컴포넌트 검색에서 퍼지 시소러스를 이용한 효율적인 질의확장 방법

Efficient Query Expansion Method using Fuzzy Thesaurus in Component Retrieval

김귀정
건양대학교 IT학부

Gui-Jung Kim (gjkim@konyang.ac.kr)
Division of Information Technology Konyang University

한정수
천안대학교 정보통신학부

Jung-Soo Han (jshan@cheonan.ac.kr)
Division of Information & Communication Cheonan University

중심어 : 질의 확장, 컴포넌트 검색, 퍼지 시소러스, 시뮬레이션

Keyword : Query Expansion, Component Retrieval, Fuzzy Thesaurus, Simulation

요 약

본 논문은 사용자 질의가 가지는 특정 클래스로부터 개념적으로 서로 연관있는 컴포넌트를 검색하기 위하여 퍼지 시소러스를 통한 질의 확장 방법을 제안하였다. 사용자 질의는 퍼지 불리언 형태로 표현되며, 퍼지 시소러스에 의한 유의어 테이블에 의해 질의 확장된다. 시소러스에 의한 사용자 질의 확장은 용어 불일치 문제를 해결함으로써 검색에 대한 일정한 정확도를 보장하면서 재현율을 향상시킬 수 있게 한다. 질의 확장과정의 효율성을 평가하기 위하여 시뮬레이션을 통한 최적의 검색 효율을 나타내는 임계치를 설정하고 재현율과 정확도를 비교하였다.

Abstract

In this paper, we used query evaluation method through thesaurus for retrieving components having concept relation with any classes in a query. Queries are presented in Boolean and expanded by similar table. Query expansion by thesaurus is the solution of the term mismatching and it enhanced precision and recall of the components retrieval. For efficiency evaluation of query expansion, we defined most critical value through a simulation and compared precision and recall each other.

I. 서론

소프트웨어의 효율적인 재사용을 위해서는 기존 시스템으로부터 사용자의 요구사항을 만족하는 컴포넌트를 신속, 정확하게 검색하는 방법이 필요하다. 이러한 검색을 극대화하기 위해서는 사용자의 요구정보와 검색하고자하는 컴포넌트의 색인정보를 명확히 표현해야 하며, 정보요구를 만족시키는 적절한 정보들만을 탐색하고 탐색된 정보들에 대해 정보요구의 만족도에 따라 적합성을 부여하는 것이다[1]. 이를 위해 시소러스와 같은 지식베이스를 이용한 정보검색 방법들이 많이 제안되었으나, 이처럼 시소러스를 사용하는 일은 많은 검색 시간을 필요로 하며, 심지어는 검색 과정에서 원하는 정보를 찾지 못하는 경우도 발생한다. 또한 질의와 시소러스 용어 사이의 불일치 문제가 발생하여 검색의 재현율을 감소시키는 주된 원인으로 작용하기 때문에 사용자의 효과적인 검색을 저해하게 된다[1],[2].

본 논문에서는 이 단점을 해결하기 위해 퍼지 시소러스를 통한 질의 평가 방법을 이용하였다. 이 검색은 퍼지 시소러스를 이용하여 용어 불일치 문제를 해결함으로써 검색에 대한 일정한 정확도를 보장하면서 재현율을 향상시킬 수 있게 한다. 즉, 컴포넌트와 클래스 사이의 관계를 퍼지 정도로 표현한 시소러스를 이용함으로써 사용자 질의와 정확히 일치하는 컴포넌트뿐 아니라 개념적으로 서로 연관된 유사한 의미를 가지는 컴포넌트까지 검색할 수 있게 한다. 이를 위해 본 연구에서는 퍼지 불리언 형태의 질의를 사용하여 각각의 질의어들에 대해 의미적 중요성을 차등 있게 표현할 수 있도록

접수번호 : #030826-001
접수일자 : 2003년 8월 26일, 심사완료일 : 2003년 11월 18일

*교신저자 : 김귀정, e-mail : gjkim@konyang.ac.kr

하였으며, 퍼지 불리언 형태로 표현된 사용자 질의는 시소러스를 통해 질의 확장과정이 이루어지도록 하였다. 또한, 효과적인 질의 확장과 검색 노이즈의 감소를 위하여 시뮬레이션을 통한 최적의 검색 효율을 나타내는 임계치를 설정하였다.

