『신영어영문학』, 16집 (2000. 8), 신영어영문학회, 227-243쪽

자연언어처리를 위한 개념도식 도구와 그 활용

정 미 애

I. 서 론

자연언어처리를 위해서는 지식을 표현하고 처리할 수 있는 효율적이고도 강력한 시스템이 요구된다. 이를 위해 지금까지 다양한 이론과 프로그램을 이용한 많은 자연언어처리 도구가 개발되고 있다. 그 중 하나가 개념도식을 이용하는 자연언어처리 시스템 개발이다. Sowa에 의해 개발된 개념도식(conceptual graph, CG)은 논리기반 형식으로서 Charles S. Peirce의 존재도식과 의미망을 바탕으로 한다. 개념구조는 자연언어에 대한 의미론적 표상으로서의 편리함이 이미 증명되었으며 CG를 이용한 자연언어처리 시스템의 개발이 전 세계적으로 활발히 진행되고 있다.

그러나 이렇듯 컴퓨터를 이용한 실제 운용시스템 개발이 다양하게 진행되고 있음에도 불구하고 우리 나라에서는 CG를 바탕으로 하는 연구가 아직은 이론적인 연구, 그 중에서도 특히 언어학적인 현상을 CG를 이용하여 설명하는 것에 편중되어 있는 것이 사실이다. 물론 이러한 언어학적인 측면에서의 연구가 자연언어처리 시스템 개발에 가장 기초적이면서도 필수적인 바탕을 제공하는 것임에는 분명하지만 이와 동시에 그러한 연구들이 컴퓨터를 이용한 자연언어처리 도구에서 어떻게 활용될 수 있는지를 고려하고 그에 활용되도록 함으로써 단순한 이론으로서가 아니라 그 연구결과를 실제로 적용할 수 있어야 할 것이다. 그러므로 본 연구에서는 개념도식을 이용한 언어처리 시스템으로는 어떠한 것들이 개발되어왔는지를 살피고 또한 그러한 도구들 중 최근에 새로이 개발된 몇 가지를 예로 소개하여 그 도구들이 어떻게 실행되며 그것이 보완해야 할 점이 무엇인지를 살펴보고자 한다.

II. 자연언어처리 시스템

최근 활발히 논의되고 있는 인지과학 분야에서는 자연언어에 대한 연구가 커다란 흐름을 이루고 있다. 또한 인공지능(AI) 분야에서는 자연언어이해 시스템(Natural Language Understanding System)에 대한 연구가 진행되고 있다. 컴퓨터 과학, 또

는 전산언어학(computational linguistics)에서는 우리가 단어에 대하여 가지고 있는 모형이 컴퓨터 시스템에 어떻게 반영될 수 있으며, 그러한 모형을 기술하거나 그것들을 외부 시스템에 어떻게 반영할 수 있는지, 그러한 외부 시스템과 연결시키기 위하여 어떠한 언어와 도구가 필요한지, 그리고 그 모형을 통해 사람들이 쉽게 컴퓨터와 인터페이스 할 수 있는지 등을 고려한다.

자연언어처리(Natural Language Processing; NLP)의 목적은 자연언어를 분석하고 생성할 수 있는 컴퓨터 처리모형을 만드는 것이라고도 할 수 있다. 그런데 컴퓨터에서 자연언어라는 개념은 ENIAC의 탄생과 동시에 구상되었다고도 볼 수 있다. 즉 자연언어가 컴퓨터 프로그래밍에서 기호나 문자열로 사용되면서 자연언어처리는 기호처리의 일반적인 형식이 되었다. 실제로 1950년대 후반에서 1960년대 중반에 걸친 자연언어처리 기술에 대한 연구는 기계번역 시스템에 대한 연구가 주를 이루었지만 그 당시의 연구는 의미처리나 문맥처리를 무시하고 단지 문장구조에 대한 분석에 중점을 두었다. 그 후에야 비로소 의미정보처리에 대한 연구의 중요성이 인식되었으며, 그러한 연구성과로 탄생한 것이 Winograd의 SCHRDLU(1973)라는 자연언어이해 시스템이다. 이러한 문장분석과 이해 시스템을 구축하기 위한 목적을 가지고 SCHRDLU 외에 MARGIE (영문번역 시스템 1973), MILISYJ (일본어 QA 시스템 1975), VISUALIZER (일본어 QA 시스템 1976), SAM, TOPLE (담화이해 시스템 1976), GUS (여행계획 QA 시스템 1977), EXPLUS (일본어 이해 시스템 1978), MSSC-78 (일본어 QA 시스템 1979) 등이 개발되었다.

최근에는 인간의 지적능력을 본뜬 인공지능 시스템 개발이 활발히 이루어지고 있으며, 인공지능의 성과를 바탕으로 하는 정보처리 분야로서 지식공학(knowledge engineering)이 급속히 발전하고 있다. 지식공학에서는 지식을 어떻게 표현할 것인가라는 것이 가장 중요한 문제이다. 즉 문제해결을 위한 전제조건, 대상, 해결방법 등에 관한 지식을 컴퓨터 프로그램이 실행할 수 있는 형식으로 표현하는 방법을 모색하는 일이다. 가능하다면 우리가 일상생활에서 사용하는 자연언어나 수식 등을 자유로이 사용할 수 있는 방법을 찾는 것이 그 최상의 목표라고 말할 수 있다. 물론 이러한 지식표현을 실현하는 일은 아직 요원해 보이지만 몇가지 개념에 착안하여 구체적인 기준을 목표로 한 연구가 활발히 진행되고 있다. 즉 프로그램상의 표현과 그것이 나타내는 대상의 개념구조가 대응하도록 지식을 표현하고자 하여 지식의 계층이나 지식간의 관계를 고려하는 지식구조화 시스템을 개발할 수 있다. 또한 지식기반(Knowledge Base; KB)의 유연성, 즉 새로운 지식을 추가, 변경하는 등 지식기반을 쉽게 조작할

수 있는 방법을 모색하는 연구가 있을 수 있으며 또한 지식자체가 갖는 여러 속성을 표현하는 것에 중점을 두거나 인간의 사고형태와의 대응, 즉 인지과학의 견지에 바탕을 둔 연구방향이 있을 수 있다.

