객체기반 다중 시소러스 시스템의 설계 및 구현

정규상° 김원중 양재동
전북대학교
{kschong° kimwj jdyang}@chonbuk.ac.kr

Design and Implementation of an Object-Based Multi-Thesaurus System

Kyusang Chong° Wonjung Kim Jaedong Yang
Chonbuk National University

요 약

본 논문은 하나의 도메인에 대해 서로 연관된 여러 개의 시소러스를 효율적으로 구축, 관리, 검색, 브라우징, 항해하기 위한 확장된 객체기반 시소러스 시스템을 제안한다. 이 시스템은 1) 대규모의 시소러스를 효율적으로 분산하여 구축할 수 있는 기능을 제공하고, 2) 구축된 시소러스간의 일관성 있는 검색, 브라우징, 항해를 제공한다. 이러한 기능을 통해 사용자들은 내부적으로 분산된 시소러스를 하나의 시소러스처럼 이용할 수 있다.

1. 서 론

현재 인터넷의 발전과 더불어, 대량의 정보를 접할 기회가 많아짐에 따라, 정보검색 시스템의 중요성이 더욱 증가하고 있다. 정보검색 시스템은 일반적으로 사용자가 기술한 탐색어와 문서 상의 색인어와의 용어 불일치로 인해 재현률이 감소하는 문제를 가지고 있다[1]. 이러한 용어의 불일치 문제를 해결하기 위해 용어간의 관계성을 정의하여 탐색어의 의미를 확장함으로써 정확률(precision)과 재현률(recall)을 높일 수 있는 시소러스가 도입되었다[2]. 이미 많은 시소러스가 국내·외적으로 구축되었으며 현재도 구축 중에 있다. 대표적인 예로, "NASA 시소러스", "INSPEC Thesaurus", 그리고 "Roget Thesaurus" 등이 있다.

이러한 시소러스는 대부분 도메인에 종속적이며, 하나의 도메인 당 각각 하나의 시소러스로 구축되어 있다. 이처럼 하나의 도메인을 하나의 시소러스로 표현하는 경우, 시간이 지남에 따라 구축된 시소러스가 방대해 짐으로써 관리 및 검색 등의 효율이 떨어질 수 있다. 따라서 시소러스의 효율적인 관리 및 검색을 위해, 시소러스를 적당한 규모의 시소러스로 분리하여 구축할 필요가 있다. 그러나 하나의 시소러스를 여러 개의 분리된 시소러스로 구축하여 유지 관리한 경우, 일관성의 결여나 구축, 관리 비용 증대 등의 문제를 해결해야 한다.

본 논문에서는 도메인 시소러스를 논리적으로 분산 구축하여 효율적으로 관리할 수 있고, 분산된 시소러스를 의미적으로 일관성이 보장되는 하나의 시소러스처럼 통합 검색, 브라우징, 항해할 수 있는 시스템을 설계하고 구현하였다.

본 논문의 구성은 다음과 같다. 먼저, 2장에서는 객체기반 시소러스와 다중 시소러스 시스템에 대한 기존 연구를 소개한다. 3장에서는 본 논문에서 제안한 시소러스 분산 구축을 위한 객체기반 다중 시소러스 시스템을 설명하고, 4장에서는 설계된 시스템의 구현을 기술한다. 마지막으로 5장에서는 결론 및 향후 연구 과제를 제시한다.

2. 관련연구

2.1 객체기반 시소러스[2]

본 논문에서 제안한 시스템은 객체기반 시소러스를 채택하여, 실세계의 지식을 보다 효과적으로 모델링할 수 있도록 하였다. 객체기반 시소러스는 기존의 시소러스에 객체지향 패러다임의 구조적인 특성을 적용한 시소러스이다. 즉, 모든 시소러스 내의 개념을 객체로 간주하고, 이러한 객체들의 관계로 시소러스를 표현하는 방식의 시소러스 시스템이다. 객체는 개념을 추상화하는 개념 객체와 인스턴스를 추상화하는 인스턴스 객체로 구성된다. 객체들 사이의 관계는 기존의 시소러스 개념들 사이에 존재하는 상위어(BT : Broader Term), 하위어(NT : Narrower Term), 관련어(RT : Related Term) 관계를 일반화(super/sub-concept-of), 클래스화(owner/instance-of), 집성화(whole/part-of), 연관화(association-of) 관계로 의미에 따라 재정의하여 표현한다.

