소프트웨어 재사용을 위한 지식기반 정보검색 모델

조 원규, 김 영환, 김 진형

한국과학기술원 전산학과 인공지능 연구실

Knowledge Based Information Retrieval Model for Software Reuse

Wongyu Cho, Young Whan Kim, and Jin H Kim

Artificial Intelligence Lab., Dept of Computer Science, KAIST

요 약

소프트웨어 재사용 라이브러리 시스템에 이용가능한 지식기반 정보검색 모델을 제안하고 이를 바탕으로한 Common LISP 라이브러리 시스템을 개발하였다. 사용자 질의어나 소프트웨어 요소들은 정해진 색인어로써 표현되고, 사용자 질의어에서는 부울리언 오퍼레이터의 사용이 가능하다. 지식베이스는 색인어들간의 상관관계를 계층적 개념그래프로 나타내었으며 그래프상에서의 용어들 간의 최단경로의 길이를 용어들간의 개념적 거리로 간주하는 정합함수를 추론기관으로서 제안하였다.

I 서론

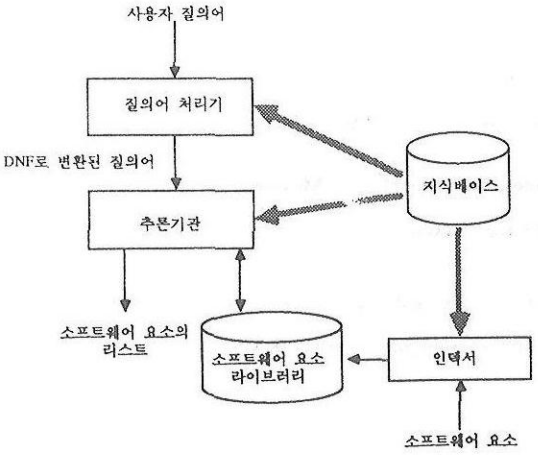
컴퓨터 기술이 점점 복잡해지고 발달되어 감에 따라 소프트웨어 시스템도 대규모화 되고 복잡해지고 있다. 반면에 소프트웨어 전문인력은 수요에 미치지 못하고 있어서 소프트웨어 생산성은 하드웨어 발달에 비해 매우 저조한 실정이다. 어떤 조사 보고에 의하면 기존의 소프트웨어 중 거의 60% 가량이 이미 작성 되었던 프로그램과 매우 비슷한 프로그램을 사용하고 있다고 한다[1]. 이러한 사실로 미루어 볼 때 이미 개발된 소프트웨어를 쉽게 재사용할 수 있게되면 소프트웨어 생산성이 상당히 증대될 것이다. 현재 소프트웨어 재사용기술은 부함수, 함수, 프로그램등과 같이 이미 개발된 재사용 가능한 코드들을 쉽게 이용하도록 하는 재사용 라이브러리에 관한 연구와 초고급 언어와 같이 목적 프로그램을 생성시키는 재사용 패턴에 관한 연구가 있다[2]. 재사용 라이브러리에 관한 연구는 필요한 것을 접근(access)하여 검색하고, 검색한 코드를 이해한 뒤, 이를 재사용 요구에 맞게 수정하여 프로그램을 개발하는 단계로 이루어진다. 이 중에서 첫번째 단계인 코드검색 단계는 정보검색 분야에서 개발된 기술들을 응용할 수 있는 부분으로서 코드의 색인, 색인어의 분류, 그리고 정합함수에 관한 연구가 필요하다.

본 논문에서는 효율적인 코드검색을 위한 지식기반 정보검색 모델을 제안한다. 기존의 정보검색 모델은 크게 부울리언 모델 벡터 모델, Fuzzy Set 모델 확장된 부울리언 모델 등으로 분류할 수 있다[3]. 이들 모델에서는 색인어의 일치에 근거하여 적합성(relevance)을 평가 하였다. 만일 아주 비슷한 색인어로 색인되어 있는 객체라 하여도 이 색인어들이 정확히 일치하지 않을 때는 그 객체들이 유사하다는 것을 구별해 낼 수 없다. 제안된 모델에서는 색인어들간에 존재하는 상관관계를 포함하는 지식베이스를 이용하여 객체들간의 개념적 거리를 계산함으로써 임의의 두 객체간의 유사도(similarity)를 수치로 나타낼 수 있는 정합함수를 설계하였다. 그리고, 이 모델을 Common LISP 재사용 라이브러리 시스템 개발에 응용하였다.

