개념기반 검색을 위한 시소러스 관계의 효과적 활용방안에 관한 연구

An Study on the Performance of the Concept-Based Information Retrieval Model Using a Relation of Thesaurus

노영희(Young-Hee Noh)*

목 차

초 록

본 연구에서는 인간 전문가에 의해 용어간의 관계가 명확하게 정의된 전통적인 시소러스를 활용함으로써 개념기반 정보검색의 성능을 향상시키고자 하였다. 이를 위해 시소러스를 활용하여 관계값기반, 관계기반, 그리고 통합형 지식베이스를 구축하였다. 관계값기반 지식베이스와 통합형 지식베이스에는 순차적 bnb 알고리즘을 적용하고 관계기반 지식베이스에는 경험적 bnb 알고리즘을 적용하여 지식베이스간 검색성능을 비교분석하였다.

ABSTRACT

This study aims to enhance the performance of concept-based information retrieval through the use of the traditional thesaurus which, clearly defines relations among terms. To achieve this, the study purports to construct relation-value-based, relation-based, and integrated knowledge bases through the use of the thesaurus. To compare and analyze retrieval performance among knowledge bases, two methods were applied. Sequential bnb algorithm is applied to the relation-value-based and integrated knowledge base while heuristic bnb algorithm is applied to the relation-based knowledge base.

키워드 : 개념기반 정보검색, 시소러스, 순차적 bnb 알고리즘, 경험적 bnb 알고리즘

* 이화여대 국제정보센터 실장

■ 논문 접수일 : 2000년 11월 8일

1 서 론

지난 수 십년 동안 다양한 통계적 정보검색기법에 대한 연구가 있었고 각종 통계적 검색기법의 단점을 보완하기 위한 모듈이 연구되고 개발되었다. 그러나 최근 들어 통계적 기법의 단점을 보완할 수 있는 개념기반 검색기법에 대한 연구가 활발하게 진행되기 시작했다. 개념기반 검색은 이용자가 입력한 탐색어가 속해 있는 문헌뿐만 아니라 탐색어와 연관된 색인어가 속한 문헌까지 검색해 주는 기법으로서 기존 시소러스확장 검색기법의 일종이라고 할 수 있다. 그러나 현재 대부분의 시스템이 채택하고 있는 시소러스확장 검색은 초기 탐색어와 연관된 각 관계의 수준을 정해서 확장해 주는 방식을 취하고 있는 경우가 많다.

개념기반 정보검색시스템은 지식베이스와 개념확장 알고리즘으로 구성되어 있으며, 일반적으로 시스템에 입력되는 문헌 데이터베이스로부터 지식베이스를 자동으로 구축하고, 이 지식베이스를 대상으로 개념확장을 수행한 후 문헌 데이터베이스로부터 관련 정보를 검색한다. 개념기반 정보검색을 위한 지식베이스는 보통 문헌에 출현한 용어들의 동시출현빈도를 기반으로 용어들간의 의미 거리를 산출함으로써 구축된다. 의미 거리를 갖는 지식베이스는 의미망으로 표현될 수 있으며, 이용자가 입력한 탐색어에 대해 의미망으로 표현된 지식베이스를 대상으로 개념확장을 함으로써 검색의 효율을 높일 수 있다. 이용자는 자신이 원하는 주제분야의 자료를 검색하기 위해 질문을 표현할 때, 개념을 정확하게 표현하지 못하거나 관련된 용어들을 미리 생각해 내지 못하는 경향이 있기 때문이다.

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 및 병렬적(parallel) 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.

위와 같이 개념기반 검색기법은 개념확장 대상이 되는 지식베이스로서 문헌기반 지식베이스를 사용할 수도 있고 시소러스기반 지식베이스를 사용할 수도 있으며 어느 방법을 선택하든지 검색성능에 큰 차이가 발생하지 않는 것으로 밝혀지고 있다(노영희 1999).

본 연구에서는 인간 전문가에 의해 구축된 시소러스를 보다 더 효과적으로 활용함으로써 개념기반 검색의 성능을 향상시킬 수 있는지를 알아보고자 하였다. 이를 위해 시소러스에 나타난 각 관계에 다양한 값을 부여하여 어떤 관계값을 부여하였을 때 가장 높은 성능을 보여주는지를 비교하였다. 또한 시소러스의 관계를 관계값으로 변환하여 개념확장을 하였을 경우와 관계를 그대로 활용하여 개념확장을 수행하였을 경우에 어떤 방법이 더 높은 성능을 보여주는지를 비교 분석하였다. 시소러스의 관계를 관계값으

로 변환한 경우에는 개념확장 알고리즘으로 순차적 bnb 알고리즘을 적용하였고 시소러스 내의 관계를 기반으로 개념확장을 하는 경우에는 경험적 bnb 알고리즘을 적용하였다. 또한 통합형 지식베이스를 구축하여 위 두 개의 지식베이스와 검색성능을 비교하였는데 통합형 지식베이스는 관계값기반 지식베이스와 문헌으로부터 자동으로 구축된 문헌기반 지식베이스를 통합하여 구축한 지식베이스로서 순차적 bnb 알고리즘을 적용하였다.

2 개념확장 알고리즘

개념기반 정보검색에서 개념확장을 위해 적용되는 알고리즘에는 bnb 알고리즘 및 홉필드넷 알고리즘 등이 있으며 bnb 알고리즘으로 순차적 bnb 알고리즘, 병렬적 bnb 알고리즘, 경험적 bnb 알고리즘 등이 일반적으로 사용되고 있다. 위 알고리즘 중 순차적 bnb 알고리즘은 개념확장이 되는 지식베이스가 의미값으로 표현되어 있을 때 적용되는 알고리즘이고 경험적 bnb 알고리즘은 시소러스 내의 관계들을 이용하여 개념확장을 수행하는 알고리즘이다. 본 연구에서 구축된 지식베이스는 시소러스이지만 시소러스 내의 관계를 관계값으로 변환하여 개념확장을 하는 경우와 시소러스 내의 관계를 그대로 활용하는 경우로 구분하고 있다. 전자의 방법으로 구축된 지식베이스에는 주로 순차적 bnb 및 병렬적 bnb 알고리즘이 적용되고, 후자의 방법으로 구축된 지식베이스에는 경험적 bnb 알고리즘이 적용된다. 순차적 bnb 알고리즘과 병렬적 bnb 알고리즘의 차이는 단지 확장방식의 차이로서, 순차적 bnb 알고리즘은 확장된 개념들 중 가장 높은 순위의 개념에 대해서만 개념확장을 하는 반면, 병렬적 bnb 알고리즘은 확장된 모든 개념에 대해 병렬적으로 개념확장을 수행한다.

2.1 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, and Dhar 1991). 이용자가 제공한 초기 탐색어에는 1의 가중치가 부여되며, 다음으로 이 용어들과 직접적으로 관련이 있는 이웃하는 용어들을 탐색한다. 확장된 용어의 가중치는 이용자가 입력한 용어와의 링크 가중치를 기반으로 산출된다. 특정 기준치까지 용어들을 확장한 후에 확장된 용어와 이용자가 입력한 용어는 용어의 가중치순에 따라 우선순위 대기행렬(priority queue: $Q_{priority}$)에 저장된다.