그림 1은 "Switching System" 객체기반 시소러스의 예이다.

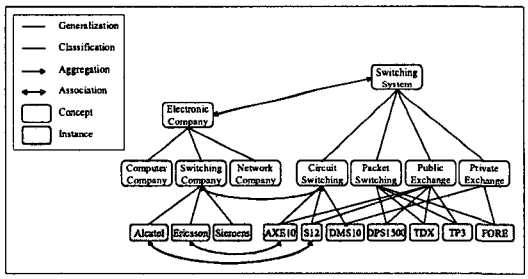

그림 1 객체기반 시소러스의 예

2.2 다중 시소러스 시스템

다중 시소러스는 넓은 의미로는 하나의 도메인을 여러 개의 시소러스로 구성하는 경우를 말하며, 좁은 의미로는 서로 다른 구축자에 의한 다른 관점의 이기종 시소러스들을 말한다[3]. 다중 시소러스 시스템은 이러한 다중 시소러스를 효율적으로 통합, 관리, 검색할 수 있는 시스템을 말한다. 지금까지의 다중 시소러스에 대한 접근법은 이러한 다중 시소러스를 일일이 도메인 전문가에 의해 통합하여, 하나의 거대한 슈퍼 시소러스(Super Thesaurus)[4]로 만들거나, 메타 시소러스(Meta Thesaurus)[5] 등을 통한 통합이었다[3]. 메타 시소러스는 하나의 시소러스의 개념을 다른 시소러스의 개념과 연관지을 수 있는 시소러스를 말한다. [4]는 통합된 하나의 시소러스로서 개념의 일관성이 높지만, 구축 비용이 많이 들며 거대한 규모로 인하여 시소러스를 이용한 복잡한 추론 및 검색에 있어 성능 저하를 초래할 수 있다. 메타 시소러스 등을 통한 통합 방법은 분산된 시소러스의 지원을 통한 구축 비용과 성능 측면에서 장점이 있지만, 단순 문자열 비교에 의한 통합 때문에 일관성이 결여될 가능성이 높다. 대표적인 예로, 동음이의어(homonym)에 대한 처리 문제를 들 수 있다.

본 논문에서는 다중 시소러스를 하나의 도메인에 대한 같은 구조를 가진 동종의 복수 시소러스로 그 의미를 제한하고, 슈퍼 시소러스가 제공하는 일관성과 통합의 효율성을 가지는 시스템을 제안한다. 슈퍼 시소러스나 메타 시소러스는 구축된 시소러스들 간의 통합을 다룸에 반해 본 논문에서 제안하는 객체기반 다중 시소러스 시스템은 시소러스를 구축하는 과정에서 이미 일관성 있는 다중 시소러스로 구축할 수 있다는 점에서 다른 관점을 가진다.

3. 객체기반 다중 시소러스 시스템

본 논문에서 제안한 객체기반 다중 시소러스 시스템은 하나의 도메인을 여러 개의 시소러스로 분산 구축이 가능하다. 또한 분산된 동종의 시소러스들 간의 의미적 일관성을 보장하며, 외부적으로는 하나의 시소러스처럼 유기적으로 작동하는 기능을 제공한다. 이를 위해 분산된 다중 시소러스에서 객체의 유일성을 보장하기 위한 GUOID(Global Unique Object IDentifier)와 참조 객체 그리고 다중 시소러스를 관리하기 위한 시소러스 리스트가 존재한다.