지식공학은 크게 지식표현(knowledge representation)과 지식모형화(knowledge modelling)라는 두 가지 영역에서의 지식을 필요로 한다. 이와 같은 두 영역을 포괄하면서 지식처리와 자연언어 처리를 통합하는 지식 시스템의 핵심적인 모형으로서 최근에 이론적으로 뿐만 아니라 컴퓨터를 통한 실제적인 실행에 있어서도 매우 활발한 연구성과를 보이고 있는 것이 개념도식 이론이다. 다음에서 개념도식이 컴퓨터 시스템에서 실제로 어떻게 적용되는지 살펴보겠다.

III. 개념도식 처리 도구

Sowa(1984)의 개념도식 이론에 대한 연구가 발표된 이후로 지식기반 시스템(knowledge-based system)을 만들기 위하여 지식표현 구조를 이용하고자 할 때 발생하는 다양한 이론적인 문제들 또는 컴퓨터 실행상의 문제들에 대한 폭넓은 연구와 개발이 진행되어왔다.

컴퓨터에서의 실행을 위한 이러한 개념도식 도구들과 프로그램, 또는 언어들은 그 연구집단에 따라 매우 다양한 유형으로 발전해 오고 있다. 지금까지 개발되어오고 있는 것들 중에서 널리 알려져 있는 도구들로는 다음과 같은 것들이 있다. 우선 지식관리(knowledge management)를 위한 도구로서 DEAKIN TOOLSET, CGKEE, CoGiTo, PROLOG+CG, CGPro, CGeditor, CG-PENMAN, LUT TOOLSET, PEIRCE, GRIP, CP, HamPeirce, PIP 등이 있으며, 지식습득과 모델링(knowledge acquisition & modelling)을 위한 도구로서 MODEL-ECS, CG-KADS, MDBCASE, CG-DESIRE, CGKAT, WebKB 등이 있다. 또한 CG 프로그래밍을 위한 것으로 SYNERGY, PROLOG+CG, ECS 가 있으며, 자연언어처리와 이해(NLP & NLU)를 위한 언어로 RECIT, CGLex, NLG, BEELINE 등이 널리 알려져 있다(Lukose 1997).

본 논문에서는 컴퓨터 실행을 위한 CG 도구 중 최근에 개발되어 널리 알려져 있는 몇 가지를 택하여 살펴보고자 한다. 본 논문에서 소개하고자 하는 CG도구들은 CG 홈페이지에 소개되어 있는 대표적인 도구들로서, CG 편집기(editor)인 CharGer와 웹을 통한 CG 탐색과 지식표현을 위한 도구인 CGWorld, 그리고 CG에 기반을 둔

객체지향적 Prolog 프로그램인 PROLOG+CG이다. 다음에서 이러한 도구들 각각의 특성과 운용에 대하여 살펴본 후 그러한 도구들이 보완해야 할 점이 무엇인지를 살펴보고자 한다.

3.1 개념도식 편집기 CharGer

CharGer는 연구프로젝트와 교육에 도움을 주기 위하여 Delugach에 의해 개발된 개념도식 편집기이다. 이는 개념도식 연구자들이 개념도식 인터페이스에 있어서의 실행문제를 탐구할 수 있도록 하기 위한 도구이다.

CharGer 2.3 버전은 2000년 6월에 발표되었으며, 다음과 같은 특징을 갖는다.

- 설치(installation)가 쉽다.
- 개념, 관계, 작용자(actor), 문맥(context), 동일지시 연결 등이 도입된다.
- 유형계층과 관계계층을 편집하고 또 이를 저장할 수 있다.
- 도식에 표지를 붙일 수 있다. (query, definition 등등)
- 내포문맥(nested context) 등을 나타낼 수 있다.
- 선택된 개념에 대한 도식결합(graph joining)이 가능하다.
- 도식매칭이 가능하다.
- 선택된 개념을 총칭적(generic) 개념으로 바꿀 수 있다.

또한 아무리 많은 수의 도식창(graph window) 일지라도 편집을 위해 동시에 열 수 있으며 하나의 도식안에서 또는 도식사이에서 도식상의 요소를 복사하거나 자르고 붙일 수 있다. 그리고 대부분의 편집상의 수정을 원상태로 돌리거나 또는 재실행(undo/ redo)할 수 있다.

아울러 CharGer는 호환성이 뛰어나서 모든 주요한 플랫폼(platform), 예를 들어 자바로 바꾸어 사용할 수 있으며 표준적이지 않은 텍스트 형태로 된 파일로부터 도식을 저장하거나 검색할 수 있다. 또한 CGIF(Conceptual Graph Interchange Form) 형태로 도식을 저장할 수 있으며 CGIF 형태로 도식을 받아들이거나 배열할 수 있을 뿐만 아니라 도식을 영어표현으로 바꾸어 주기도 한다.