의미망 구조의 지식베이스기반 bnb 알고리즘은 적절한 이용자 정의 상태에 도달할 때까지 확장 과정을 반복한다. 개념기반 정보검색에 채택된 알고리즘과의 상호작용 과정에서 이용자는 시스템에 적절한 확장용어 수(p)와 확장될 용어의 최저 가중치(W_p)를 제공하도록 요청 받는다. 이용자가 제시한 이 두 개의 변수는 bnb 반복 확장 과정의 중지 조건(stopping condition)으로 작용한다. 초기 탐색어는 처음에 동일한 가중치를 갖기 때문에 모두 활성화되며, 첫 번째 확장 후에 $Q_{priority}$는 내림차순으로 모든 초기 탐색어와 직접적으로 연결된 이웃 노드들을 찾아 그것의 W_p를 산출하게 된다. 다음으로 $Q_{priority}$에서 가장 상위의 용어에 대하여 개념

확장을 하는 과정을 반복한 후, p번째까지의 노드를 구분해서 이용자가 제시한 두 조건을 모두 만족하는 용어만 최종 탐색문으로 선택된다.

이와 같이 이용자가 지정한 기준치는 시스템이 이용자의 초기 탐색어와 유사한 용어를 적어도 p개 생성할 것을 보장한다. 반복이 진행되는 동안 대기행렬에서 보다 높은 가중치를 갖는 용어들은 $Q_{priority}$에서 높은 순위에 놓이게 될 것이다. 시스템이 검색을 중지하는 시점은 출력 대기행렬의 노드가 p개 이상으로 구성되어 있을 때, 또는 $Q_{priority}$에서 가장 높은 순위에 있는 노드의 가중치가 이용자가 제시한 기준치 즉, W_p보다 낮을 때이다. 순차적 bnb 알고리즘의 개념확장 과정을 단계적으로 기술해 보면 다음과 같다.

(1) 링크에 가중치를 부여한다. 의미망의 초기 상태는 노드와 링크로 표현되어 있다. t_{ij}는 노드 i로부터 노드 j까지의 링크 가중치이다.

(2) 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.

(3) 순차적 bnb 알고리즘은 내림차순으로 우선 순위 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.

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

(4) 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 공식에서 $\mu_j(t+1)$는 순차적 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.

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

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

2.2 경험적 bnb 알고리즘

정보검색시스템에서 주요 문제는 하나의 용어와 연관된 다양한 용어들이 출현한 문헌들을 검색해 내지 못한다는 것이다. 따라서 완전일치 기법을 채택하고 있는 정보검색시스템에서 이용자가 탐색어로 입력할 수 있는 어휘는 문헌을 표현하는데 사용되었던 용어로 제한된다. 이용자에게 무제한적으로 어휘를 제공할 수 있는 방법으로 다소 일반적인 것은 이용자가 탐색문을 재형성할 수 있도록 시소러스를 참조할 수 있게 하는 것이다.

시소러스 브라우징 전략은 이용자에게 탐색어의 불확실성을 감소시키는 도구로 사용된다 즉, 이용자가 탐색어를 입력하면 탐색어와 관련된 용어들로 확장하기 위해 전통적인 시소러스에 대해 확장 알고리즘을 적용한다.

전통적인 시소러스를 기반으로 하여 탐색어를 확장해 가는 bnb 알고리즘을 보통 경험적 bnb 알고리즘(heuristic bnb algorithm)이라 한다. 경험적 bnb 알고리즘은 시소러스가 가지고 있는 용어간의 명확한 관계를 참조하여 용어를 확장하여 가기 때문에 시소러스기반 개념확장 정보검색에 사용된다. 이 알고리즘에 대해 구체적으로 살펴보면 아래와 같다.

개념확장을 진행하기 전에 먼저, 용어간의 관계에 대한 값을 정의해야 한다. 첸 등(Chen et. al. 1994)은 용어간의 관계값의 범위를 0에서 10까지로 지정하고 세 가지 유형의 링크 즉, RT/NT/BT에 각각 3/10/1의 관계값을 부여하였다.

첸과 다(Chen, and Dhar 1991)의 또 다른 연구에서는 USE/NT/RT/BT 등의 용어 관계에 각각 3/9/5/1의 가중치를 부여하였다. USE 링크는 두 개의 동의어를 연결하기 때문에 USE 링크를 따라 경로를 확장할 때에는 USE 링크에 의해 확장된 새로운 용어가 확장 전의 가중치를 갖도록 하였다.

시소러스를 기반으로 입력된 탐색어에 대하여 개념확장을 해 나가는 과정을 구체적으로 살펴보면 다음과 같다(Chen, and Dhar 1991).

(1) 경로 대기행렬을 생성한다. 최초의 경로 대기행렬은 이용자가 입력한 탐색어만으로 구성되어 있다. 탐색어들은 이웃하는 노드의 수에 의해 가중치가 부여되며 가중치순으로 배열된다. 즉 특정성이 높은 용어로 먼저 확장한다는 원칙에 따라 가장 적은 수의 이웃 노드를 가진 탐색어가 대기행렬의 가장 위에 놓이게 된다. 최초의 대기행렬이 T_1, T_2, T_3, T_4의 탐색어로 구성되어 있고 각각의 용어가 가지는 이웃 노드의 수가 각각 0, 3, 6, 5 라고 한다면 대기행렬은 다음과 같이 생성된다.

$$Q_{IN} = \{T_1, T_2, T_4, T_3\}$$

T_1의 이웃 노드가 하나도 없기 때문에 더 이상 확장이 이루어지지 않는다. 따라서, T_1이 제일 먼저 Q_{IN}에서 제거되어 Q_{OUT}에 저장된다.

$$Q_{OUT} = \{T_1\}$$

(2) 다음으로 T_2 용어에 대한 개념확장이 이루어지며 확장된 용어에 대한 가중치는 다음과 같은 공식에 의해 산출된다.

$$N_{path} = O_{term} * RW * N_{term}$$

여기에서 N_{path}는 확장될 노드의 가중치이고 O_{term}은 확장되기 전 용어가 가지는 이웃 노드의 수이며, RW(relative weight)는 링크의 상대적

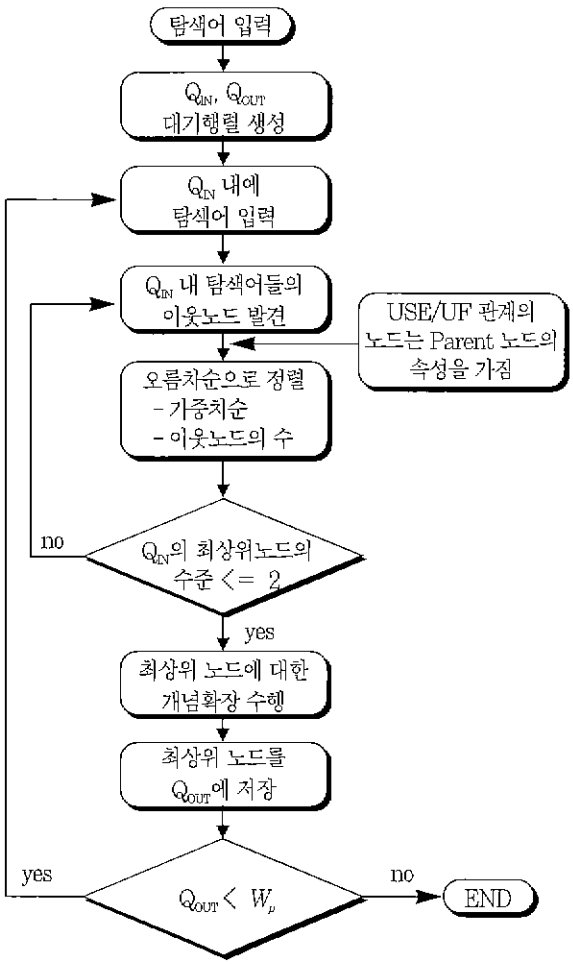
가중치이다. 그리고 N_{patn}은 확장될 용어의 이웃 노드의 수이다.