II 지식기반 정보검색 모델

본 논문에서 제안한 정보검색 모델은 그림1에 나타난 것과 같이 크게 4가지 요소로 이루어진다. 첫째로, 사용자 질의어는 프로그래머의 재사용 요구를 나타내는 것으로서, 색인어와 부울리언 오퍼레이터를 사용하여 표현된다. 둘째로 소프트웨어 라이브러리는 재사용될 소프트웨어 요소들을 저장하는 데이타베이스로서 각 소프트웨어 요소는 facet 분류[4]에 의거하여 facet 별로 지식베이스에 있는 색인어들로 색인되어 있다. 세째로 지식베이스는 정보검색 분야에서 말하는 용어사전(Thesaurus)의 일종으로서 질의어나 소프트웨어 요소들을 색인하는데 사용되는 색인어 및 이들간의 관계로 이루어진다. 네째로, 추론기관은 프로그래머의 재사용 질의어와 각 소프트웨어 요소들간의 적합도(Relevance)를 측정할 수 있는 정합함수로 구현되며 이는 지식베이스와 함께

시스템의 성능에 가장 영향을 미치는 부분이다.



<그림 1 > 시스템의 구조

2.1 사용자 질의어 (User Query)

사용자 질의어는 사용자가 쉽게, 그리고 명확하게 재사용 요구를 표현할 수 있는 방법으로 이루어져야 한다. 정보검색 분야에서 개발된 기존의 방법론은, 부울리언 오퍼레이터의 사용 유무와, 색인어에 대한 가중치의 사용 여부에 따라, 부울리언 모델, 벡터 모델, 확장된 부울리언 모델등으로 구분된다. 본 모델에서는 부울리언 오퍼레이터를 사용하고, 색인어의 가중치는 사용하지 않는 부울리언 모델을 채택하였다. 따라서, 사용자 질의어는 2.2에서 설명할 facet 분류에 의거하여, 각 facet별로 따로 구성된다. 각 facet의 표현은 색인어들과 부울리언 오퍼레이터 and, or, not으로 구성되는 부울리언 식이 된다. 이때 각 facet에서 사용될 수 있는 색인어는 미리 정해진 controlled vocabulary에서 선택되며 이 색인어들로 2.3에서 설명될 HCG가 구성된다.

2.2 소프트웨어 요소 라이브러리 (Software Component Library)

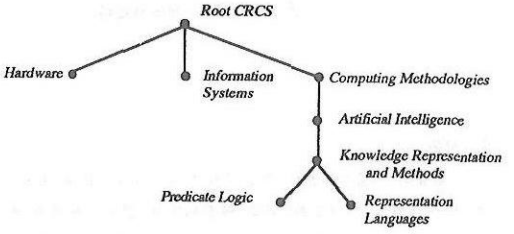
소프트웨어 시스템 전체를 다른 목적으로 그대로 사용하는 것은 불가능한 일이다. 그러나, 이러한 시스템을 구성하는 함수나 부프로그램과 같은 독립적인 소프트웨어 조각들은 약간의 수정을 통하여 다른 시스템 구성에 재사용할 수 있다. 소프트웨어 요소란 이처럼 독립적인 기능을 가지고 큰 시스템 구성의 기본 요소가 되도록 일반적으로 만들어진 소프트웨어 조각을 말한다.

소프트웨어 요소들을 목적에 맞게 재사용하기 위해서는 정해진 분류방법에 따라 분류를 해두어야한다. 소프트웨어 요소들을 분류하는 방법에 따라, 필요한 요소를 검색할 수 있는 능력이 좌우되고 라이브러리의 구성과 검색 방법이 영향을 받게 된다. 대표적인 분류방법으로는 enumerative 분류와 facet 분류를 들 수 있다[4]. Enumerative 분류는 객체들간의 관계를 표현하기에는 좋으나, 모든 가능한 class가 미리 정의되어야 하기 때문에 객체가 점차 증가하는 경우에는 많은 노력이 소요되는 단점이 있다. 반면에 facet 분류는 새로운 객체의 추가가 용이하며 분류 대상체들의 중요한 성격을 잘 표현하는 facet를 선정하면 객체들을 효과적으로 나타낼 수 있다. 이러한 점들을 고려하여, 제안된 모델에서는 facet 분류법을 수용하였다.