3.1 GUOID와 시소러스 리스트

본 시스템에서 GUOID는 3요소 튜플 <서버 식별자, 시소러스 식별자, 객체 식별자>로 이루어진다. 여기서 서버 식별자는 구축된 시소러스들이 존재하는 서버의 ID를, 시소러스 식별자는 시소러스 서버에 존재하는 시소러스의 유일한 ID를, 객체 식별자는 시소러스를 구축하는 객체들의 ID를 각각 명시한다. 예를 들어, <2, 2, 1>가 주어졌다면, 이는 서버 식별자는 '2'이고 시소러스 식별자는 '2'이며, 객체 식별자는 '1'인 GUOID를 명시하고 있다. 참조 객체는 다른 시소러스 내의 원본 객체를 지정하기 위한 객체이며, 일반적인 시소러스 객체의 속성 외에 reference-of라는 추가적인 속성을 가지게 된다. reference-of 속성은 원본 객체에 대한 GUOID 정보를 가진다.

시소러스 통합 데이터베이스(그림 4 참조)는 4요소 튜플 <서버 주소, 서버 식별자, 시소러스 식별자, 시소러스 이름>로 이루어진 시소러스 리스트를 가진다. 여기서 서버 주소는 시소러스 서버가 존재하는 컴퓨터의 주소를, 서버 식별자는 구축된 시소러스들이 존재하는 서버의 ID를, 시소러스 식별자는 시소러스 서버에 존재하는 시소러스의 유일한 ID를, 시소러스 이름은 시소러스 유일한 이름, 그룹 식별자는 같은 도메인 시소러스를 나타내는 ID를 각각 명시한다. 예를 들어, <21.11.17.21, 2, 2, Switching System>이 주어졌다면, 이는 서버 주소는 '21.11.17.21'이고 서버 식별자는 '2'이며, 시소러스 식별자는 '2'이고 시소러스 이름이 'Switching System'인 시소러스를 명시한다.

3.2 참조 객체를 통한 시소러스 구축

본 시스템에서 새로운 객체를 추가하거나, 객체간의 관계를 설정할 경우, 먼저 모든 시소러스 내에 해당 객체가 존재하는지 여부가 중요하다.

먼저, 용어를 추가하는 경우를 설명하기로 한다. 일단 모든 시소러스들로부터 각각의 용어에 대한 객체를 검색한다. 검색되어 나온 각각의 결과를 reference-of 속성을 이용하여 통합한다. 통합된 각 객체들의 시소러스 정보를 이용하여, 추가하고자 하는 객체와 동일한 객체가 존재할 경우, 그 객체가 현재 시소러스 내의 객체이면 용어 추가는 취소된다. 만일 그 객체가 다른 시소러스 내의 객체이면 현재 시소러스에 그 객체에 대한 참조 객체를 생성한다.

용어간에 관계를 설정하는 경우에는, 일단 위와 같이 용어 추가의 단계를 거쳐서 생성된 객체(원본 객체나 참조 객체)에 관계를 설정하면 된다.