(3) 대기행렬이 비워지거나 이용자가 요구한 조건을 만족할 때까지 확장을 계속한다. 이용자의 조건을 만족하지 못한 상태라면 Q_{IN}에서 가장 높은 순위에 있는 용어로부터 새로운 용어로 확장한다.

(4) 이용자가 요구한 조건을 만족하면 개념확장을 중지한다. 그리고 Q_{OUT}의 용어들을 용어 가중치의 오름차순으로 정렬한다. 즉 가장 작은 값을 가진 용어가 대기행렬의 가장 위에 놓이게 된다.

시소러스 내의 관계를 기반으로 개념확장을 하는 과정을 순서도로 표현하면 〈그림 1〉과 같다.

위와 같이 경험적 확장 활성화 알고리즘(heuristic expansion activation algorithm)은 관계 정의가 명확한 전통적인 시소러스에 적용하기에 적합한 알고리즘이다. 이용자의 요구를



〈그림 1〉 시소러스 내 관계기반 개념확장 순서도

표현하기에 적합한 용어를 발견하기 위한 경험적 활성화 과정은 다음과 같이 4가지 원칙을 바탕으로 진행될 수 있다(Shoval 1985; Cohen. and Kjeldsen 1987).

첫째, 특정성이 높은 단어로 확장한다. LCSH나 UMLS 등 대부분의 시소러스를 볼 때, 의미망 구조에서 가장 소수의 이웃 노드를 가지는 노드는 보다 많은 이웃 노드를 가지는 노드보다 특정성이 높다는 것을 알 수 있다. 탐색자들은 자신의 정보요구를 다소 광범위하게 기술하는 경향이 있기 때문에, 보다 소수의 이웃 노드를 가진 노드로 먼저 확장한다.

둘째, 특정성이 높은 링크를 따라 확장한다. 링크의 유형에는 NT, BT, RT, USE/UF 관계 등이 있다. 시스템에 따라 다르게 적용될 수 있지만 대부분 USE/UF, NT, RT, BT 순으로 링크를 확장한다. 즉, NT링크를 탐색하기 전에 먼저 USE/UF링크를 탐색하고 RT링크를 탐색하기 전에 NT링크를 탐색하며, BT링크를 탐색하기 전에 RT링크를 탐색하여 용어를 확장한다. 이렇게 함으로써 보다 특정성이 높은 링크를 따라 개념확장을 해 나갈 수 있다.

셋째, 이용자가 입력한 탐색어를 초기 노드라고 할 때, 이 초기 노드에 가까운 노드로 먼저 확장한다. 확장이 진행될 때, 초기 노드로부터 가까운 거리의 노드로 먼저 확장하고 그 다음으로 가까운 노드로 확장해 간다. 초기 노드로부터 가까운 거리에 있는 노드가 먼 거리의 노드보다 더 적합할 것이라는 가정은 합리적이다.

넷째, 확장수준을 2단계로 제한한다. 의미망에서 두 노드간의 링크의 수는 용어간의 의미적 관계를 설명해 준다. 이용자가 입력한 초기 노드와 밀접한 관계에 있는 용어만을 발견해 내기 위해 2단계까지 초기 노드를 확장한다. 즉, 초기 노드로부터 2단계까지의 링크로만 확장한다. 2단계까지로 확장을 제한한 이유는 하나의 노드에 연결된 수백 개의 링크를 추적할 수 없기 때문이며, 또한 탐색자들은 자신의 탐색어들을 2단계 이상 확장하거나 축소시키지 않으려는 특성이 있기 때문이다. 경험적 알고리즘은 초기 노드에 의미적으로 적합한 용어만으로 확장하게 하며 그 이상의 확장은 부적합한 용어를 질문에 포함시킬 수 있다. 하나의 용어가 두 개 이상의 링크에 의해 확장 대상이 되는 경우가 있다. 이 용어는 하나의 링크에 의해 방문되는 용어보다 높은 가중치를 갖게 될 확률 즉, 이용자에게 더 의미 있는 용어일 확률이 높다. 두 개 이상의 링크에 의해 참조되는 용어의 가중치는 참조 횟수나 참조 수준 등을 고려하여 산출할 수 있을 것이다.

2.3 선행 연구

시소러스 브라우징 전략에 의해 검색성능을 향상시키려는 연구는 오래 전부터 있어 왔으며, 주로 시소러스를 활용함으로써 용어의 불일치성을 해결하려한 연구가 주를 이루었다. 그러나 최근에는 이용자가 입력한 탐색어와 시스템에 저장된 색인어와의 불일치성을 해결하고 단순한 용어확장 방법으로 성능을 향상시키는데서 더 나아가 개념확장을 효과적으로 함으로써 이용자의 초기 탐색어와 가장 밀접한 용어들을 최종 탐색문으로 포함시키려는 연구노력이 있어 왔다. 시소러스를 활용하여 이용자의 초기 탐색어를 기반으로 수동 또는 자동으로 개념확장을 하려한 연구가 등장하기 시작되었다.

데이비슨(Davison 1986)은 시소러스를 그래픽형태로 이용자에게 보여줌으로써 검색성능을

약 6% 향상시킬 수 있다는 문헌조사결과를 보여주었다. 그는 시소러스의 그래픽 표현을 위한 이상적인 기준을 약 10가지 정도 제시하기도 하였다. 그는 또한 지금까지 시소러스를 그래픽하게 표현할 수 없었던 것은 기술의 한계나 연구의 부족이었으며 이제는 기술의 발달로 이용자가 편리하게 시소러스를 참조하여 탐색어를 확장할 수 있다고 주장하였다.

폭스(Fox 1987)의 Corder시스템은 'Handbook of Artificial Intelligence'와 'Collin's Dictionary'로부터 구축된 시소러스로 구성되어 있다. CANSEARCH(Pollitt 1987)에서 시소러스는 메뉴로서 제공함으로써 검색성능을 향상시키고자 하였다. 이용자는 그 메뉴를 브라우즈하고 메뉴로부터 질문을 위한 탐색어를 선택한다.

문헌에 대한 검색성능은 시소러스와 시소러스간에 형성된 연관성 정보에 의해 향상되어질 수 있을 것이다. UMLS(National Library of Medical Language System)는 생물의학 관련 용어들과 각 용어간의 관계를 이해하게 함으로써 이용자가 기계가독형 자료로부터 정보를 검색해서 조직할 수 있도록 한 지능형 자동검색시스템이다(Humphreys and Lindberg 1989). 이 시스템은 메타시소러스, 의미망, 그리고 정보자료지도를 포함한다. 메타시소러스는 생물의학 개념에 관한 정보를 포함하고 이 개념을 10개이상의 다른 어휘로 표현하며, 각 언어별 시소러스를 포함한다. 의미망은 메타시소러스 내의 용어의 유형에 관한 정보와 이들 유형간에 존재할 수 있는 관계들에 대한 정보를 포함한다. 또한 정보자원지도는 모든 종류의 생물의학 데이터베이스에 대한 범위, 위치, 어휘, 그리고 접근조건 등에 관한 정보를 포함한다.