검색대상이 되는 소프트웨어 요소들을 분류하기 위하여 타입(type), 기능(operation), 그리고 리스프 키워드(LISP keyword)의 세가지 facet으로 나누었으며, 각 facet별로 미리 정해둔 색인어(controlled vocabulary)들을 사용하여 색인을 하였다. 이때, 이진색인(binary indexing) 방법을 택하였다. 따라서 하나의 소프트웨어 요소는 각각 세가지 facet별로 이진색인되어 있고, 각 facet별 색인은 벡터형식을 갖게 된다.

2.3 지식베이스 (Knowledge Base)

정보검색 분야에서 thesaurus는 주로 두가지 용도로 사용되어 왔다[5]. 한가지는, 사용자가 그의 검색요구를 색인어를 사용하여 정확하게 질의어로 나타낼 수 있도록 도와주는 것이다. 사용자는 thesaurus에 나타난 색인어들간의 관계나 정보를 이용하여 보다 적절한 색인어를 선택함으로써 질의어를 정확하게 작성할 수 있다. 다른 한가지 용도는 색인 전문가들이 문서를 색인할 때 문서의 내용을 보다 정확히 표현하기 위하여 thesaurus를 사용하는 것이다. 이와 같이 지금까지는 thesaurus를 이 두가지 용도로 사용하기 위한 연구가 수행되어 왔지만, thesaurus에 나타난 색인어들간의 관계를 지식으로 보고 이를 정합과정에 직접 이용하여 검색의 효율을 높이려는 시도는 없었다. 본 모델에서는 controlled vocabulary들간의 Is-a 혹은 generalization 관계를 정의하고 이를 계층적 개념그래프 (HCG: Hierarchical Concept Graph)로 나타낸 후 지식베이스로 사용하였다. HCG의 예는 그림2와 같다.



< 그림 2 > Example HCG

널리 사용되는 Thesaurus 가운데 제안된 모델에서 추구하는 구조와 유사한 것으로 MeSH와 CRCS가 있다. MeSH(Medical Subject Headings)는 미국의 NLM(National Library of Medicine)에서 제공하고 있는 MEDLINE이라고 하는 정보검색 시스템에서 사용하는 Thesaurus로서 약 15,000개의 생의학 분야 용어로 이루어진 깊이(depth) 9의 계층적 목구조이다[6]. CRCS (Computing Reviews Classification Structure)는 Association for Computing Machinery에서 출판물들을 색인하기 위한 목구조의 thesaurus이다. 이는 깊이가 5이고 약 1,000개의 용어로 구성되어 있다[7].

계층적 개념그래프는 목구조 혹은, 래티스(lattice)로 이루어진 thesaurus이다. 각 노드는 하나의 색인어를 의미하며, 노드간의 계층적 관계는 노드들을 연결하는 가지로써 표현된다. 따라서, 두 색인어간의 개념적 거리(conceptual distance)는 단순히 이들을 연결하는 최단거리 경로상에 위치하는 가지의 갯수를 셈으로써 구할 수가있다. 그러나, 그림2와 같은 그래프에서 볼때, "Predicate Logic"과 "Representation Languages"의 개념적 거리는 "Hardware"와 "Information Systems"간의 개념적 거리와 똑같이 2가 된다. 그런데 일반적인 우리의 생각에는 전자가 후자

보다 개념적으로 가까와야 한다. 이러한 문제점을 극복하기 위하여 가중치를 사용한다[8]. 다시말해서, 모든 가지에는 그 가지가 연결하는 노드간의 개념적 거리를 나타내는 수치가 매겨질 수 있다. 이경우, 두 노드간의 개념적 거리는 이들을 연결하는 최단거리 경로상에 위치하는 가지들의 가중치를 더함으로써 구해진다. 그림2의 예에서는 계층의 상위 부분에 위치하는 가지들에게 하부의 가지보다 높은 가중치를 부여하면 사람의 생각에 가까운 결과를 낼 수 있을 것이다. 이와 같이 가중치의 사용은 보다 정확한 지식 표현을 제공한다.