이제 CharGer를 실행해 보자. 일단 허브창(hub window)이 뜨면 기존의 도식을 열거나 아니면 새로운 도식을 만들 수 있다. 다음은 편집창(editing window)에 작성된 도식을 보여준다. 도식을 작성하거나 편집하기 위해서는 왼쪽의 아이콘이 보여주

는 편집자질들을 이용한다. 이러한 편집도구들로는 선택, 개념, 관계, 작용자, 화살표, 유형표지, 유형계층, 글자편집, 삭제, 문맥형성, 수직/수평배열 등의 것들이 있다.

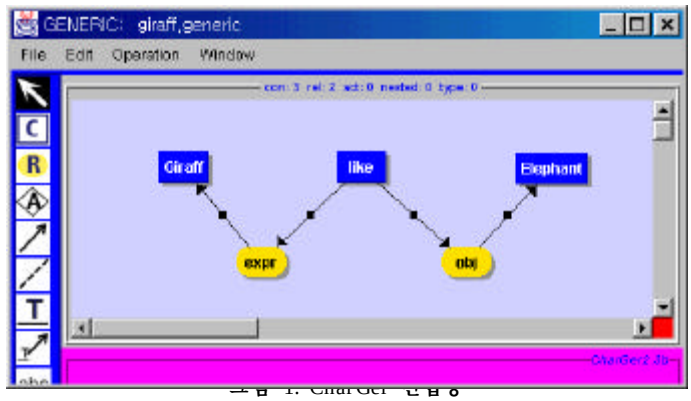


그림 1. CharGer 실행화면

일단 이와 같은 도식이 형성되면 파일메뉴에 CGIF와 영어문장으로의 변환장치를 통해 표현방식을 바꿀 수 있다. 위의 도식에 대하여 이를 실행시키면 다음과 같은 창을 볼 수 있다.

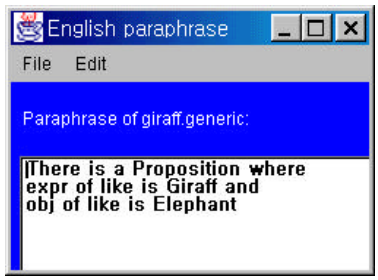


그림 2. CharGer 영어변환

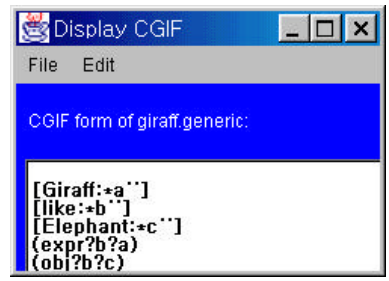


그림 3. CharGer CGIF 변환

아울러 CharGer는 데이터베이스를 위한 파일을 별도로 관리함으로써 유형계층등의 자료를 작성하여 보관할 수 있으며 이를 도식으로 불러들일 수 있음으로써 유형이나 기타 기본자료들을 매번 작성해 넣어야 하는 번거로움을 줄여준다. 아래의 두 창은 데이터베이스에 저장되어 있는 하나의 예를 보여주며 왼쪽의 'Set Up Graph'를 선택하면 이 자료에 대한 도식을 자동적으로 얻을 수 있다.

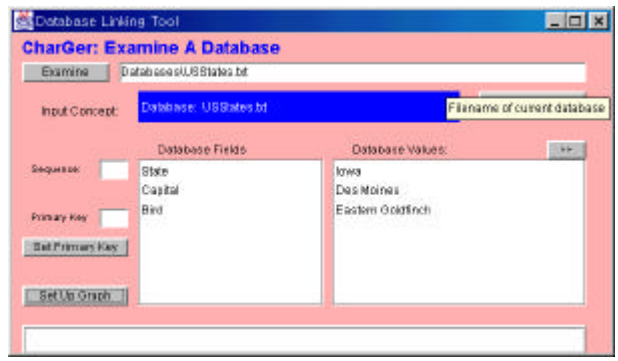
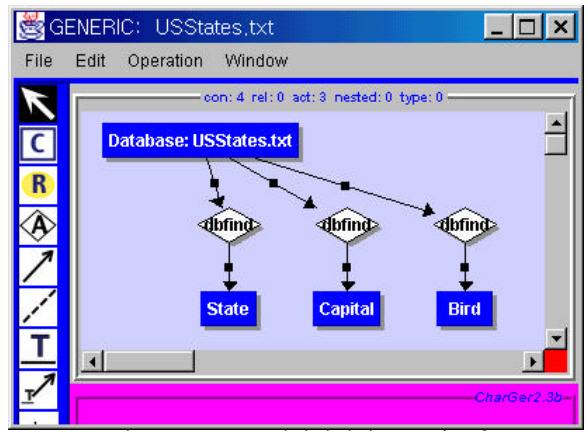


그림 4. CharGer 데이터베이스



이와 같이 CharGer는 손쉽게 도식을 작성하고 편집하여 저장할 수 있으며 특히 그 조작방법이 간단하여 전문가가 아니라도 손쉽게 작업을 할 수 있다는 장점이 있다. 그러나 개념도식 이론의 연산들, 즉 도식들 사이의 연합, 복사, 일반화, 특수화 등의 연산을 위한 도구들을 제공하지 못한다는 점에서 보완해야 할 점이 남아 있다.