첸과 다(Chen and Dhar 1991)는 LCSH (Library of Congress Subject Headings)를 지능형 검색시스템에 통합 설계하였다. 이 시스템은 이용자가 자신의 질문을 명확히 작성하는 것을 돕기 위해 경험적 탐색알고리즘을 채택했다. 그들은 METACAT에 인간의 탐색전략과 LCSH를 서지탐색의 보조로 통합시켰으며, 자동적인 시소러스 참조과정에 경험적 개념확장 알고리즘을 적용하였다.

3 실험설계 및 구현

3.1 실험환경

본 연구에서는 전통적인 시소러스를 개념기반 검색에서 효과적으로 활용할 수 있는 방안을 모색하고자 하였다. 실험을 위해 경제분야 신문기사 5,843건을 실험대상문헌으로 선정하였다. 실험대상이 되는 전통적인 시소러스는 한국경제신문사의 〈경제신문 시소러스〉이다.

시소러스에는 USE, NT, RT, BT 등의 관계가 나타난다. 관계값으로 0에서 1사이의 값을 부여할 때 USE 관계에는 보통 1의 값을 부여하는데 그 이유는 USE관계에 있는 용어는 디스크립터와 동의어이기 때문이다. NT, RT, BT 등에는 학자마다 각각 다른 값을 부여하고 있으나 본 연구에서는 USE관계에 1의 값을 부여하고 BT관계에 0.1의 값을 부여하였으며 NT, RT관계에 다양한 값을 부여함으로써 검색성능을 비교하고자 하였다. 즉 NT의 관계값으로 0.4에서 0.7까지의 값을 부여하고 RT의 관계값으로 0.2에서 0.4까지의 값을 부여한 후 다양하게 조합하여 가장 높은 성능을 보여주는 경우를 발견하

고자 하였다.

또한, 시소러스의 관계에 부여되는 값은 검색대상이 되는 지식베이스의 크기에 따라 차이를 보일 수 있을 것으로 보고 검색대상이 되는 문헌 수를 2,000건, 4,000건, 6,000건으로 다양하게 변형하여 다양한 값의 조합으로 구축된 지식베이스의 성능을 비교 분석하였다.

3.2 지식베이스 구축

본 연구의 실험을 위해 구축한 지식베이스는 문헌기반 지식베이스, 시소러스기반 지식베이스, 그리고 통합형 지식베이스이다. 문헌기반 지식베이스는 문헌으로부터 자동으로 구축된 지식베이스이며, 이 지식베이스를 구축하기 위해서는 먼저, 데이터베이스 내의 모든 문헌으로부터 용어를 추출한다. 각 용어는 통계적 정보검색시스템에서 사용하고 있는 자동색인기법을 사용해서 추출된다. 한국어 형태소 분석기를 거친 이 용어는 불용어, 관사, 조사 등이 제거된 명사들이다. 이 용어들은 문헌 내 출현빈도에 따라 가중치를 갖는데 각 용어의 가중치 산출공식은 tfidf 공식이다. 이 공식은 용어의 문헌 내 출현빈도 및 데이터베이스 내 출현빈도를 동시에 고려한 가중치 산출공식이다. 다음으로 용어 추출과정에서 발견된 용어들이 의미망으로 표현되기 위해서는 용어간의 유사도를 측정해야 한다. 용어간의 유사도를 측정하는 공식으로 본 논문에서는 코사인 유사계수 공식을 사용하고 있다. 코사인 유사계수 공식은 각각의 용어와 다른 모든 용어들과의 동시발생빈도를 기반으로 용어간의 관련도, 즉 유사도 가중치를 산출한다.

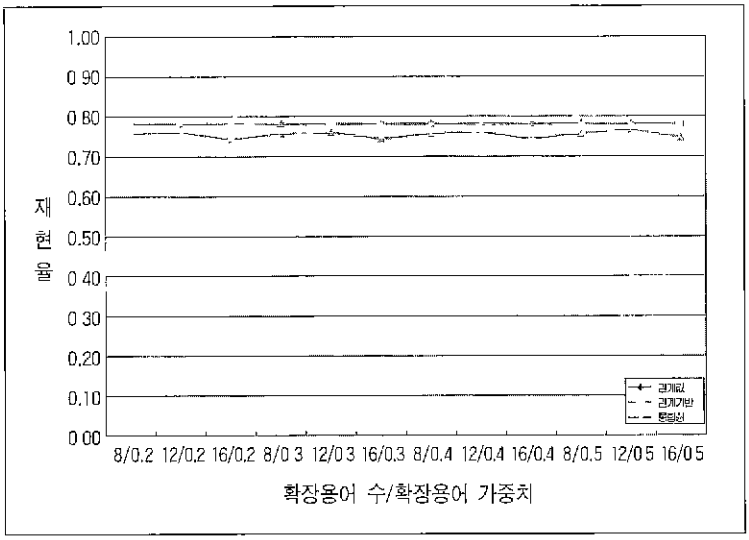
시소러스기반 지식베이스는 두 가지 형태로 재구축하였다. 하나는 전통적인 시소러스 내의 어의적 관계를 적절한 관계값으로 표현하여 구축한 형태로서, 순차적 bnb 알고리즘과 같이 의미값으로 표현된 지식베이스를 대상으로 개념확장을 하는 알고리즘이 적용된다. 다른 하나는 시소러스 내의 관계에 대한 정보를 그대로 가지고 있는 형태로서, 정확한 시소러스 내의 관계를 대상으로 개념확장을 하는 알고리즘이 적용된다. 본 연구에서는 전자의 형태를 관계값기반 지식베이스라 하였고 실험을 위해 순차적 bnb 알고리즘을 적용하였다. 후자의 형태를 관계기반 지식베이스라 하였고 경험적 bnb 알고리즘을 적용하였다.

통합형 지식베이스는 문헌기반 지식베이스와 전통적인 시소러스를 통합하여 구축한 지식베이스로서 여기에서 전통적인 시소러스는 시소러스 내의 관계를 관계값으로 표현하여 구축한 형태로서 순차적 bnb 알고리즘을 적용하여 실험하였다.

4 실험 결과 비교분석

4.1 관계값의 변화에 따른 성능차이

본 실험에서는 시소러스 내에 나타난 각 관계 즉, USE/UF, NT, RT, BT에 다양한 값을 부여하여 가장 높은 성능을 보여 주는 조건을 발견하고자 하였다. 이 관계들 중 USE/UF 관계는 보통 동의어를 연결하기 때문에 관계값으로 1을 부여하였고 BT관계에는 가장 낮은 0.1값을 부여하였다 그리고 NT, RT값만을 다양하게 조합하였는데 NT에는 0.4에서 0.7까지의 값을, RT에는 0.2에서 0.4까지의 값을 부여하여 12가지



〈그림 2〉 50위까지의 재현율 (2,000건) - NT: 0.5, RT: 0.3

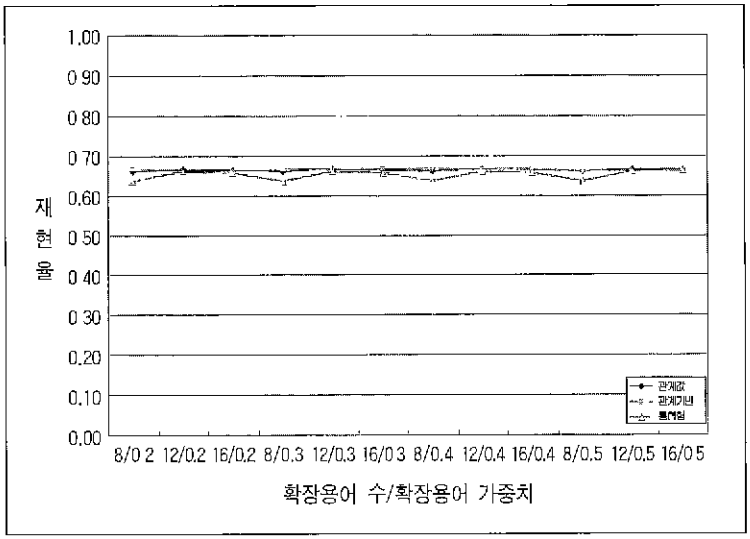
조합조건에서 성능을 측정하였다. 데이터베이스의 크기별로 NT, RT값의 변화에 따른 성능차이를 분석해 보면 다음과 같다.

