustics Speech and Signal Processing Magazine*, 4(2): 4-22.

Lynch, K. J., and H. Chen. 1994. "Knowledge discovery from historical data- an algorithmic approach". [1997. 11. 4]. <http://ai.bpa.arizona.ed/papers/kdhd/kdhd.html>

Maron, M. E., and J. L. Kuhns. 1960. "On relevance, probabilistic indexing and information retrieval". *Journal of the ACM*, 7(3): 216-243.

Quinlan, J. R. 1983. "Learning efficient classification procedures and their application to chess end games". In *Machine Learning, An Artificial Intelligence Approach* Edited by R. S. Michalski, J. G. Carbonell, and T. M. Mitchell. Palo Alto, CA: Tioga Publishing Company, 463-482.

Salton, G., Edward A. Fox, and Harry Wu. 1983. "Extended Boolean information retrieval". *Communications of ACM*, 26(12): 1022-1036.

Shoval, P. 1985. "Principles, procedures and rules in an expert system for information retrieval". *Information Processing & Management*, 21(6): 475-487.

Simon, H. 1991. "Artificial intelligence: where has it been, and where is it going?". *IEEE Transaction on Knowledge and Data Engineering*, 3(2): 128-136.

Simpson, P. K. 1990. *Artificial Neural Systems: Foundations, Paradigms, Applications, and Implementations*. New York: McGraw-Hill Book Company.

Sowa, J. F. 1991. "Panel: current issues in semantic network". In *Principles of Semantic Network: Explorations in the Representation of Knowledge*, Edited J. F. Sowa. San Mateo, CA: Morgan Kauffmann Publishers, Inc., 13-43.

Sparck Jones, K. 1972. "A statistical interpretation of term specificity and its application in retrieval". *Journal of Documentation*, 28(1): 11-21.

Tank, D. W., and J. J. Hopfield. 1987. "Collective computation in neuronlike circuits" *Scientific American*, 257 (6): 104-114

Winston, P. H. 1984. *Artificial Intelligence*. 2nd ed. Reading, MA: Addison-Wesley Publishing Company, Inc.

Wu, H., and G Salton. 1981. "The estimation of term relevance weight using relevance feedback". *Journal of Documentation*, 37(4): 194-214.