3.2 웹기반 편집기 CGWorld

CG 연구자들이 직면하는 문제중 하나는 CG를 기반으로 하는 응용도구를 개

발하기 위하여 사용하고 있는 CG의 인코딩과 플랫폼, 그리고 소프트웨어 도구들이 서로 다르다는 것이다. 또한 개념도식의 대형 지식기반의 개발을 위해 세계 각국에서 수년동안 다양하게 연구되어온 결과들을 함께 수용할 만한 그럴듯한 CG 편집도구가 없다는 것이 또다른 문제점이다. 이러한 문제점을 인식하고 해결할 수 있는 방법을 모색하고자 Dabrew와 Toutanova에 의해 개발된 것이 바로 CGWorld(http://www.larflast.bas.bg:8080/login.html)이다. 이는 개별적으로 전개되어온 개념도식의 지식기반과 응용을 통합하는 틀이라 할 수 있다.

CGWorld는 개념도식의 대형 지식기반의 개발을 위한 웹기반(web-based)의 작업공간으로서 중심서버에 존재한다. CGWorld는 그래픽 형식, CGLex, FOPC, 그리고 CGIF 등 네가지 형식들을 개념도식으로 번역하기 위한 기능, 규범적 형성규칙(canonical formation rules)의 실행, 그리고 대형 지식기반(KB)에서의 탐색을 용이하게 만드는 브라우징 기능들을 포함하고 있다.

이 시스템은 LARFLALST 프로젝트를 수행하기 위하여 만들어졌는데 이는 전반적인 지능 외국어 용어 학습 시스템(generic intelligent foreign language terminology learning system)을 개발하는데 있어서 자연언어처리 기술을 적용하는 것이 그 목적이다. 그 프로젝트는 지식표상 형식 안에서 다양한 소스언어, 목적언어 그리고 기술적인 분야에 대한 많은 다중언어적 도구(multilingual tool)를 실행시키는 것뿐만 아니라 언어학적, 의미론적, 그리고 교육학적 지식을 인코딩하기 위한 지식기반을 개발하고자 하였다.

CGWorld는 보편적인 인터넷 클라이언트, 즉 브라우저(Browser)를 사용하여 인터넷에 연결된 모든 컴퓨터로부터의 지식기반(KB)에 접근하고 이를 편집하는 것을 가능하게 한다.

간단히 말해서 CGWorld는 다음과 같은 요구를 충족시키고자 개발되었다.

- 협력적으로 또는 산재되어 진행되고 있는 CG 지식기반에 대한 습득과 편집을 도입하여 활용하고자 한다.
- 대형 지식기반에서의 탐색이 쉽다.
- 다양한 표상언어를 제공함으로써 CGWorld의 다양한 사용자들의 욕구와 CG의 지식기반을 이용하는 다양한 응용들을 충족시키고자 한다.
- CG 이론에 관한 전문가가 아닐지라도 CG 그래픽 편집기와 뷰어를 쉽게 사용할 수 있도록 한다.

● 다양한 프로그래밍 언어로 쓰여진 현재 또는 기존의 여러 가지 CG 응용에 능가하는 웹을 통합하고 부가하고자 한다.

CGWorld의 클라이언트는 인터넷 사용자라면 누구나 가능하다. 서버는 클라이언트를 관리하고 프롤로그 에이전트와 클라이언트 사이의 정보전달을 관리한다. 즉 CGWorld는 다음과 같은 구조를 통해 작용한다.

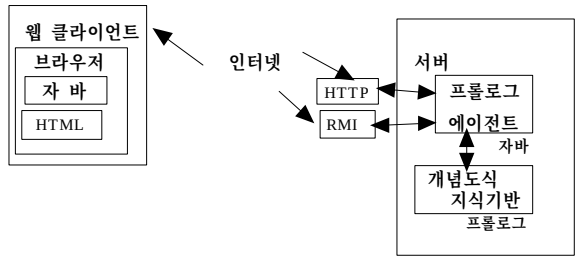


그림 6. CGWorld의 작용구조

CGWorld 편집기가 시작되면 다음과 같이 이미 저장되어 있는 자료를 로드하거나 도식을 새로 작성할 수 있다.