비교대상 데이터베이스의 크기를 2,000건으로 한 경우, NT값을 0.4와 0.5로 하고 RT값을 0.2에서 0.4로 변화시켜 가면서 성능을 비교한 결과 10위와 20위에서는 통합형, 관계값기반, 관계기반 지식베이스순으로 나타났으나 30위에서 50위까지는 통합형 지식베이스의 성능이 가장 낮게 나타났고 다른 두 개의 지식베이스의 성능은 유사하게 나타났다. 통합형 지식베이스는 확장용어 수가 16개인 경우 낮은 성능을 나타냈고 확장용어 가중치가 0.5인 지점에서 대체적으로 높은 성능을 보여 주었다. 관계값기반 지식베이스는 확장용어 수가 적을 때 높은 성능을 보이고 관계기반 지식베이스는 확장용어 가중치가 낮을 때 낮은 검색성능을 보여 주었다. 〈그림 2〉는 NT, RT값을 각각 0.5와 0.3으로 한 경우의 재현율이다.

NT값을 0.6과 0.7로 하고 RT값을 0.2에서 0.4로 변화시켜 가면서 성능을 비교한 결과, 10위와 20위에서는 검색성능이 통합형, 관계기반, 관계값기반 지식베이스순으로 나타났으나 30위에서 50위까지는 관계기반, 통합형, 관계값기반 지식베이스순으로 나타났다.

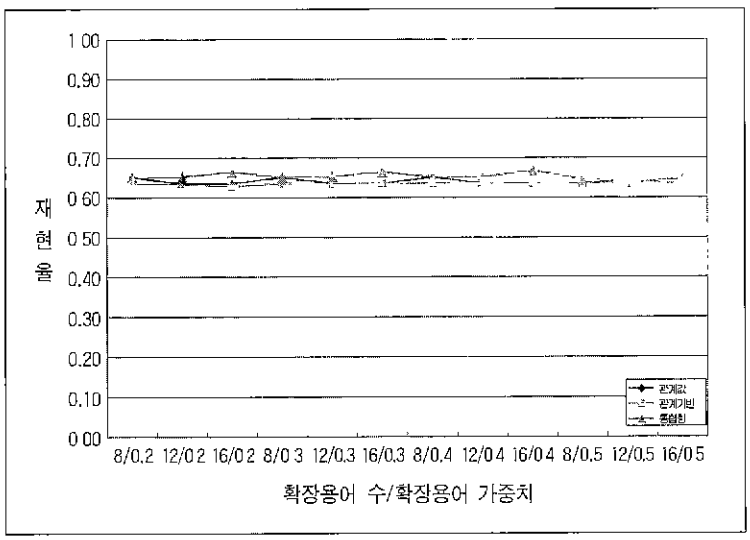
통합형 지식베이스는 확장용어 수를 8개로 할 때 가장 낮은 성능을 보여 주었고, 관계값기반 지식베이스는 확장용어 수를 16개로 할 때 비교적 낮은 성능을 보여 주었으며 관계기반 지식베이스는 변수 조정의 영향을 거의 받지 않는 것으로 나타났다. 〈그림 3〉은 NT, RT값을 각각 0.7과 0.2로 한 경우의 재현율이다.

다음으로 데이터베이스의 크기를 4,000건으로 하여 지식베이스간 성능차이를 분석하였다. NT값을 0.4와 0.5로 하고 RT값을 0.2에서 0.4로 변화시켜 가면서 성능을 비교한 결과, 2,000건에서와 달리 10위에서 50위까지 통합형 지식베이스의 성능이 가장 높게 나타났으며 관계기반 지식베이스와 관계값기반 지식베이스의 경

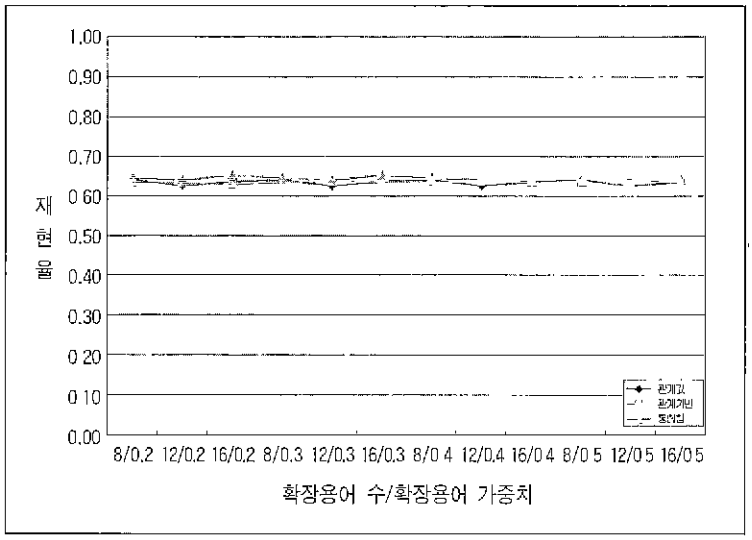


〈그림 3〉 50위까지의 재현율 (2,000건) - NT: 0.7, RT: 0.2

우 10위에서는 뚜렷한 성능차이를 보이지만 50위로 갈수록 성능차이가 점점 줄어들어서 50위에서는 거의 차이가 없는 것으로 나타났다. 통합형 지식베이스는 확장용어 수가 많을수록, 관계값기반 지식베이스는 확장용어 수가 적을수록 높은 성능을 보이고 있다. 모든 지식베이스는 확장용어 가중치가 0.5로 되면 검색성능이 전반적으로 낮아지는 것으로 나타났다. 〈그림 4〉는 NT, RT값을 각각 0.5와 0.4로 한 경우의 재현율이다. 그림 4는 NT, RT값을 각각 0.5와



〈그림 4〉 50위까지의 재현율 (4,000건) - NT: 0.5, RT: 0.4

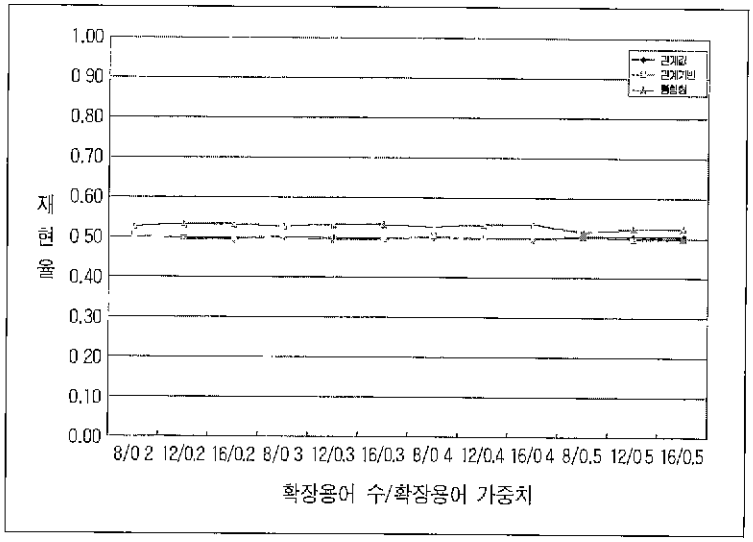


〈그림 5〉 50위까지의 재현율 (4,000건) - NT: 0.6, RT 0.2

0.4로 한 경우의 재현율이다.