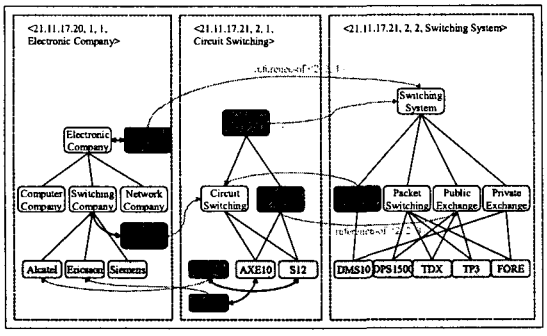

그림 2 참조 객체를 통한 시소러스간 용어 관계 설정

그림 2는 그림 1의 "Switching System" 시소러스를 세 개의 시소러스("Switching System", "Electronic Company", 그리고 "Circuit Switching")로 분리 구축한 경우이다. 이 경우 시소러스간 관계 설정의 예를 들면, "Electronic Company" 시소러스의 "Electronic Company" 객체를 "Switching System" 객체와 연관 관계를 설정할 경우, 시소러스 통합 데이터베이스의 <21.11.17.21, 2, 1, Circuit Switching>, <21.11.17.21, 2, 2, Switching System> 시소러스 리스트로부터 "Switching System" 객체를 검색한다. 만약, "Switching System" 원본 객체가 존재한다면 <2, 2, 1> GUOID를 가지는 "Switching System" 참조 객체를 생성하고, 생성된 참조 객체와 "Electronic Company" 객체와 연관 관계를 설정하고, 존재하지 않는다면 새로운 객체 "Circuit Switching"을 생성하여 연관 관계를 설정하면 된다.

3.3 객체기반 다중 시소러스 검색

본 시스템에서는 분산되어 있는 각각의 시소러스로부터 참조 객체를 통해 하나의 완전한 객체로 통합할 수 있다. 먼저, 특정 객체에 대한 시소러스 정보를 요청받으면 각 시소러스에서 요청된 시소러스 정보들을 구한다. 이렇게 구해진 여러 개의 분리된 시소러스 정보는 병합 과정을 거쳐 하나의 통합된 결과로 합쳐지게 된다.

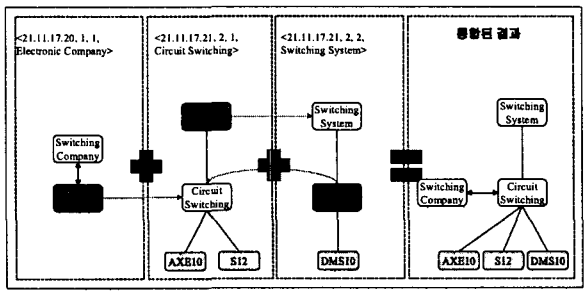

그림 3 참조 객체를 통한 검색 결과의 통합

그림 3은 분산된 3개의 시소러스로부터 각각 검색한 결과를 하나의 검색 결과로 통합하는 과정을 보여준다. 예를 들어, "Circuit Switching" 객체를 검색한다고 하자. 우선, 시소러스 통합 데이터베이스에 존재하는 시소러스 서버 리스트로부터 "21.11.17.20" 서버에 "Electronic Company" 시소러스가, "21.11.17.21" 서버에 "Circuit Switching", "Switching System" 시소러스를 검색할 수 있다. 검색된 각 시소러스에 "Circuit Switching" 탐색어를 질의한 결과는 다음과 다.

"Electronic Company" 시소러스에서 "Switching Company" 객체는 "Circuit Switching" 참조 객체와 연관 관계를, "Circuit Switching" 시소러스에서 "Circuit Switching" 객체는 "Switching System" 참조 객체와 일반화 관계를, "AXE10" 객체, "S12" 객체와는 클래스화 관계에 있음을 알 수 있다. 마지막으로 "Switching System" 시소러스에서 "Circuit Switching" 참조 객체는 "Switching System" 객체와 일반화 관계를, "DMS10" 객체와는 클래스화 관계에 있음을 알 수 있다. 이들 각각의 검색 결과로부터 존재하는 참조 객체를 이용하여 하나의 통합된 "Circuit Switching" 정보를 보여준다.

3.4 객체기반 다중 시소러스 항해

본 시스템에서는 분산되어 있는 각각의 시소러스로부터 참조 객체를 통해 다른 시소러스의 객체로 항해할 수 있다. 항해하고자 하는 객체가 참조 객체인 경우, reference-of 속성에 지정된 원본 객체의 GUOID를 이용하여 원본 객체로의 항해가 이루어진다.

예를 들어, 그림 2의 "Circuit Switching" 시소러스에서 "Public Exchange" 객체로의 항해를 한다고 하자. 우선, "Public Exchange" 객체가 참조 객체이기 때문에, reference-of 속성의 <2, 2, 3>의 GUOID를 이용하여 서버 식별자가 '2'이고, 시소러스 식별자가 '2'인 시소러스를 시소러스 리스트에서 파악할 수 있다. 그리고 이 시소러스의 서버정보와 시소러스 식별자를 통해 원본 객체로의 항해를 할 수 있다.