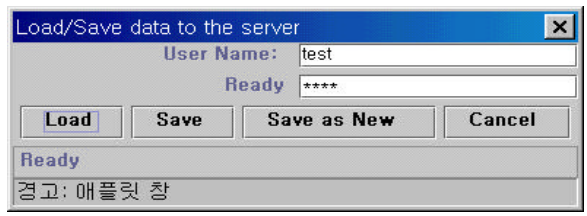
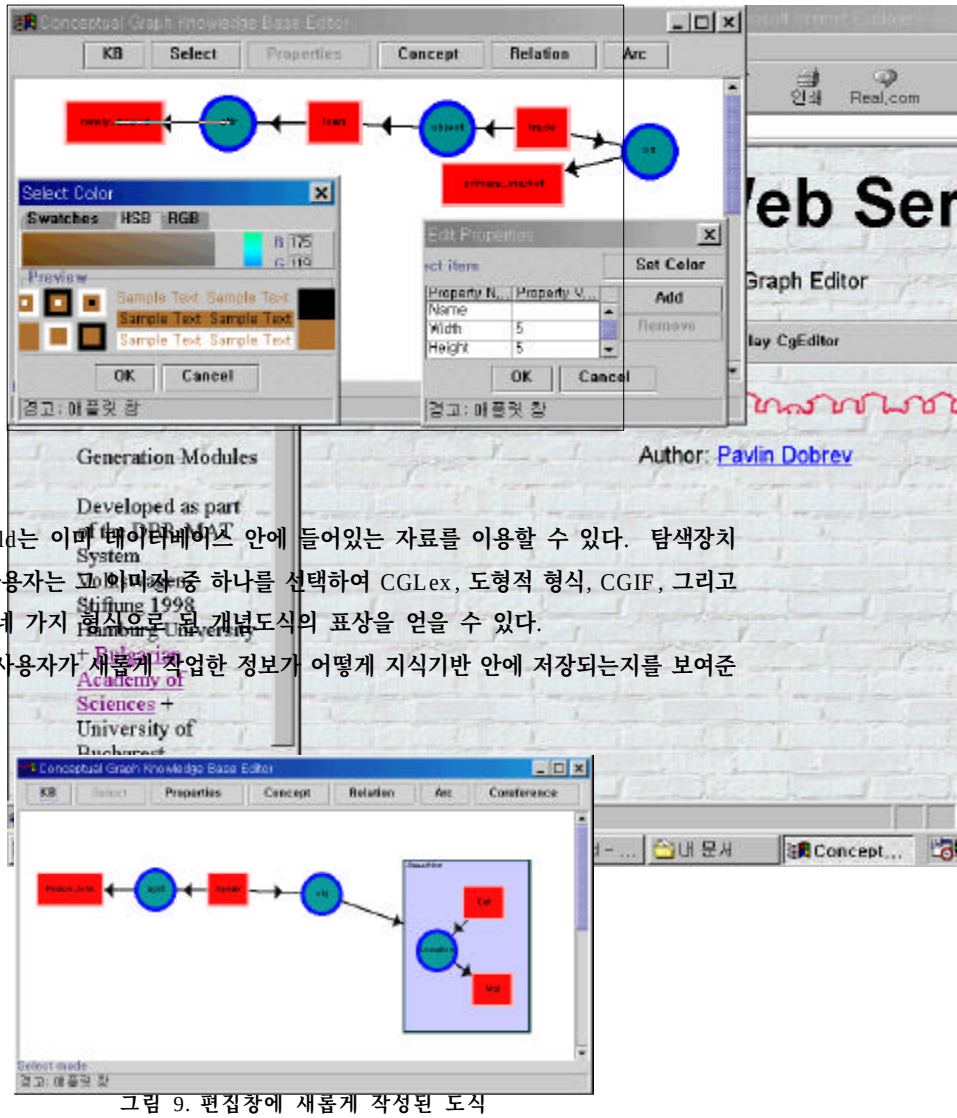


그림 7. 서버와의 인터페이스

또한 도식을 새로 만들 때 다음과 같이 개념과 관계, 또는 문맥에 대한 크기, 이름, 색상 등의 속성을 자유로이 변경하는 인터페이스가 가능하다.

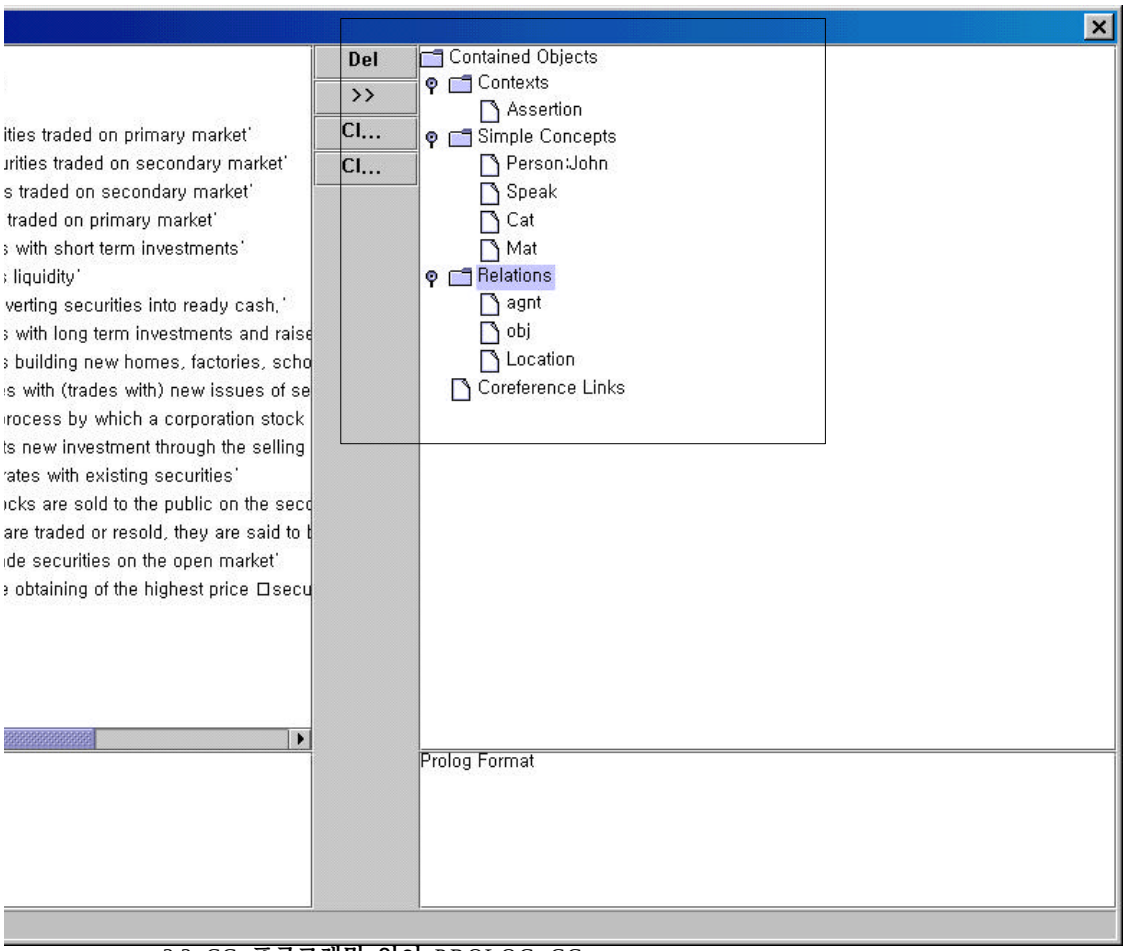


CGWorld는 이미 데이터베이스 안에 들어있는 자료를 이용할 수 있다. 탐색장치를 통해 사용자는 그 이미지 중 하나를 선택하여 CGLex, 도형적 형식, CGIF, 그리고 FOPC의 네 가지 형식으로 된 개념도식의 표상을 얻을 수 있다.