NT값을 0.6과 0.7로 하고 RT값을 0.2에서 0.4로 변화시킨 경우도 위의 경우들과 거의 동일한 현상을 보이지만 평가할 검색문헌 수를 늘릴수록 세 지식베이스간 성능차이가 점점 줄어드는 것으로 나타났다. 〈그림 5〉는 검색문헌 수 중 상위 50건에 대한 평가 결과로서 확장용어 수와 확장용어 가중치가 각각 8과 0.4인 지점을 비교해 보면 10위에서 통합형 지식베이스의 재현율은 0.2781로 관계값기반 지식베이스보다 0.07%, 관계기반 지식베이스보다 0.19% 높았으나 50위에서는 통합형 지식베이스의 재현율은 0.6439로 관계값기반 지식베이스보다 단지 0.005%, 관계기반 지식베이스보다는 0.02%의 매우 근소한 차이를 보일 뿐이었다. 〈그림 5〉는 NT, RT값을 각각 0.6과 0.2로 한 경우의 재현율이다.

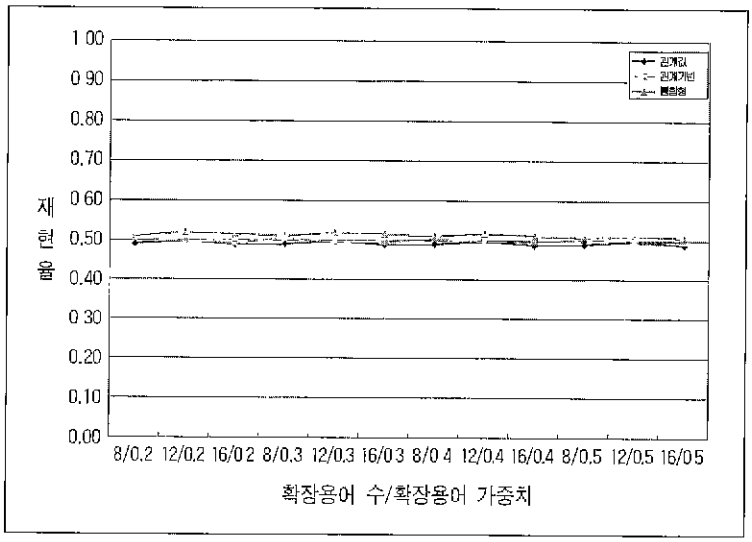
다음으로 데이터베이스의 크기를 6,000건으로 하여 지식베이스간 성능차이를 분석하였다. NT값을 0.4와 0.5로 하고 RT값을 0.2에서 0.4로 변화시켜 가면서 성능을 비교한 결과, 검색성능은 통합형, 관계값기반, 관계기반 지식베이스순으로 나타났고 확장용어 수가 많을수록 높은 성능을 보여 주었으며, 확장용어 가중치가 0.5로 되면 성능이 전반적으로 낮아지는 것으로 나타났다. 10위에서는 세 지식베이스간의 성능차이가 뚜렷하지만 비교대상 검색문헌 수를 늘릴수록 관계값기반 지식베이스와 관계기반 지식베이스의 성능은 거의 비슷해지는 반면 이 두 지식베이스와 통합형 지식베이스와의 성능차이는 점점 더 커지는 것으로 나타났다 〈그림 6〉은 검색문헌 수를 50위까지로 하여 분석한 재현율을 보여 주고 있다. 즉, 검색문헌 수를 10위 까지로 하였을 때 통합형, 관계값기반, 관계기반 지식베이스의 재현율은 각각 0.1894, 0.1792, 0.1894로 세 지식베이스간에 약간의 차이가 있었으나 50위까지로 하였을 때는 재현율이 각각 0.5268, 0.5013, 0.5032로 관계값기반 지식베이스와 관계기반 지식베이스의 검색성능은 거



〈그림 6〉 50위까지의 재현율 (6,000건) - NT: 0.4, RT: 0.4

의 동일하였다. 〈그림 6〉은 NT, RT값을 각각 0.4로 한 경우의 재현율이다.

NT값을 0.6과 0.7로 하고 RT값을 0.2에서 0.4로 변화시켜 가면서 검색결과를 비교한 결과, 비교대상 검색문헌 수를 10건으로 했을 경우 세 지식베이스간의 성능차이가 뚜렷한 반면 검색문헌 수를 늘릴수록 세 지식베이스간 성능차이가 점점 줄어드는 것으로 나타났다. 〈그림 7〉은 NT, RT값을 각각 0.7과 0.2로 한 경우의 재현율이다.



〈그림 7〉 50위까지의 재현율 (6,000건) - NT: 0.7, RT: 0.2

4.2 개념확장 조건의 평균 비교

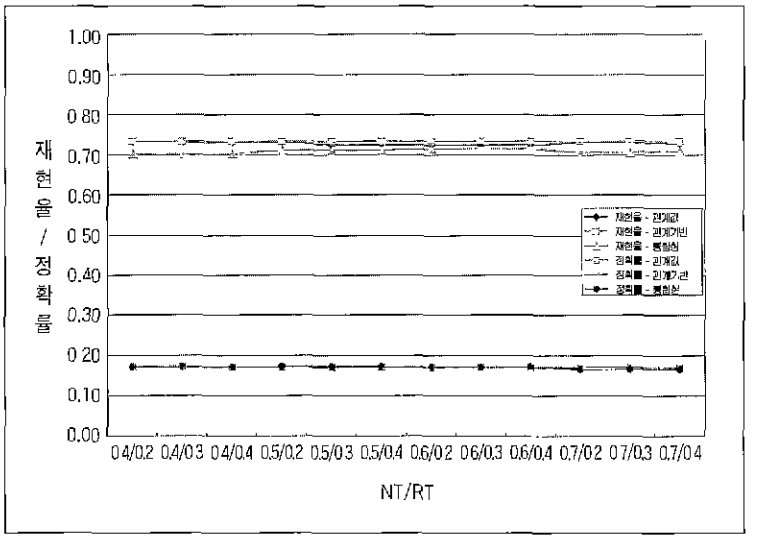
NT, RT의 값을 12가지 경우로 변화시키면서 다양한 개념확장 조건의 평균을 비교하였다. 먼저, 데이터베이스 크기를 2,000건으로 한 경우, 10위와 20위까지는 통합형 지식베이스의 성능이 가장 높게 나타났으나 30위에서 50위까지에서는 가장 낮은 성능을 보여 주었다. 관계기반 지식베이스와 관계값기반 지식베이스의 성능은 유사한 성능을 보여 주었다.