4. 객체기반 다중 시소러스 시스템의 구현

4.1 시스템 구조

그림 4는 본 논문에서 제안한 객체기반 다중 시소러스 시스템으로

3-계층 구조를 가진다.

Application 계층은 도메인 전문가들을 위한 객체기반 다중 시소러스 관리기와 웹 검색 인터페이스로 구성된다. 이 관리기는 도메인 전문가가 분산된 객체기반 시소러스를 마치 통합된 하나의 시소러스처럼 구축할 수 있는 기능, 통합 검색 등의 기능을 제공한다. 웹기반 검색기는 사용자들에게 웹브라우저를 통해 시소러스 확장 검색을 제공한다.

Mediator 계층은 다양한 어플리케이션의 요구를 받아, 시소러스 통합 데이터베이스(Thesaurus Integration DB)와 각각의 독립적인 시소러스 데이터베이스에 검색을 수행하고, 리턴된 결과를 어플리케이션에 되돌려 주는 역할을 하는 계층이다. 시소러스 관리 서버(Thesaurus Management Server)는 시소러스 통합 데이터베이스를 관리하는 기능과 각각의 시소러스 데이터베이스에 시소러스를 구축, 관리하는 기능을 제공한다. 시소러스 병합 서버(Thesaurus Merging Server)는 시소러스 관리 서버와 웹 검색 인터페이스의 요청을 받아, 각각의 시소러스 데이터베이스로부터 부분적인 정보를 리턴 받은 뒤, 객체의 reference-of 속성에 의해 하나의 완전한 정보로 병합을 수행한다.

Database 계층은 시소러스 서버 리스트와 각 서버들의 시소러스 리스트를 관리하는 시소러스 통합 데이터베이스와 각 시소러스 데이터베이스로 구성된다.

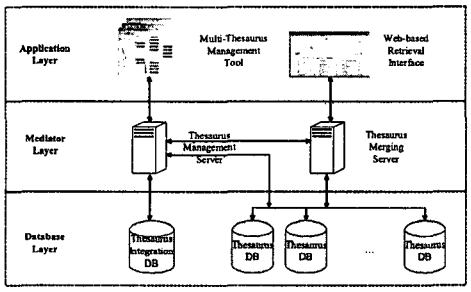

그림 4 객체기반 다중 시소러스 시스템 구조

4.2 관리기 사용자 인터페이스

본 시스템에서 객체를 추가하거나, 객체간의 관계를 설정할 경우, 시소러스 관리기는 시소러스 병합 서버에 객체에 대한 정보를 요청한다. 시소러스 병합 서버는 각 시소러스 데이터베이스에 대해 스레드를 생성한뒤, 관리기로부터 넘겨받은 탐색어를 이용하여, 병렬적으로 검색을 수행한다. 이 과정을 통해 각 스레드는 자신이 담당하는 시소러스 데이터베이스로부터 일차적으로 탐색어와 일치한 결과를 리턴받게 된다. 각각의 결과를 얻은 시소러스 병합 서버는 reference-of 속성을 가진 객체들의 시소러스 정보를 원본 객체의 시소러스 정보와 통합한 뒤, 통합 결과를 관리기에 되돌려 준다.

