

성능 향상을 위한 제약 만족에 의한 지식베이스 재사용

현우석

한국성서대학교 정보과학부
wshyun@bible.ac.kr

Knowledge Base Reuse by Constraint Relaxation for Performance Improvement

Woo-Seok Hyun

Department of Information Science, Korean Bible University

요 약

지식베이스 시스템의 중요한 구성 요소로는 유용한 지식베이스의 어떤 조합이 주어진 과업을 위해서 선택된 문제 해결자의 요구사항을 만족시키는지를 고찰하는 지식베이스 집합과 문제 해결자들의 집합을 들 수 있다. 가능한 조합을 인식하는 것은 일련의 제약만족문제를 실험하는 것과 해결될 수 없는 것을 제거하는 것을 필요로 하는데, 제약만족 문제의 모순성 발견을 신속히 하기 위해 본 연구에서는 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법을 사용하였다. 완화 접근법을 시험하기 위해서, 기존의 급성복통과 관련된 지능형 질환 진단 시스템(A-IDS-DAAP)과 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법을 적용한 급성복통과 관련된 지능형 질환 진단 시스템(CR-IDS-DAAP)을 가지고 실험을 하여 평균 수행시간을 감소시켜 시스템의 성능을 향상시키게 되었다.

1. 서 론

지식 공학은 새로운 지식획득과 관련되고 새로운 문제를 해결하는 시스템을 구축하고자 할 때 시간 소모적이고 비용이 드는 과정이 된다[1-3]. 지식 재사용은 존재하는 구성 요소로부터 부분적으로 새로운 시스템을 구축함으로써 인하여 이러한 논제에 주목하고 있다. 어떤 구성 요소가 새로운 태스크(task)에 주목하기 위하여 사용되어질 수 있는지 인식하는 것은 어려운 일 중 하나이며, 이것은 절실히 요구되는 문제이다. 그 문제를 다루는 하나의 접근 방법은 하나의 단순한 계산 틀에서 문제 해결, 지식 획득, 기계학습을 병합하기 위한 MUSKRAT(Multistrategy Knowledge Refinement and Acquisition Toolbox) 틀(framework)과 같은 자문 시스템과 도구를 만드는 것이다[4]. 자문 시스템의 중요한 구성 요소는 자문가인데, 이것은 유용한 지식베이스의 어떤 조합이 주어진 과업을 위해서 선택된 문제 해결자의 요구사항을 만족시키는지를 고찰하는 주어진 지식베이스 집합과 문제 해결자들의 집합으로 구성된다. White는 Meta-PS를 제안했는데[4], 모순된 지식베이스-문제해결자 조합을 인식하고 제거하는 다루기 쉬운 실험

을 수행한다. 그러나 Meta-PS의 주된 결점은 성공적인 지식베이스-문제해결자 조합이 잘못 무시되지 않는다는 것을 보증할 수 없다는 것이다. 이것을 보완하기 위하여 나타난 연구는 성공적인 지식베이스-문제해결자 조합이 잘못 무시되지 않는다는 것을 보증하는 제약만족기법을 사용하여 그럴듯한 test를 생성해내는 방법을 고려함에 의해 White의 제안을 발전시킨 방법[5]이다. 본 연구에서는 문제 해결자와 같은 제약 해결자를 사용하였고, 합성 제약만족문제를 만들기 위해서 병합될 수 있는 제약만족문제(CSP:Constraint Satisfaction Problem)로서 존재하는 스케줄링 지식베이스를 사용하였다. 합성 제약만족문제가 해결될 수 없다면 지식베이스-문제해결자 조합은 주어진 문제를 해결하기 위해서 재사용되어질 수 없다. 가능한 조합을 인식하는 것은 일련의 제약만족문제를 실험하는 것과 해결될 수 없는 것을 제거하는 것을 필요로 한다. 제약만족문제가 모순 된다는 것을 증명하는 것은 시간이 오래 걸리는 과정이 될 수도 있어서 이러한 모순성의 발견을 신속히 하기 위해 본 연구에서는 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법[5]을 사용하였다. 그리고 완화된 버전은 모순성을 증명하여 본래의 제약만족문제는 더 이상 해결책을 가지지

않게 된다. 실제적으로 해결될 수 없는 제약만족문제를 완화하는 것은 더 쉬운 문제를 만들어 낸다. 임의로 제약을 제거하는 것은 실질적으로 몇 배 어려운 문제를 만들어낸다-제약이 점점 더 느슨해지거나 문제들 간의 연관성이 희미해지고 임의의 문제에 대한 모순성을 증명하기 위한 시간이 증가되기도 한다. 완화 접근법을 시험하기 위해서, 급성복통과 관련된 지능형 질환 시스템[6]을 가지고 실험을 하여 시스템의 성능이 향상되었음을 보였다.