2.4 추론기관 (Inference Engine)

추론기관은 HCG를 지식으로 이용하여 적절한 소프트웨어 요소를 검색할 수 있도록 추론하는 부분으로서 HCG와 함께 가장 핵심을 이룬다. 추론기관을 구성하는 정합함수는 질의어와 소프트웨어 요소들간의 적합도를 구하여 준다. 소프트웨어 요소들은 이렇게 계산된 적합도에 준하여 순서가 매겨지며, 검색될 요소의 갯수, 혹은 적합도등에 관한 임계값(threshold)에 의하여 선택 여부가 결정된다.

정합함수

추론기관이 모델의 핵심이 된다면, 추론기관의 핵심을 이루는 것은 정합함수이다. 정합함수 M은 다음과 같이 정의된다.

$$M: Q \times C \rightarrow [0, \infty]$$

여기서, Q는 facet별로 부울리언 수식으로 표현된 질의어이고, C는 facet별로 색인어들로 표현된 소프트웨어 요소를 나타낸다. 계산된 결과는 질의어 Q에 대한 소프트웨어 요소 C의 개념적 거리가 된다. 이때, 결과값이 작을수록 질의어와 소프트웨어 요소가 개념적으로 가깝다는 뜻이다. 이것은 다음과 같이 정의되는 두 복합개념(compound concept)간의 개념적 거리를 구하는 함수 DISTANCE에 의하여 구하여진다[9].

$$DISTANCE(X, Y) = \frac{DIS(X, Y) + DIS(Y, X)}{2}$$

$$DIS(X, Y) = \frac{1}{m} \sum_{t_i \in X}^{m} \min_{t_j \in Y} d_{ij}$$

$$DIS(Y, X) = \frac{1}{n} \sum_{t_i \in Y}^{n} \min_{t_j \in X} d_{ij}$$

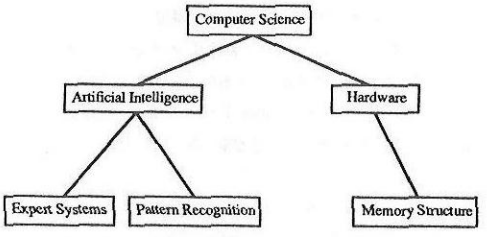
$$\text{where, } X = L_{x_1} \wedge L_{x_2} \wedge \ldots \wedge L_{x_m}$$

$$Y = L_{y_1} \wedge L_{y_2} \wedge \ldots \wedge L_{y_n}$$

$$L_{x_i} = t_{x_i}, \text{ or } \neg t_{x_i}$$

$$L_{y_j} = t_{y_j}, \text{ or } \neg t_{y_j}$$

X, Y는 부정된 색인어(not t_x 혹은 not t_y)를 포함하는 색인어들의 conjunction으로서 복합개념을 나타낸다. m, n은 각각 X, Y를 이루는 색인어의 갯수이다. 1/m, 1/n 항은 계산 결과를 정규화(normalize)하기 위하여 사용되었으며, d_{ij}는 색인어 Lx_i와 Ly_j간의 개념적 거리로서, 두 색인어가 부정되지 않은 경우에는 단순히 HCG상에서 이들간의 최단경로상에 위치하는 가지들의 가중치의 합이다. 그러나, 부정된 색인어의 경우에는 간단한 문제가 아니다. Mili 와 Rada는, 부정된 색인어를 HCG상에서 자신과 가장 멀리 떨어져 있는 색인어의 집합으로 치환함으로써 이 문제를 해결하였다[10]. 그러나, 그림3의 경우, 이들의 방법에 따르면 "(Artificial Intelligence) AND (NOT (Pattern Recognition))"이라는 복합개념은, 복합개념 "(Artificial Intelligence) AND (Expert Systems)"보다는 "(Memory Structure)"라는 개념과의 정합함수의 결과가 더 작다. 이는, 일반적으로 사용자가 부정된 색인어를 사용할 때 부정의 대상이 되는 개념을 미리 다른 색인어들을 사용하여 질의어에 표현하는데 반하여, 이들의 방법에서는 부정된 색인어를 치환할 때 사용자가 표현한 부정의 대상을 고려하지 않고 사용자가 고려하지않은 전체 색인어를 부정의 대상으로 삼아서 치환되는 집합을 구했기 때문이다. 이러한 모순을 극복하기 위하여 제안된 모델에서는 부정된 색인어가 포함된 복합개념의 의미상 문맥(context)을 고려하여 치환되는 집합을 제한한다. 즉, 사용자 질의어 중에서 부정된 색인어와 다른 색인어들간의 상호관계를 고려하여 HCG상에서 부정 색인어의 문맥을 찾아서 이 문맥내에서 치환집합을 구하였다.