다음은 사용자가 새롭게 작업한 정보가 어떻게 지식기반 안에 저장되는지를 보여준다.

그림 9. 편집창에 새롭게 작성된 도식

일단 위와 같은 도식을 만들면 그곳에서 사용된 정보들이 기존의 지식기반 안에 새롭게 추가됨을 다음의 지식기반 창을 통해 알 수 있다.



3.3 CG 프로그래밍 언어 PROLOG+CG

이제 CG가 직접 프로그래밍 언어로 사용되는 CG도구를 살펴보겠다. 객체지향적 논리 프로그래밍 언어(object-oriented logic programming language)인 PROLOG+CG는 CG 연구에 있어서 CG를 기반으로 하는 Prolog가 필요하다는 점을 고려하여 Kabbaj에 의해 개발되었다.

PROLOG+CG는 자바로 실행되며 텍스트 편집기(text editor), 컴파일러(compiler), 그리고 해석기(interpreter)로 이루어져 있다. 컴파일러는 프로그램의 구문분석을 수행하며 만일 분석이 성공적으로 이루어진다면, 즉 프로그램이 문법에 맞

는다면 일단 그에 대한 목적파일(object file; .obj)을 생성한다. 그런 다음 해석기는 그 목적파일에 작용한다.

인터페이스 단계에서 PROLOG+CG는 프로그램, 디버그, 그리고 질의를 위한 창이 있다. 일단 시스템의 창이 뜨면 저장된 파일을 불러오거나 PROLOG+CG 파일 창에 새로운 프로그램을 작성할 수 있다. 그 후 목적파일을 만들기 위하여 프로그램을 컴파일하며 디버그(debug)가 성공적으로 끝나면 제어(request session console)창의 ?- 뒤에 질문을 하여 답을 얻을 수 있다.

PROLOG+CG가 제공하는 기본연산(primitive operation)으로는 산술적 목적(arithematic_goal), 관계적 목적(relational_goal), 논리적 목적(logical_goal), 목록목적(list_goal) 뿐만 아니라 개념유형 목적(conceptualTypes_goal)과 메타목적(meta_goal)이 있으며 각각은 세부문법구조를 갖는다. 다음은 PROLOG+CG의 세가지 작업창을 보여준다.

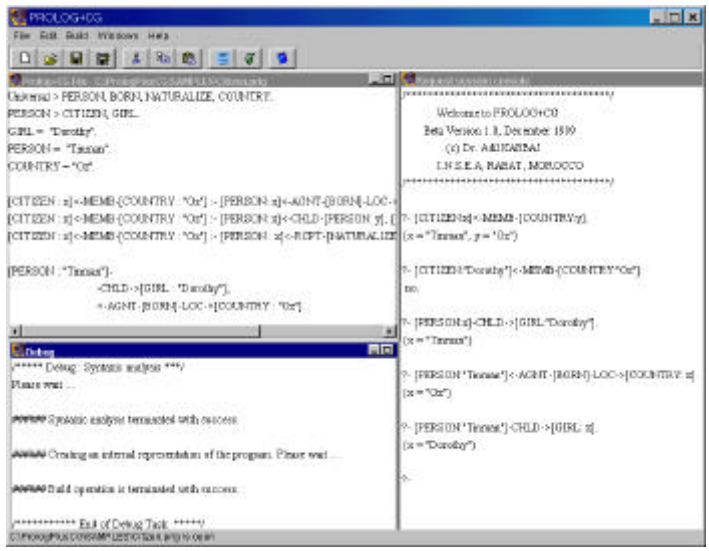


그림 11. PROLOG+CG의 세가지 창

우선 PROLOG+CG를 통해 개념유형계층에 대한 하위유형(SubType) 검증과 최대하위공통유형(Maximal Common SubType)에 대한 탐색을 할 수 있다.

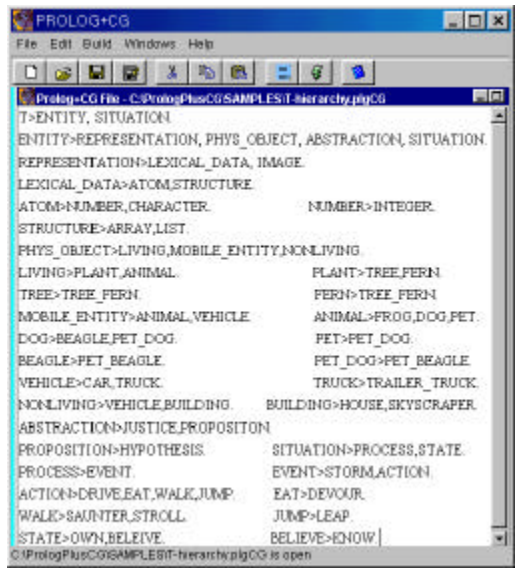


그림 12. 개념유형 계층 프로그램의 한 예

그림 12는 Sowa(1992:8)의 유형계층의 한 예를 프로그램으로 작성한 것이며 이를 바탕으로 다음과 같은 질문을 통해 유형간의 관계를 파악할 수 있다.