2. 연구 배경

지식베이스 시스템은 은퇴한 사람의 지식을 보존하기 위해서, 회귀한 기술을 보존하기 위해서, 의사결정을 지원하기 위해서, 특정 전문영역에서 몇몇 전문가와 기계들로부터 유용한 모든 지식을 융합하기 위해서 등 다양한 이유로 개발되어 왔다. 지식베이스 시스템은 회사들이 허용하기 쉬운 인공지능의 한 분야로서 비즈니스에서 성공적인 많은 예들[7]이 나오고 있다. 지식베이스 시스템은 생산, 마케팅, 고객 서비스에서도 자주 나타나고 스케줄링 영역[8-9]에서 가장 성공적인 시스템이 나오고 있다.

대부분의 지식베이스 시스템에서 지식 획득은 시간 소모적이어서 비용이 많이 드는 작업이다[1-3]. 만약 새로운 시스템이 존재하는 구성요소를 재사용함에 의해서 구축될 수 있다면 더 신속히 구축될 수도 있고 비용도 절감될 수도 있다. 결과적으로 지식베이스 시스템을 만드는 가장 중요한 목표는 지식베이스 시스템의 구성요소를 사용하기 위한 것[10]이고, 몇몇 다른 구성요소들을 재사용하기 위해서 제안되어 진다. 지식공학의 세부영역에서 초기 단계의 연구자들은 문제 해결의 전 영역에서 포함되어야 할 문제해결방법(PSM :Problem Solving Method)의 범위를 명시하고 합병 테스크(Synthesis Task)라 불리는 계획(Planning)을 통한 진단(diagnosis)과 분류(Classification)를 위한 방법들을 포함한다[11]. 문제해결방법의 재사용에 대한 초기의 강력한 예 중의 하나는 전염병의 특정 전문영역의 다양성을 지니는 EMYCIN shell[12]이었다. 재사용과 관련된 현재의 연구로는PS/PSM[13], 온톨로지[13-14], 지식베이스[3, 10, 15] 등 다양한 구성요소가 재사용되어지는 시스템에 대한 연구를 들 수가 있다. 또한 사례기반 추론에서 사례의 사용[16]도 관련된 연구로 들 수가 있다.

2.1 지식베이스 재사용

검색, 번역, 이해, 비교, 분할, 재형성, 병합[1] 등은 비표준화된 지식베이스를 재사용하는 어려운 작업에 도움을 주는 몇 가지 과정들로 언급될 수 있다. 온톨로지

(ontology)는 지식해결자의 지식 요구와 할당(mapping), 병합(merging), 지식베이스의 재사용을 위한 기회를 증가시키기 위하여, 지식베이스에서 존재하는 지식 사이에서 구조적이고 문법적인 유사성을 위한 검색을 유용하게 함에 의해 중요하게 부각되고 있다[15]. PROTEGE는 OKBC(Open Knowledge Base Connectivity)와 OWL(Web Ontology Language)[17]과 같은 표준화된 형식으로 지식베이스를 쓰기 위한 옵션을 제공하는데, 이것들은 지식베이스에 있어서 필요한 병합과 할당을 유용하게 해 준다. 본 논문에서는 고찰되어야 할 지식베이스가 동일한 언어로 쓰여지고 공통적인 온톨로지를 사용하여 표준화되어있다고 가정하였다.

2.2 제약 프로그래밍

제약 프로그래밍은 여러 가지 실세계 문제에 성공적으로 적용되어졌다. 왜냐하면 이런 문제들은 스케줄링, 계획, 배치, 설계, 자원 할당과 결정 지원[18] 등 제약을 조건으로 쉽게 모델링 되어질 수 있기 때문이다. 이 기법은 제약만족문제들에 대해서 해결책을 찾고자 하며 이러한 문제들을 다루기 위해서 고안된 효율적인 여러 가지 도구 모음[19]이 존재한다. 제약만족문제는 다음과 같이 정의된다.