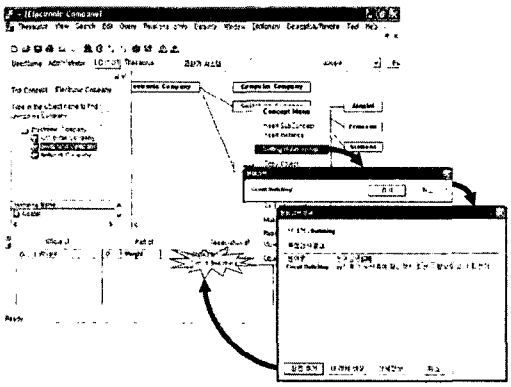

그림 5 객체 검색 및 통합검색 결과 리스트

예를 들어, 그림 5처럼 사용자가 "Electronic Company" 시소러스의 "Switching Company" 객체와 "Circuit Switching" 객체를 연관 관계로 설정하기 위해, 모든 시소러스로부터 "Circuit Switching" 객체들의 리스트를 얻을 수 있다. 사용자는 이 리스트의 객체들을 상세정보를 통해 객체의 상하위 관계 및 관련어 정보를 참조할 수 있다. 이러한 상세정보를 통해 사용자는 동음이의어(homonym)에 대한 여부를 판별할 수 있다. 그림 5의 예에서, 사용자가 리스트에서 "Circuit Switching" 객체를 선택하고, "참조 설정" 버튼을 누르면, 관리기의 "Association of" 정보 창을 통해 연관 관계가 설정된 "Circuit Switching" 객체를 참조할 수 있다. 또한 관리기의 객체나 정보창을 이용하여, 통합적인 시소러스 정보를 항해할 수 있다.

4.3. 웹 검색 인터페이스

사용자가 웹 검색 인터페이스를 통해 시소러스를 이용한 확장 검색을 할 경우, 관리기 사용자 인터페이스의 경우처럼, 시소러스 병합 서버를 통해, 통합된 확장 검색 결과를 얻는다.

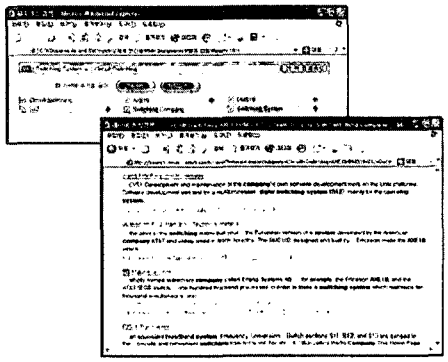

그림 6 시소러스를 이용한 확장 검색

예를 들어, 그림 6은 사용자가 "Circuit Switching"을 탐색어로 시소러스를 이용한 확장 검색 결과이다.

5. 결론 및 향후 연구 과제

본 논문에서는 분산 환경에서 하나의 도메인을 여러 개의 시소러스로 분산, 구축, 관리할 수 있는 확장된 객체기반 시소러스 시스템을 제안하였다. 이 시스템은 참조 객체를 통해 외부적으로 하나의 시소러스처럼 동작하면서 사용자에게 효율적으로 정보를 제공할 수 있다.

향후 연구로는 다국어 시소러스나 패싯 등의 특별한 개념을 지원하기 위한 포인터 객체 지원에 대한 연구와 대형 포털 사이트에 범용적 시소러스 제공을 위한 본 시스템의 확장에 대한 연구이다.

참고문헌

[1] M. Mitra, A. Singhal and C. Buckley, "Improving Automatic Query Expansion", In Proceedings of the 21th Annual International ACM/SIGIR Conference, pp.206-214, Melbourne, Australia, 1998

[2] 최재훈, 김기헌, 양재동, "객체기반 시소러스 시스템의 설계 및 구현: 반자동화 방식의 구축, 추상화 방식의 개념 브라우징 및 질의 기반 참조", 정보과학회 논문지(데이타베이스), Vol.27, No.1, pp.64-78, March, 2000

[3] Ralf Nikolai, Andreas Traupe, Ralf Kramer, Thesaurus Federations: A Framework for the Flexible Integration of Heterogeneous, Autonomous Thesauri, Advances in Digital Libraries Conference, pp.46-55, April, 1998

[4] A. Stern and N. Rischette, On the construction of a super thesaurus based on existing theauri, In Tools for Knowldge Organization and the Human Interface, volume 2, pp.134-144, Indeks Verlag, Frankfurt/Main, 1990

[5] Steven J Squires, Access to biomedical information: The unified medical language system, Library Trends, 42(1), pp.127-151, 1993