통합형 지식베이스는 NT값을 0.6으로 한 경우 RT값에 무관하게 높은 성능을 보여 주었고 NT값을 0.5로 할 경우는 RT값을 0.2로 하였을 때 높은 성능을 보여 주었다.

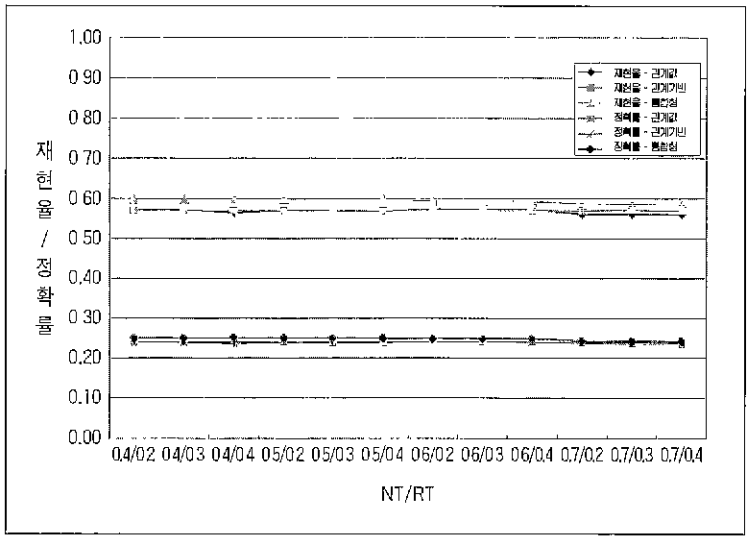
시소러스의 관계를 기반으로 개념확장을 한 경우 NT, RT값의 변화에 거의 영향을 받지 않는 것을 알 수 있으며, 관계값기반 지식베이스를 대상으로 개념확장을 한 경우 NT, RT값을 모두 0.4로 한 경우와 각각 0.7과 0.4로 한 경우에 비교적 높은 성능을 보이지만 역시 많은 차이를 보이고 있지는 않았다. 〈그림 8〉은 40위까지의 재현율 및 정확률을 비교한 것이다.

데이터베이스의 크기를 4,000건으로 하였을 경우에 10위에서 50위까지 통합형 지식베이스를 기반으로 한 개념확장이 가장 높은 성능을 보여주었고 관계값기반, 관계기반 지식베이스 순으로 나타났다. 통합형 지식베이스의 경우 NT값을 0.5로 한 경우에 가장 높은 성능을 보여 주었고 0.7로 한 경우에 RT값의 변화에도 불구하고 가장 낮은 성능을 보여 주었다.

관계값기반 지식베이스의 경우에는 NT값보다 RT값의 영향을 더 많이 받는 것으로 나타났다. NT관계에 0.4에서 0.7까지의 값을 부여한 모든 경우에 RT값을 0.2로 하였을 때 가장 높은 성능을 보여 주었으며 NT값을 0.7로 한 경우에는 검색성능이 매우 낮아지는 것으로 나타났다. 경험적 bnb 알고리즘을 적용한 관계기반 지식베이스의 경우는 NT, RT값의 변화에 성능



〈그림 8〉 2,000건-40위까지의 재현율 및 정확률 평균

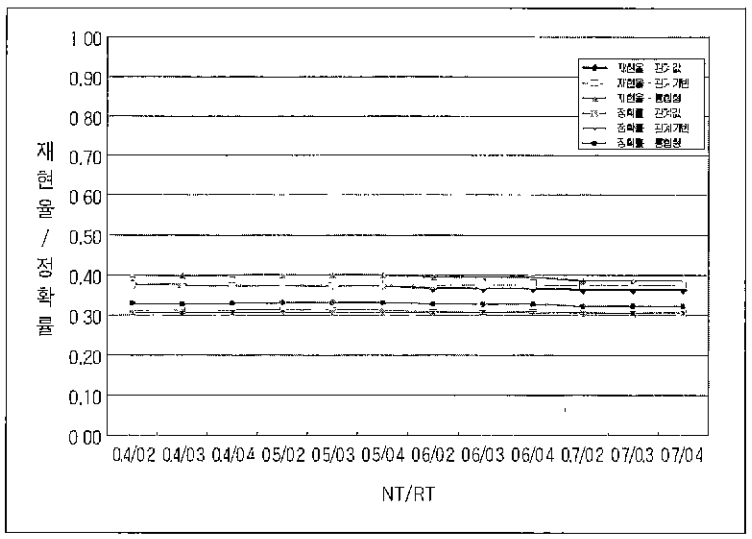


〈그림 9〉 4,000건-40위까지의 재현율 및 정확률 평균

차이가 거의 발생하지 않는 것으로 나타났다(〈그림 9〉참조).

데이터베이스의 크기를 6,000건으로 한 경우 역시 통합형 지식베이스를 대상으로 한 경우에 가장 높은 성능을 보이고 관계기반 지식베이스와 관계값기반 지식베이스는 유사한 성능을 보여주었다.

검색문헌 수를 30건으로 한 경우를 보면 통합형 지식베이스는 NT값을 0.5로 하였을 때 높은 성능을 보이고 다음으로 0.4, 0.6, 0.7순

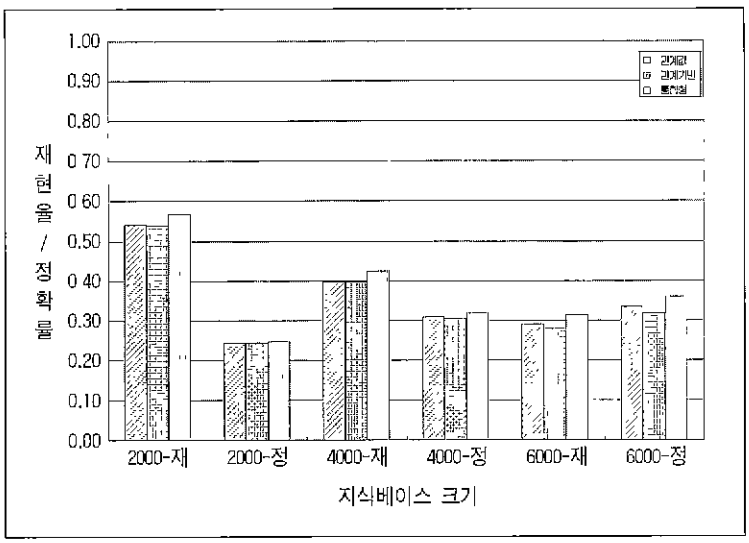


〈그림 10〉 6,000건-30위까지의 재현율 및 정확률 평균 비교

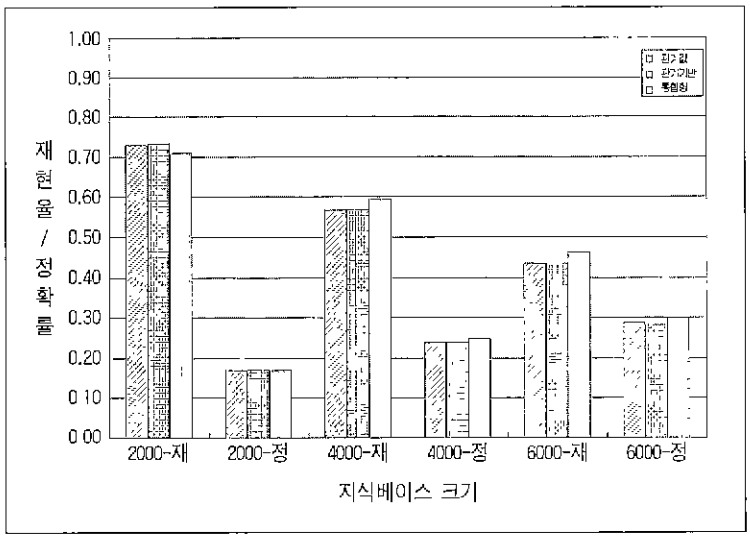
으로 나타났으며 각 경우에 RT값을 높게 하였을 때 좀더 좋은 성능을 보여 주는 것으로 나타났다. 관계기반 지식베이스는 NT, RT값의 변화에 영향을 받지 않는 것으로 나타났다(〈그림 10〉 참조).