- $X = \{X_1, \dots, X_n\}$, X : 변수들의 집합
- 각각의 변수 X_i 에 대하여 전문영역에서 가능한 값들의 한정된 집합 D_i
- 제약의 집합 $C_{\langle j \rangle} \subseteq D_{j_1} \times D_{j_2} \times \dots \times D_{j_t}$, 여기에서는 동시에 일어날 수 있는 변수들의 부분집합의 값들을 제한한다.

제약만족문제에 대한 해결책은 모든 제약들이 동시에 만족되는 방식으로, 모든 변수에 값을 할당하는 것이다.

제약의 차원(arity)은 제약이 연결되어 있는 변수의 개수이다. 예를 들면, 이진 제약은 두 개의 변수를 가진다. 임의의 이진 제약만족문제(Constraint Satisfaction Problem)의 문제 클래스는 일반적으로 $\langle n, m, c, t \rangle$ 4개의 쌍(tuple)으로 표현된다. 여기서 n 은 변수의 개수, m 은 각각의 도메인에서 값의 개수, c 는 제약의 개수, t 는 각 제약에서 허용되지 않은 쌍의 개수이다. 제약만족문제 검색 알고리즘의 표준적인 측정은 수행된 제약 체크의 횟수이지만 속행(resumption) 등과 같은 다른 속성들도 사용되어진다.

3. 제약만족에 의한 지식베이스 재사용

본 연구에서는 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법[5]을 사용하였다. 만약 합성된 제약만족 문제가 불일치하는 것으로 판단된다면 지식베이스-문제 해결자 조합은 문제 해결자 요구사항을 만족하지

못하므로 제거될 수 있다. 이 방법은 제약을 제거함에 의해서 제약만족 문제를 완화시키고 만약 완화된 버전이 해결될 수 없다면 본래의 제약 만족 문제는 해결책을 가지지 못하게 된다. 이 방법을 사용하게 되면 완화된 제약만족 문제가 본래의 제약 만족 문제보다 해결하기 쉽게 된다. 만약 하나의 문제가 있는데 몇 가지 제약이 제거될 수 있다면 새로운 문제는 본래의 문제보다 해결하기 쉽게 된다.

3.1 지식베이스 시스템의 스케줄링

본 논문에서는 급성복통과 관련된 지능형 질환 시스템 [6] 영역에서 지식베이스 시스템을 스케줄링하는 프로토타입을 구축하였다. 급성복통과 관련된 지능형 질환 시스템에서는 급성복통과 관련된 질환을 진단하기 위하여 담낭염 관련 2개의 지식베이스, 간질환 관련 3개의 지식베이스, 대장질환 관련 2개의 지식베이스, 위염 및 십이지장 관련 5개의 지식베이스, 생식 및 비뇨기계 관련 2개의 지식베이스 등 14개의 지식베이스를 가지게 된다. “수술 위험도가 0.21이고 구토정도가 0.12이며 하복부 통증정도가 0.97이고 직장 검사 정도가 0.92이고 혈변 정도가 0.98인 경우 어떤 질환으로 진단이 될 수 있는가?” 라는 질의에 대해 시스템에서는 질환에 대한 진단을 결정하고 병합하기 위해서 5개의 제약 만족자 (constraint satisfier)와 14개의 지식베이스를 사용하게 된다. 이 때 시스템에서는 체계적으로 모든 가능한 지식베이스-문제 해결자 조합을 고찰하게 된다. 이 경우 시스템에서 탐색해야 하는 가능한 지식베이스-문제해결자 조합은 120가지($2 \times 3 \times 2 \times 5 \times 2$)이다.

3.2 재사용 탐색 과정

지식베이스-문제 해결(KB-PS) 조합들을 찾기 위한 과정은 그림 1과 같이 세 단계로 나타낼 수 있다. 첫째, 앞절에서 나타난 지식베이스-문제 해결(KB-PS) 조합은 제약 해결자(constraint solver)에 의해서 복합적인 제약만족문제(CSP:Constraint Satisfaction Problem)로 변형되어진다. 이 변형에서 표준화된 지식베이스가 사용되어진다면, 자동적으로 수행되어질 수 있다. 둘째, 제약 만족 문제의 완화(relaxation) 모듈은 그 때 그 문제에 대하여 완화된 버전으로 만들기 위해서 완화 전략(strategies)에 따라서 명시된 제약을 조심스럽게 제거함에 의해서 제약만족 문제를 완화시킨다. 셋째, 태스크 명시(하나 또는 모두 혹은 최선의 해결책)를 지니는 모듈을 해결하는 것은 완화된 제약 만족 문제들을 해결하기 위해서 사용되어질 수 있다. 만약 이렇게 완화된 제약만족 문제들 중 일부가 불일치된다면 그 때 본래의 지식베이스-문제 해결 조합들은 모순이 되어 제거될 수 있다.