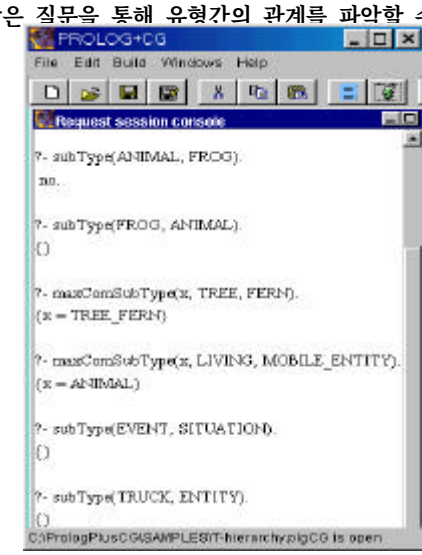


그림 13. 계층관계에 대한 질의응답

이제 다음과 같이 

를 살펴보자.

그림 14. 개념도식을 이용한 프로그램

개념도식으로 작성된 이러한 프로그램에 다음과 같은 질문을 할 수 있다.

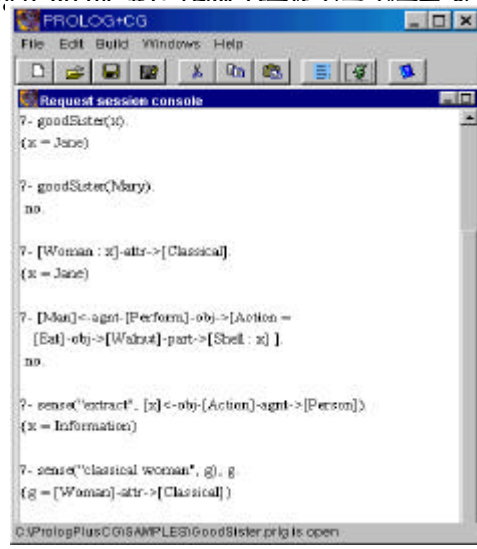


그림 15. 그림 14의 프로그램에 대한 질의응답

지금까지 살펴본 PROLOG+CG는 Prolog를 객체지향적으로 확장한 프로그램으로서 CG를 기반으로 하는 다양한 도구들에 응용될 수 있는 계기가 되었다. 그러나 작업상의 공학적 인터페이스 문제를 고려하지 않더라도 해결해야할 문제를 여전히 안고 있다. 즉 언어단계 특히 기본연산 단계에서 비 함수적인 CG를 고려해야 할 것이며, 상위유형(superType)의 탐색을 제공할 수 있어야 할 것이다. 그리고 단순, 복합 CG에 대한 다양한 연산들 (MaximalJoin, Specialize, Generalize, Subsume, Analog, Expand, Contrast, 등등)을 이용할 수 있도록 확장되어야 한다. 또한 PROLOG+CG는 앞에서 살펴보았던 편집기에서 볼 수 있는 지식기반을 별도로 가지고 있지 않다. 즉 유형계층관계등을 각 프로그램마다 일일이 다시 입력해야 하는 번거로움이 있다. 또한 비록 PROLOG+CG가 그래픽적인 CG 편집기는 아니라 할지라도 앞선 편집기에서와 같이 도식편집을 위한 도구를 추가함으로써 더욱 손쉽게 프로그램을 작성하고 또 다양한 연산과정을 시각적으로 더 잘 보여줄 수 있을 것이다. 필자는 이러한 몇가지 문제를 PROLOG+CG 시스템 개발자인 Kabbaj와 직접 논의한 바 있으며, 위에서 언급한 문제중 몇가지를 해결할 수 있는 장치를 기존 시스템에 추가시키는 작업이 현재 진행중임을 알 수 있었다. 또한 기타 언어학적인 여러 현상들, 예를 들어 조응관계 해결등에 대한 처리 문제 등에 대하여 계속하여 의견을 교환하고 있다. 반을 마련하는

IV. 결 론

본 연구에서는 자연언어처리를 위한 도구로서 널리 사용되고 있는 CG기반 시스템들 중 최근에 개발된 몇가지를 중심으로 살펴보고 이러한 도구들을 실제로 실행시켜 봄으로써 그들이 갖는 장점과 보완점을 살펴보았다.

우선 CG를 생성하고 편집할 수 있는 그래픽 편집도구로서의 CharGer와 CGWorld를 살펴보았다. CharGer는 여러 가지 편집장치를 갖추고 있어서 도식을 형성하기가 매우 편리하였다. 또한 형성된 도식에 대한 CGIF와 영어로의 자동변환 장치는 매우 흥미로웠다. 그리고 KB를 별도로 관리할 수 있으므로 무수한 자료를 언제나 손쉽게 끌어쓸 수 있는 장점을 가지고 있음을 알 수 있었다. 반면 연합, 복사, 일반화, 특수화 등과 같은 CG연산을 위한 장치를 제공하지 못하는 등 여전히 보완해야 할 점을 가지고 있음도 알 수 있었다.

CharGer는 개별적인 연구를 위한 프로그램인 반면 CGWorld는 웹을 통해 자료를 주고받음으로써 이미 개발되어 있는 정보를 이용하거나 추가시킴으로써 공동작업이

가능한 CG 편집기이다. 이는 인터넷을 통해 원거리에서 행해지고 있는 CG연구를 통합할 수 있다는 점에서 주목할 만하다. CGWorld는 CG 편집뿐만 아니라 로드된 무수한 지식을 이용할 수 있으며 여러 가지 CG형식을 자동적으로 제공해 준다. 반면 인터넷을 통해서만 접속되고 실행되는 프로그램이므로 개별 PC에 다운로드할 수 없으므로 어느 정도의 공간적 제약을 받는다는 한계가 있다.