4.3 데이터베이스의 크기에 따른 평균 검색성능 차이 분석

데이터베이스 크기에 따라 NT, RT에 대한 모든 조건의 평균을 10위에서 50위까지 비교하였



〈그림 11〉 20위까지의 전체평균



〈그림 12〉 40위까지의 전체평균

다. 그 결과 10위와 20위에서는 통합형 지식베이스가 재현율과 정확률에 있어서 2000건, 4000건, 6,000건 모든 경우에 가장 높은 성능을 보여주었고, 관계기반 지식베이스가 가장 낮은 성능을 보여 주었다. 그러나 30위에서 50위까지의 경우 통합형 지식베이스는 4000건과 6000건으로 한 경우에는 다른 지식베이스보다 높은 성능을 보여주지만 2,000건으로 한 경우에는 다른 지식베이스보다 낮은 성능을 나타내고 오히려 다른 모든 경우에 가장 낮은 성능을 보여준 관계기반 지식베이스는 가장 높은 성능을 나타냈다. 그러나 전반적으로 데이터베이스의 크기 변화에도 불구하고 통합형 지식베이스가 가장 높은 성능을 보이고 관계기반 지식베이스가 가장 낮은 성능을 보여줌을 알 수 있다. 그림 11과 12는 검색문헌 수를 각각 20건과 30건으로 했을 때의 재현율과 정확률이다.

5 결론 및 제언

최근에 개념기반 정보검색에 대한 연구가 활발하게 진행되고 있으며 다양한 조건변화에 따른 개념확장 알고리즘의 성능향상 방안 및 효과적인 지식베이스 구축방안이 연구되고 있다.

본 연구에서는 시스템에 의해 자동으로 구축된 문헌기반 지식베이스 이외에 인간 전문가에 의해 구축된 시소러스를 보다 더 효과적으로 활용함으로써 개념기반 검색의 성능을 향상시킬 수 있는 방안을 모색하고자 하였다.

이를 위해 관계값기반 지식베이스와 통합형 지식베이스에는 순차적 bnb 알고리즘을 적용하고 관계기반 지식베이스에는 경험적 bnb 알고리즘을 적용하여 검색성능의 차이를 발견하고자 하였다. 실험결과 다음과 같은 사실을 발견할 수 있었다.

첫째, 관계기반 지식베이스의 경우 확장용어 수 및 확장용어 가중치뿐만 아니라 개념확장시 각 관계에 부여되는 값에 의해 영향을 받지 않으며, 데이터베이스의 크기변화에도 영향을 받지 않는 것으로 나타났다. 관계값을 기반으로 한 개념확장은 NT값을 0.4나 0.5로 하고 RT값을 0.2나 0.3으로 한 경우에 높은 성능을 보여주었으며 NT값을 0.7로 하면 검색성능이 크게 낮아지는 것으로 나타났다. 통합형 지식베이스의 경우 NT값을 0.5로 한 경우에 가장 높은 성능을 보이고 NT값을 0.7로 한 경우 낮은 성능을 보이고 있는 것을 알 수 있다.

둘째, 관계기반 지식베이스는 변수조정의 영향을 많이 받지 않으며 또한, 평균으로 보았을 때 NT값, RT값의 변화에 영향을 거의 받지 않는 것으로 나타났다. 이는 관계기반 지식베이스를 대상으로 개념확장을 하는 경험적 bnb 알고리즘이 시소러스내의 전통적인 관계를 기반으로 확장하기 때문에 확장용어 가중치가 개념확장에 큰 영향을 미치지 않는 것으로 분석된다.

셋째, 데이터베이스의 크기가 2,000건일 때 대부분의 조건에서 관계기반 지식베이스의 성능이 높게 나타났다. 그러나 데이터베이스의 크기가 4,000건인 경우와 6,000건인 경우, 통합형 지식베이스의 성능이 가장 높게 나타났다. 또한 NT값이 0.4와 0.5인 경우 평가대상 검색문헌 수를 늘릴수록 관계기반 지식베이스와 관계값기반 지식베이스의 성능은 유사해지고 통합형 지식베이스와의 성능차이는 커지는 것으로 나타났다. 반면에 NT값이 0.6과 0.7인 경우 10위에서 뚜렷한 성능차이를 보이다가 검색문헌 수를 늘릴수록 세 지식베이스의 성능이 거의

유사해지는 것을 발견할 수 있었다.

넷째, 시소러스 내에 나타난 관계 중 NT관계와 RT관계에 다양한 값을 부여하여 검색성능을 비교하였는데, 검색성능은 세 지식베이스 모두에서 RT값보다는 NT값의 영향을 더 많이 받는 것으로 나타났으며 NT값을 0.7로 할 때 가장 낮은 성능을 보이고 0.4와 0.5로 할 때 대체적으로 높은 성능을 보이고 있는 것으로 나타났다.

위의 결과를 토대로 분석해 볼 때 개념확장에 있어서 검색성능은 개념확장조건으로 부여되는 확장용어 수 및 확장용어 가중치의 변화에 따라 약간의 성능차이가 발생하고 알고리즘 또는 지식베이스에 따라 높은 성능을 보여주는 조건에 약간의 차이가 있음을 발견할 수 있다. 또한 시소러스를 활용할 경우 NT값 및 RT값의 변화에 의해 약간의 성능변화가 발생하는 것을 알 수 있었다.

위와 같이 본 연구에서는 시소러스를 개념확장 정보검색에 활용할 때 개념확장 대상으로 활용하였다. 그러나 시소러스를 개념확장 대상 지식베이스로 활용하기 전에 검색대상이 되는 데이터베이스를 색인하는 과정에 사용함으로써 시소러스의 활용효과는 높일 수 있을 것으로 보인다.

참고문헌

노영희. 1999. 『의미망 구조의 지식베이스를 이용한 개념기반 정보검색기법에 관한 연구』. 박사학위논문, 연세대학교 대학원, 문헌정보학과.

정영미. 1993. 『정보검색론』. 개정판. 서울: 구미무역.

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

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., 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.

Davison, Colin H. 1986. "Improved of graphic displays in thesauri - through technology and ergonomics." *Journal of Document*, 42(4): 225-251.

Fox, E. A. 1987 "Development of the CODER system: a testbed for artificial intelligence methods in information retrieval." *Information Processing and Management*, 23(4): 341-366.

Humphreys, B. L. and Lindberg, D. A. 1989. "Building the unified medical

language system." In *Proceedings of the Thirteenth Annual Symposium on Computer Applications in Medical Care*. Washington, DC: IEEE Computer Society Press, November, pp. 5-8.

Pollitt, S. 1987. "Cansearch: An expert systems approach to document retrieval," *Information Processing and Management*." 23(2): 119-138.

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