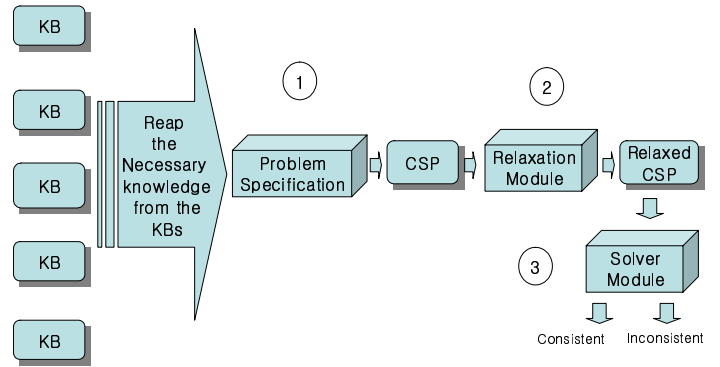


그림 1 재사용을 위한 표준화된 지식베이스 병합을 분석하는 과정

4. 실험 및 평가

본 연구에서는 기존의 급성복통과 관련된 지능형 질환 진단 시스템(A-IDS-DAAP)[6]과 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법을 적용한 급성복통과 관련된 지능형 질환 진단 시스템(CR-IDS-DAAP)에 대하여 평균 수행시간을 비교하였다. 데이터는 G 병원으로부터 획득한 350명의 실제 데이터를 최대 18개의 제약 만족자로부터 생성될 수 있는 질환명을 나타내는 7가지 test set 으로 나누어 진단을 수행하였다. 실험에 사용된 실제 데이터는 환자가 느끼는 증상들을 G 병원의 전문가인 의사들이 수치적 데이터로 변환하여 준 것을 퍼지 입력 데이터로 사용하였다. 그림 2에서 보는 바와 같이 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법을 적용한 급성복통과 관련된 지능형 질환 진단 시스템(CR-IDS-DAAP)이 기존의 급성복통과 관련된 지능형 질환 진단 시스템(A-IDS-DAAP) 보다 평균 수행시간을 감소시켜 시스템의 성능을 향상시킬 수 있었다.

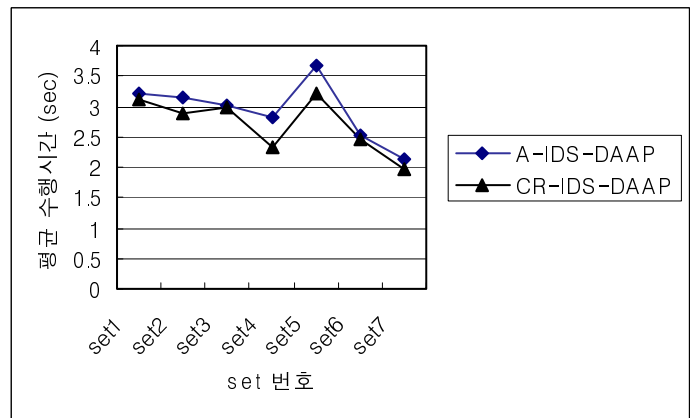


그림 2 시스템에 따른 평균 수행시간 비교

5. 결론 및 향후과제

본 연구에서는 제약을 제거함에 의해서 제약만족문제를 완화시키는 방법을 기존의 급성복통과 관련된 지능형 질환 진단 시스템에 적용하여 평균 수행시간을 감소시켜 시스템의 성능을 향상시키게 되었다. 이것은 시스템에서 가능한 조합을 인식할 때 해결될 수 없는 제약을 미리 제거함에 따라 수행시간이 감소되었기 때문이다.