II. 관련 연구

CRCS(Computing Reviews Classification Structure)[3]는 5단계 계층구조의 트리 형태를 가지고 1000여 개의 키워드로 구성되어 있으며 불리언 질의를 통하여 검색한다. CRCS의 단점은 분류구조를 초기에 설정하여 사용하기 때문에 고정된 구조를 가지며 새로운 특성의 컴포넌트를 추가하거나 기존의 컴포넌트를 제거할 경우, 전체적으로 시스템 구축방법을 재구성해야 하므로 확장이 어려운 단점이 있다.

퍼지 검색 시스템[4]은 문제 은행에서 연상 학습을 효과적으로 지원하기 위하여 전통적인 문서 정보 검색 방법을 문제 컨텐츠 검색에 적용하는데 목적을 둔 시스템이다. 그러나 질의로부터 개념적인 연상과정의 명확성이 모호하여 용어 정의가 어렵고, 현재 완성된 시스템이 아닌 평가과정의 제안된 시스템으로써 색인 과정은 모두 수작업을 통하여 이루어진다.

FIRMS(Fuzzy Information Retrieval and Management System)[5]는 퍼지 객체를 처리하기 위해서 개발된 시스템으로 퍼지 속성을 가진 객체의 표현방법을 제안하고 객체를 검색하는 데 그 목적이 있다. 또한 용어 사이의 관계를 표현하기 위해 시소러스를 사용하였는데, 관계의 연결정도 값을 이용하여 퍼지 시소러스로 구축되었다. 그러나 이 시스템은 유사 용어의 관계를 연결하기 위하여 용어들의 개념적인 정의를 해야하는 어려움이 있을 뿐만 아니라 전문가가 용어의 속성과 가중치를 분류해야하며, 또 다른 속성에 대한 가중치들을 집합으로 분류해야하는 단점이 있다. 따라서 불확실하고 모호한 질의어에 대한 문제점을 해결하는 장점이 있지만 객체들의 수가 증가하고 용어들의 수가 증가하면 증가할수록 검색과정이 너무 복잡하고 노이즈가 많이 검색되기 때문에 신뢰도가 떨어지는 단점이 있다.

계층적 시소러스 시스템[6]은 기존의 통계적 방법만을 사용한 시소러스에서 벗어나 객체지향 코드에서 추출한 동사와 명사를 기본으로 하여 그들 사이의 계층 관계와 문맥을 이용한 시소러스를 구축하였다. 그러나 추출된 특성에 대해 도메인 전문가의 시간과 노력이 너무 많이 요구되고, 검색 과정과 유사도 계산 과정이 복잡한 문제점을 가지고 있다. 또한 도메인 전문가에 따라 일관된 시소러스를 유지하기가 어렵다. 그리고 이 시스템은 컴포넌트의 검색이 아닌 멤버함수와 파라미터를 이용한 클래스 검색이기 때문에 실제 응용분야에 적용하기가 어려운 단점이 있다. 클래스가 증가할수록 멤버함수가 기하급수적으로 증가하기 때문에 중복된 멤버함수와 파라메터로 인하여 노이즈가 많다는 단점이 있다.

III. 컴포넌트 검색을 위한 질의 확장

1.퍼지 불리언 질의

일반적인 정보 검색 시스템에서 사용자는 정형화된 형태의 질의를 통해 자신이 요구하는 정보를 검색한다. 그러나 컴포넌트 검색에서 대부분의 사용자들은 특정 도메인에 대해 상당한 지식을 가지고 있는 전문가이기 때문에 포괄적인 의미의 검색이나 구체적인 검색을 모두 요구할 수 있다. 따라서 사용자 질의는 사용자의 요구사항을 정확히 표현할 수 있도록 설계되어야 한다. 그러나 일반적인 정보 검색 시스템은 모두 동일한 정도의 의미적 중요성을 가지는 질의에 대한 불리언 형태로만 처리된다. 본 연구에서는 퍼지 불리언 형태의 질의를 사용하여 각각의 질의어들에 대해 의미적 중요성을 차등 있게 표현할 수 있도록 하였다. 퍼지 불리언 모델은 사용자 의도에 따라 질의어 관련 정도를 부여할 수 있을 뿐 아니라, 도메인 개념들 사이의 관계를 정의한 시소러스와 쉽게 통합될 수 있다는 장점이 있다. 다음은 질의를 형성하는 질의어들에 대한 불리언 연산을 표현한 식이다. AND와 OR는 불리언 연산자이며, c는 하나의 질의어이고, α는 사용자가 질의 형성 시 입력한 퍼지 질의어 중요도이다. 포괄적인 의미로 질의를 표현하기 위해서는 퍼지 질의어 중요도인 α를 생략할 수 있는데, 이 경우에는 α=1.0으로 간주된다.