<그림 3>

사용자 질의어와 소프트웨어 요소들간의 적합도는 다음과 같이 구할 수 있다. 우선 각 facet별로 부울리언 식으로 이루어진 사용자의 질의어는 DISTANCE 함수에 적용하기 위하여 DNF (Disjunctive Normal Form)으로 변환된다. 변환된 질의어는 복합개념들의 disjunction이라고 볼 수 있다. 이렇게 facet별로 구하여진 복합개념들은 DISTANCE 함수에 의해 소프트웨어 요소들의 해당 facet과의 적합도가 구해지며, 이들 중에서 가장 작은 값이 해당 facet에서의 질의어와 소프트웨어 요소간의 개념적 거리이다. Facet별로 계산된 결과를 각 facet의 가중치와 곱한 값의 합은 곧 사용자 질의어와 소프트웨어 요소간의 적합도가 되며, 이를 기준으로 하여 소프트웨어 요소를 검색한다.

III. Common LISP 라이브러리 시스템의 개발

제안된 모델은 현재 Symbolics 3650 리스프 머신에서 시제품이 완성되어 성능 평가 단계에 있다. Common LISP 표준함수는 약 700개로서 그 기능이 잘 정의되어 있고, 널리 사용되며, 또한 기계에 의존성이 적기 때문에 프로그램시 프로그래머의 생산성을 높일 수 있고 재사용효과가 뛰어나다고 판단되어 검색대상이 되는 소프트웨어 라이브러리 요소로 채택하였다. 그러나, 시험 단계가 끝나고 실용화 단계에 이르러서는 쉽게 일반 함수나 부함수, 프로그램 등으로 검색 대상의 확장이 용이하도록 시스템을 설계하였다.

정합함수는 전체 성능평가 중이다. 검색방법은 사용자 질의어와 검색대상체(문서 소프트웨어 요소등)로 이루어진 실험 데이타에 의해 전문가가 판단한 적합도 순위와 정합함수가 계산한 적합도 순위가 어느정도 일치하는가, 즉 정합함수가 전문가의 판단력과 어느정도 비슷하는가를 측정하는 것이다. 이를 위해서 Spearman's Correlation Coefficient[11]를 사용한다. 이는 두개의 순위간의 상호도를 측정하는 척도로서 -1에 가까울 수록 상이하며 1에 가까울 수록 유사한 것이다.

구현된 시스템에서는 타입, 기능 리스프 키워드, 3개의 facet가 사용되었다. 기능 facet는 소프트웨어 요소가 수행하는 기능을 표현하는 용어들로 이루어진다. 현재 176개의 용어로 구성되었으며 대부분이 동사이기 때문에 이들간의 정확한 계층적 관계를 구하는 것이 난제로 남아있다. 타입 facet는 104개의 Common LISP 타입으로 구성되어 있다. 리스프 키워드 facet는 리스프 환경에서 자주 쓰이나 리스프 타입은 아닌 36개의 용어(버퍼, 폼, 클로즈등) 로 이루어져 있다. 이상의 용어들을 사용하여 소프트웨어 라이브러리 내의 함수들이 색인되었다. 이들 중 색인하는데 있어서 세 facet 전부를 필요치 않는 경우에는 필요한 facet만을 채웠다. facet의 중요도를 표시하는 가중치는 소프트웨어 요소를 색인할 때는 모두 동등하게 1로 하였고, 사용자가 필요에 따라 질의에 포함시킬 수 있도록 하였다.

사용자 인터페이스는 사용자가 시스템을 접촉하는 창구로서 사용자가 목적을 달성하기 위하여 수행하는 일련의 작업을 통제한다. 구현된 사용자 인터페이스는 사용자가 내부를 모르더라도 손쉽고 신속하게 목적을 달성할 수 있도록 윈도우와 메뉴를 이용하여 설계되었다.