본 논문에서는 지식베이스-제약만족 문제 변형을 문제 해결자에 의해서 쉽게 변형될 수 있는 표준화된 지식베이스만을 사용하였는데, 향후 연구과제로는 지식베이스에서 제약만족문제로 쉽게 변환해 주는 틀에 관한 향후 연구가 남아있다.

참고 문헌

[1] Chaudhri, V.K., M.E. Stickel, J.F. Thomer, and R. J. Waldinger, "Using Prior Knowledge: Problems and Solutions," in Seventeenth National Conference on Artificial Intelligence and Twelfth Innovative Applications of Artificial Intelligence Conference, Austin, Texas, USA:AAAI Press/MIT Press, pp.437, 2000.

[2] Dieng, R., O. Corby, A. Giboin, and M. Ribiere, "Methods and Tools for Corporate Knowledge Management," in 11th Banff Workshop on Knowledge Acquisition, Modelling and Management, pp.42, 1998.

[3] Sauer, J. and H.-J. Appelpath, "Knowledge-Based Design of Scheduling System," in World Manufacturing Congress, International Symposium on Manufacturing Systems, Auckland:ICSC Academic Press, pp.247-252, 1997.

[4] White, S. and D. Sleeman, "A Constraint-Based Approach to the Description & Detection of Fitness for Purpose," Electronic Transactions on Artificial Intelligence, Vol. 4, pp.155-183, 2000.

[5] Tomas E. Nordlander, D. Sleeman, Ken N. Brown, "Knowledge Base Reuse Through Constraint Relaxation", Proceedings of the 3rd International Conference on Knowledge Capture K-CAP '05, pp.59-66, 2005.

[6] 현우석, "급성복통과 관련된 지능형 질환 진단시스템을 위한 퍼지 규칙 생성과 이의 최적화", 한국정보처리학회논문지, Vol. 11-B, No. 7, pp.855-860, 2004. 12.

[7] Nordlander, T., "AI Surveying: Artificial Intelligence in Business", De Montfort University, Thesis, 2001.

[8] Prosser, P. and I. Buchanan, "Intelligent scheduling: past, present and future", Intelligent system engineering, Vol. 3, Issue 2, Summer, pp.67-78, 1994.

[9] Saucer, J. and R. Bruns, "Knowledge0Based Scheduling Systems in Industry and Medicine", IEEE Expert, Vol. 12, Issue 1, pp.24-31, 1997.

[10] AKT, "Reuse Knowledge", The Advanced Knowledge Technologies project, <http://www.aktors.org/publications/reuse>, 2007,

[11] Hayes-Roth, F., D.A. Waterman, and D.B. Lenat, "Building Expert Systems," Teknowledge series in knowledge engineering; Vol. 1, Reading, Mass.: Addison-Wesley, pp. 444, 1983.

[12] Bennett, J. and R. Engelmores, Experience using EMYCIN, Rule-Based Expert Systems, E. Shortliffe, Addison-Wesley, London, pp.314-328, 1983.

[13] Kalfoglou, Y., T. Menzies, E. Motta, and K.-D. Althoff, "Metaknowledge in systems design:panacea...or undelivered promise", The Knowledge Engineering Review, Vol. 15, Issue 4, pp.381-404, 2000.

[14] Uschold, M., P. Clark, M.Healy, K. Williamson, and S. Woods, "An Experiment in Ontology Reuse", 11th Banff Knowledge Acquisition Workshop, Calgary Canada: SRDG Publications, pp.33, 1998.

[15] Park, J.Y., J.H. Gennari, and M.A. Musen, "Mappings for Reuse in Knowledge-based Systems" 11th Workshop on Knowledge Acquisition, Modelling and Management, Canada:Springer, pp.1-21, 1998.

[16] Kolodner, J.L., "Case-Based Reasoning", San Mateo, CA:Morgan Kaufmann Publishers, pp.668, 1993.

[17] Gennari, J.H., M.A. Musen, R.Ferguson, "The Evolution of Protege: An Environment for Knowledge-Based Systems Development for Knowledge-Based Systems Debelopment," International Journal of Human-Computer Studies, Vol. 58, Issue 1, pp.89-123, 2003.

[18] Wallace, M., "Practical application of constraint programming", Constraints, Vol. 1, Issue 1-2, pp.139-168, 1996.

[19] ILOG Solver, Version:5.3, from ILOG Inc., <http://www.ilog.com>