$$Q=(AND|OR)[c_i:\alpha_i]_{i=1}^{n}, \qquad 0\le\alpha\le1$$

퍼지 불리언 질의를 3 가지 형태의 질의로 표준화할 수 있다. 단순질의(mono-query), 분리질의(disjunctive-query), 그리고 결합질의(conjunctive -query)가 그것인데, 단순질의는 질의로 하나의 질의어만이 사용된 경우이며, 두 개 이상의 질의어가 있을 경우 분리질의는 OR의 역할을 하고 결합질의는 AND의 역할을 수행한다.

단순질의 : $q_\ell=[c_1:\alpha_1]$

분리질의 : $q_{\ell(\mathrm{EXP})}=OR[q]_{i=1}^{n}$

결합질의 : $Q=AND[q_{\ell(\mathrm{EXP})}]_{\ell=1}^{m}$

본 연구에서는 분리질의를 단순질의에 대한 질의확장의 결

과로 정의하였다. 즉, 하나의 단순질의에 대해서 확장된 질의는 모두 OR로 표현된다. 또한, 분리질의로 표현된 확장된 질의어들은 AND 연산을 함으로써 결합질의로 표현될 수 있다. 따라서, 모든 퍼지 불리언 질의는 단순질의의 분리질의에 대한 결합질의로 나타낼 수 있다.

2. 퍼지 시소러스에 의한 질의 확장과 검색

본 연구의 컴포넌트 검색은 퍼지 불리언 형태로 표현된 사용자 질의를 시소러스를 통해 확장함으로써 이루어진다. 먼저, 한 컴포넌트(i)에 대한 컴포넌트 색인 집합(Collection(i))은 클래스명으로 이루어져 있으며, 각 색인어는 컴포넌트에 대한 가중치를 가지고 있다. 컴포넌트 i에 대한 색인어 c의 가중치를 Wcom(i,c)= α와 같이 표현하기로 한다. 또한 사용자는 단순질의 q=[c:β]를 질의로 입력함으로써 질의어 중요도를 선택할 수 있다. 이때 컴포넌트 i가 사용자 질의 q를 만족하는 정도를 γ라 할때, 이 값은 γ=min(α,β)로 계산된다. 예를 들어 'CommonSocket' 컴포넌트가 'ToolBar : 0.65', 'Document : 0.60', 'Socket : 0.9', 'SocketFile : 0.91', 'DC : 0.87'와 같은 색인 집합을 가지고 있고, 사용자 질의 q가 [SocketFile : 0.9]와 같이 주워졌을 때, 'CommonSocket' 컴포넌트는 질의 q를 0.9=(min(0.91, 0.90)) 정도로 만족하고 있음을 뜻한다. 또한 질의 평가에 대해서 본 연구에서는 앞 절에서 설명한 것과 같이 모든 퍼지 불리언 질의는 단순질의의 분리질의에 대한 결합질의로 수행하도록 하였다. 따라서, 각 질의어(단순질의)에 대한 질의 확장은 OR로 연결된 분리질의로 이루어지며, 확장된 질의어들에 대해서 AND 연산(결합질의)이 이루어져 후보 컴포넌트를 검색할 수 있도록 하였다.

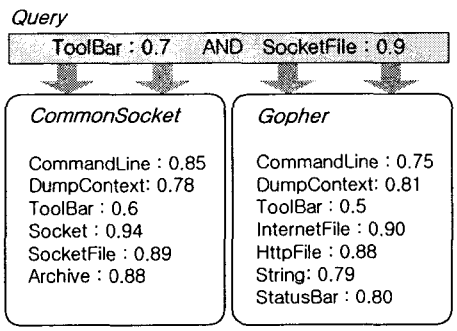

그림 1. 질의 확장과 컴포넌트 검색

그림 1은 사용자 질의가 'ToolBar : 0.7 AND SocketFile : 0.9' 일 때 두 컴포넌트 'CommonSocket'와 'Gopher'의 질의 확장 과정과 컴포넌트 검색 과정을 보여주기 위한 예이다. 두 질의 사이의 AND 연산은 확장된 질의어들에 대한 결합질의를 의미한다. 'CommonSocket' 컴포넌트의 색인 집합 Collection(CommonSocket)은 "CommandLine : 0.85, DumpContext: 0.78, ToolBar : 0.6, Socket : 0.94, SocketFile : 0.89, Archive : 0.88" 이고, 'Gopher' 컴포넌트의 색인 집합 Collection(Gopher)는 "CommandLine : 0.75, DumpContext: 0.81, ToolBar : 0.5, InternetFile : 0.90, HttpFile : 0.88, String: 0.79, StatusBar : 0.80"로 정의되어 있다.