이러한 CG 편집기 이외에 지식관리 도구로서 뿐만 아니라 프로그래밍을 위한 언어로서의 PROLOG+CG를 살펴보았다. 이 시스템은 지식을 CG로 프로그램화하여 저장할 수 있으며 다시 이러한 저장된 지식으로부터의 질의응답이 가능하다. 이는 개념유형계층에 따른 정보뿐만 아니라 추론적인 탐색도 가능하다는 점에서 매우 흥미롭다. 그러나 아직 다양한 기본연산(primitive)을 제공하지 못한다는 점에서 프로그램 작성상의 한계가 있다는 단점이 있다.

전반적으로 볼 때 지금까지 개발되어온 CG 도구들은 컴퓨터 실행의 용이성이라는 공학적 측면이 중시되어왔으므로 이를 통해 실제 자연언어를 만족스럽게 처리하기란 현재로서는 그리 수월해 보이지는 않는다. 즉 CG가 자연언어 처리 시스템으로서의 역할을 충분히 수행하기 위해서는 언어학적인 연구의 결과를 충분히 받아들여야 할 것이다. 한편 언어학적인 연구, 특히 개념도식을 이용하는 언어학적인 연구에 있어서는 단순한 표기방법으로서 개념도식을 이용하는 것이 아니라 실제적인 컴퓨터 운용에도 관심을 가짐으로써 이론과 실제의 상호협력을 조화롭게 이루어 나갈 수 있도록 해야 할 것이다. <혜전대>

인용문헌

Bontacheva, K. “Other Activities.” http://www.dcs.shef.ac.uk/ kalina/activities.html

Burrow, A. L. and P. W. Eklund. “Visual Structure Representations and Conceptual Graphs.” *Proceedings of the Fourth International Workshop on Peirce: A Conceptual Graph Workbench*.. 1994, 3-10.

Conceptual Graph Standard Information Technology(IT)-Conceptual Graphs draft proposed American National Standard(dpANS) NCITS.T2/98-003. http://www.bestweb.net/ sowa/cg/cgpansw.html.

Delugach, H. S. “CharGer-A Conceptual Graph Editor.” http://www.cs.uah.edu/ delugach/CharGer/body.html

Dobrev, P. and K. Toutanova. “CGWorld-A Web Based Workbench for conceptual Graphs Management and Applications.” http://www.larflast.bas.bg:8080/Index.htm

Ellis, G. “Object-Oriented Conceptual Graphs.” *Conceptual Structures: Applications, Implementation and Theory*. ICCS ’95, 1995, 144-57.

Kabbaj, A. “INSEA_CGTools.” http://www.insea.ac.ma/CGTools/CGTools/CGTools.htm #Installation.

Lukose, D. “Conceptual Graphs Tools Home Page.” http://www.cs.uah.edu/delugach/CG/CGTools.html.

Munday, C. and D. Lukose. “Object-Oriented Design of Conceptual Graph Processor,” *Proceedings of the Fourth International Workshop on Peirce: A Conceptual Graph Workbench*. 1994, 55-70.

Möller, J. “Activities concerning Conceptual Graphs.” http://nats-www.informatik.uni-hamburg.de/ jum/research/cg.html

_____. “Proposal of a Research Project: A Modular Conceptual Graphs Knowledge Base.” http://nats-www.informatik.uni-hamburg.de/jum/research/modularkb.html

Möller, J. and D. Wiese. “Editing Conceptual Graphs.” *Conceptual Structures: Knowledge Representation as Interlingua*. ICCS '96, 1996, 175-87.

Petermann, H., L. Euler, and K. Bontcheva. “CGPro-a PROLOG Implementation of Conceptual Graphs,” FBI-HH-M-25/95, 1995.

Sebesta, R. W. *Concepts of Programming Languages* (4th ed.) Addison Wesley Longman, Inc. 1999.

Southey, F. “Notio Project Page.” http://backtrack.math.uwaterloo.ca/CG/projects/notio/

Sowa, J. F. “Conceptual Graphs Summary.” In Nagle, Nagle, Gerholz, and Eklund, eds., *Conceptual Structures: Current Research and Practice*. Ellis Horwood Workshops, 1992, 1-51.

Sowa, J. F. *Knowledge Representation: Logic, Philosophical, and*

Computational Foundations. Pacific Grove, CA: Cole Publishing, 2000.

<Abstract>

CG Tools and their applications for the Natural Language Processing

Jung, Mie-Ae

Natural language processing requires efficient and powerful tools for representing and processing knowledge. And one of the prominent theories for NLP is Conceptual Graph Theory by Sowa. The purpose of CG theory is to express meaning in a form that is logically precise, humanly readable, and computationally tractable. Because it can be directly mapped to language, CG can serve as an intermediate language for translating computer-oriented formalism to and from natural languages. Since Sowa(1984), there have been a large number of researchers carrying out extensive research and development on various theoretical and practical issues in utilizing this knowledge representation scheme for building knowledge based systems. However, in Korea, there have been few researches on developments and applications of CG tools. Therefore, this paper tries to introduce some of the prominent CG tools for CG editing and programming such as CharGer as a graphical CG editor, CGWorld as a web-based knowledge Base Browser, and PROLOG+CG as a knowledge management and CG programming tool.