사용자는 두가지의 방법으로 질의어를 구성할 수 있다. 우선 키보드로 입력을 할 수 있다. 이때, 시스템은 사용자가 한 글자를 입력할 때마다 지금까지 입력된 글자들을 이용하여 완성시킬 수 있는 색인어들을 나열하여 준다. 다른 방법으로는 브라우저(browser)를 사용하는 것이 있다. 브라우저란, 계층적 개념 그래프를 그림으로 표현하여 사용자가 직접 지식의 구조를 탐색해 가며 적절한 용어를 선택할 수 있도록 도와 주는 기구이다. 검색된 결과도 윈도우와 아이콘을 사용하여 사용자가 쉽게 이해할 수 있도록 하였고 검색된 각각의 소프트웨어 요소에 대한 자세한 정보도 쉽게 볼 수 있게 설계하였다.

IV 결 론

본 논문에서는 지식기반 정보검색 기술을 응용한 소프트웨어 재사용 라이브러리 모델을 제시하였다. 지식베이스로는 HCG를 이용하였고 HCG상의 거리는 개념적 거리로 간주하는 DISTANCE 함수를 정합함수로 제안하였다. 완벽한 시제품의 구현을 위해서는 대상 소프트웨어의 속성에 관한 세밀한 연구와 정합함수의 성능평가를 위한 충분한 실험이 요구된다.

시스템의 성능은 HCG의 질에 크게 좌우되며 HCG의 성격에 따라 검색 결과가 달라진다. 사람은 각기 다른 성격과 경향을 지니고 있으므로 효율적인 시스템이 되려면 각 사용자에 맞게 정적으로 혹은 동적으로 HCG를 변화시킬 수 있는 사용자 모델링에 관한 연구가 필요하다.

앞으로는 소프트웨어 요소를 색인하거나 색인어간의 상관관계를 유추하는 과정을 자동적으로 수행할 수 있는 방법론의 개발, 이전의 검색결과를 이용하여 사용자 질의어를 보다 정확하게 변경하는 학습에 관한 연구 및 효율적인 사용자 인터페이스에 관한 연구가 필요하다.

참고 문헌

[1] I C Jones "Reusability in Programming A Survey of the State of the Art" *IEEE Tr on Software Eng*, Vol SE-10 No 5 Sept 1984 pp 488-494

[2] T J Biggerstaff and A J Perlis, "Foreword of special issue on Software Reusability," *IEEE Tr on Software Eng*, Vol SE-10, No 5 Sept 1984, pp 474-477

[3] G Salton E A Fox, and H Wu "Extended Boolean Information Retrieval" *Communications of the ACM*, Vol 26 No 11, Nov 1983 pp 1022-1036

[4] R Prieto-Diaz and P Freeman "Classifying Software for Reusability," *IEEE Software*, Vol 4, No 1, Jan 1987, pp 6-16

[5] G Salton and M McGill *Introduction to Modern Information Retrieval* New York McGraw-Hill Inc 1983

[6] J Backus S Davidson and R Rada "Searching for Patterns in the MeSH Vocbulary," *Bulletin of the Medical Library Association*, Vol 75, No 3 July 1987, 00 221-227

[7] J Sammet and A Ralston, "The new(1982) Computing Reviews Classification System - Final Version" *Communications of ACM*, Vol 25, No 1, Jan 1982, pp 13-25

[8] M Oaks and Piet-Hein Speel, Separation Layer in a Thesaurus Computer Science Dept Univ of Liverpool England, June 1989

[9] Young Whan Kim and Jin Hyung Kim A Model of Knowledge Based Information Retrieval with Hierarchical Concept Graph," Computer Science Dept, KAIST, CS-TR-89-42, June 1989

[10] R Rada, H Mili E Bicknell, and M Blettner, "Development and Application of a Metric on Semantic Nets," *IEEE Tr on Systems, Man and Cybernetics*, Vol 19 No 1, Jan 1989 pp 17-30

[11] M G Kendall, *Rank Correlation Methods* London and High Wycombe Charles Griffin & Company LTD, 1975