먼저, 'CommonSocket' 컴포넌트에 대한 질의 평가와 검색 여부 평가 방법은 다음과 같다. 사용자 질의가 q1=[ToolBar : 0.7] 이고 q2=[SocketFile : 0.9] 라면, 'CommonSocket' 컴포넌트는 질의어 q1을 0.6=(min(0.6, 0.7)) 정도로 만족하고 있으며 질의어 q2를 0.89=(min(0.89, 0.9)) 정도로 만족하고 있다. 따라서, 'CommonSocket' 컴포넌트가 질의어 q1과 q2를 동시에 만족하는 정도는 min(0.6, 0.89)로 해석되므로, 질의에 대한 컴포넌트의 만족 정도는 0.6으로 평가될 수 있다. 질의에 대한 컴포넌트 만족도가 0.5 이상이면, 질의에 대해 컴포넌트가 어느 정도 연관성이 높다고 인정되므로 'CommonSocket' 컴포넌트를 검색될 후보 컴포넌트에 포함시킨다.

3. 유의어 테이블을 이용한 질의 확장

'Gopher' 컴포넌트와 같은 경우에는 의미적으로 사용자 질의와 상당히 관련이 있지만, 색인 집합 Collection(Gopher)에 'SocketFile'이 포함되어 있지 않기 때문에 질의어 q2를 0.0=(min(0.0, 0.9)) 정도로 만족하고 있다. 즉, 'Gopher' 컴포넌트가 질의어 q1과 q2를 동시에 만족하는 정도는 0.0=(min(0.5, 0.0))으로 해석되므로, 'Gopher' 컴포넌트는 검색되지 않는다. 이 단점은 검색의 재현율을 감소시키는 주된 원인으로 작용하기 때문에 사용자의 효과적인 검색을 저해하게 된다. 본 논문에서는 이 단점을 해결하기 위해 퍼지 시소러스를 통한 질의 확장 방법을 이용하였다. 퍼지 시소러스를 이용한 질의 확장은 각각의 질의어에 대해 수행되기 때문에 사용자 질의 중 하나의 단순질의 q=[c:β]에 대해 확장된 질의 집합 Exp(c)는 질의어 c 와 관련된 모든 시소러스 클래스들을 OR 연산자로 연결한 분리질의 형태로 표현될 수 있다.

표 1. 퍼지 시소러스 유의어 테이블

class명 \ class명	…	InternetFile	HttpFile	Archivet	Menu	…
…	…	…	…	…	…	…
SocketFile	…	0.79	0.73	0.65	0.4	…
…	…	…	…	…	…	…

'Gopher' 컴포넌트 검색을 위한 질의 확장 과정은 다음과 같다. 'Gopher' 컴포넌트의 색인 집합 Collection(Gopher) = {"CommandLine : 0.75, DumpContext: 0.81, ToolBar : 0.5, InternetFile : 0.90, HttpFile : 0.88, String: 0.79, StatusBar : 0.80"}에서 질의어 q2=[SocketFile : 0.9]에 대한 만족 정도는 0이기 때문에 초기 질의 "q1=[ToolBar : 0.7] AND q2=[SocketFile : 0.9]"에 의해 'Gopher' 컴포넌트는 검색될 수 없다. 그러나, 퍼지 시소러스에 의한 유의어 테이블에 의해 "q2=[SocketFile : 0.9]"는 표 1과 같이 질의 확장될 수 있다[7]. 'SocketFile' 클래스는 모든 클래스에 대한 유의값을 가지고 있으며, 이중 유의값이 0.7이상에 해당하는 클래스들만이 질의 확장의 대상이 된다. 이는 정확도를 적절히 유지하면서도 재현율이 높게 나타나는 임계치 범위를 시뮬레이션 통하여 설정한 것으로, IV. 성능평가에 자세히 설명하고 있다. 이에 따라 퍼지 시소러스에서 질의어 q2='SocketFile' 클래스의 질의 확장 집합 Exp(SocketFile)은 {SocketFile : 1.0, InternetFile : 0.79, HttpFile : 0.73}과 같이 구성될 수 있다. 여기에서 질의어 q2의 'SocketFile' 중요도가 사용자에 의해 0.9로 설정되었기 때문에 질의 확장 집합 Exp(SocketFile)의 각 확장 질의에 대한 유의값이 조절되어야 하는데, 질의어에 주워진 사용자 중요도와 각 확장 질의의 유의값 중 적은 값을 선택한다. 그러므로 질의어 q2=[SocketFile : 0.9]에 대한 최종적인 질의 확장 집합 Exp(SocketFile)은 {SocketFile : 0.9, InternetFile : 0.79, HttpFile : 0.73}과 같이 확장된다.

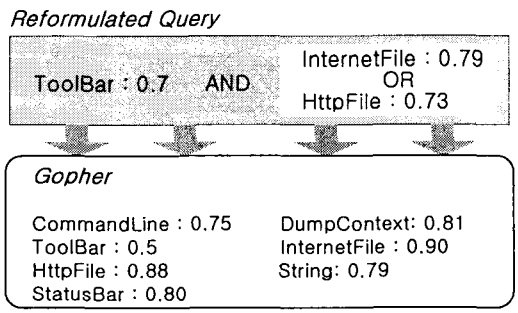

그림 2. 확장된 질의

그림 2는 이를 바탕으로 재형성된 질의를 나타내며, 'Gopher' 컴포넌트의 검색은 기본적으로 'CommonSocket' 컴포넌트의 검색 과정과 같다. 사용자 질의가 q1=[ToolBar : 0.7] 이고 q2=[InternetFile : 0.79 OR HttpFile : 0.73]이므로, 'Gopher' 컴포넌트는 질의어 q1을 0.5=(min(0.5, 0.7)) 정도로 만족하고 있다. 또한 질의 q2에 대해 확장된 각 질의 중요도와 컴포넌트 가중치 중 적은 값을 선택한 후, 퍼지 OR 연산을 수행한다. 즉, max(min('InternetFile'의 질의 중요도, 'InternetFile'의 컴포넌트 가중치), min('HttpFil'의 질의 중요도, 'HttpFil'의 컴포넌트 가중치))로 계산되므로, max(min(0.79, 0.90), min(0.73, 0.88))=0.79의 값을 얻을 수 있다. 이는 'Gopher' 컴포넌트가 질의어 q2를 0.79 정도로 만족하고 있음을 나타낸다. 따라서, 'Gopher' 컴포넌트가 질의어 q1과 q2를 동시에 만족하는 정도는 min(0.5, 0.79)로 해석되므로, 질의에 대한 컴포넌트의 만족 정도는 0.5로 평가되어 최종적으로 'Gopher'는 후보 컴포넌트에 포함되어 검색될 수 있다.

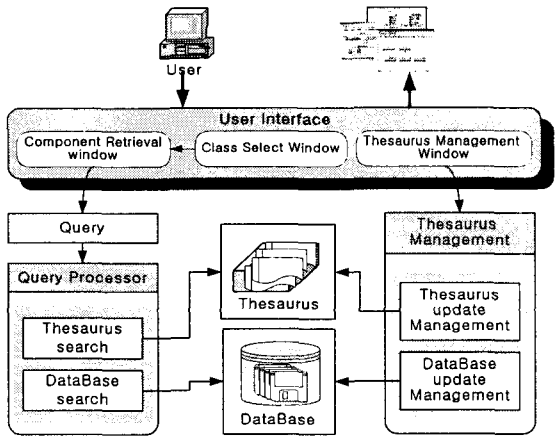

그림 3. 시소러스 검색 구조

4. 시소러스 검색 구조

그림 3은 본 연구에서 제안한 시소러스 검색 시스템 구조를 보여준다. 여기서 Query process는 입력된 질의를 확장시키는 역할을 하며, Thesaurus management는 갱신(update)기능이 있어 새로운 컴포넌트가 추가될 때마다 자동으로 시소러스를 구축할 수 있다. 또한 사용자 인터페이스는 컴포넌트 검색 뷰의 기능이 있어 검색을 위한 질의어 입력, 검색된 컴포넌트들을 우선순위로 보여줄 수 있도록 하였다. 본 연구에서는 이와 같이 질의확장을 통한 컴포넌트 검색 시스템 구축에서 질의 확장 방법에 중점을 두었다. 확장된 질의는 유사 컴포넌트까지 검색할 수 있으며 검색된 컴포넌트들을 가중치가 높은 순서로 검색 결과를 나타낼 수 있도록 설계하였다.

IV. 성능평가

1. 시뮬레이션환경

퍼지 시소러스의 성능 평가를 위한 질의 확장 과정과 질의 확장의 임계치 설정을 위한 시뮬레이션에 대해서 기술한다. 시뮬레이션에 사용된 컴포넌트는 Visual C++ Class Library로 구성하였다. 이 클래스들은 범용 라이브러리이며, 특정 범주에 치우쳐 포함되지 않으므로, 응용 개발을 위하여 일반적인 수준에서의 다양한 클래스를 제공해 줄 수 있다. 또한 이 라이브러리 내에 있는 클래스의 이름은 일반적인 명명 규칙을 따르고 있기 때문에 개발자로 하여금 쉽게 클래스를 구별할 수 있도록 해준다. 본 연구에서 사용한 시뮬레이션 환경이 표 2에 나타나 있다. 총 131개의 컴포넌트가 11개의 Class Concept Category(CCC)[7]에 포함되어 있으며, 각 CCC는 컴포넌트에 따라 최소 1개에서 최대 22개까지의 클래스로 구성되어 있다. 컴포넌트에 나타난 CCC의 평균 클래스 수는 약 8개이고, 모든 CCC에는 중복을 포함하여 총 1023개의 클래스가 존재하며 독립된 303개의 클래스로 구성되어 있다.

표 2. 시뮬레이션 환경

특징 \ CCC	All	Service	Interface	Window	Structure	Network
컴포넌트 수	131	43	37	55	42	27
CCC 내의 최소 클래스 수	1	1	1	1	1	1
CCC 내의 최대 클래스 수	22	23	19	15	22	11
평균 클래스 수	7.8	8.8	6.2	8.0	7.5	5.7
CCC 내의 전체 클래스 수	1023	377	299	442	314	154

특징 \ CCC	Exception	OLE	Frame	Processing	Graphic	Interactive
컴포넌트 수	41	23	42	32	49	18
CCC 내의최소 클래스 수	2	1	1	1	1	1
CCC 내의최대 클래스 수	11	8	9	13	22	12
평균 클래스수	5.0	3.2	4.1	6.1	12.7	6.0
CCC 내의전체 클래스 수	203	74	172	197	620	108

2. 질의 확장 임계치 평가

검색 시스템의 시소러스 효율성은 질의에 대한 효과적인 확장에 달려 있다. 본 연구에서는 컴포넌트 검색의 재현율을 최대한 보장해줄 수 있는 최적의 질의 확장 임계치를 시뮬레이션을 통하여 설정하였다. 이를 위해 시소러스 내에서 임의의 클래스 10개를 선택하여 클래스에 대한 유사 확장 집합(*similar expended sets*)을 선정하였다. 10개의 클래스에 대해 확장된 질의와 유사 확장 집합과의 비교를 통하여 정확도와 재현율을 측정하였다. 이때 질의 확장에 있어 유의값을 0.6에서부터 1.0까지 0.05의 간격으로 각각의 정확도와 재현율을 측정하였다. 정확도와 재현율 측정 방법은 다음과 같다[8].

$$\text{재현율} = \frac{\text{확장된 유의어 중 유사확장 집합에 속한 유의어의 수}}{\text{유사확장 집합의 수}}$$

$$\text{정확도} = \frac{\text{확장된 유의어 중 유사확장 집합에 속한 유의어의 수}}{\text{확장된 유의어의 수}}$$

표 3. 임계값 검색 효율

임계치	검색 효율	
	정확도	재현율
0.60	0.287	0.853
0.65	0.323	0.853
0.70	0.603	0.811
0.75	0.620	0.500
0.80	0.734	0.428
0.85	0.812	0.329
0.90	0.829	0.285
0.95	0.860	0.285
1.00	0.870	0.210

표 3은 임의로 추출한 10개 클래스에 대한 임계값 별 검색 효율을 나타낸 것이고, 그림 4는 정확도와 재현율의 평균이 임계값에 따라 보여준다. 임계값이 높아질수록 정확도가 좋아지고, 낮아질수록 재현율이 향상됨을 알 수 있다. 그러나 임계값 0.7 미만이 되면 정확도가 0에 가깝게 되어 검색 효율이 떨어지고, 임계값 0.8부터는 0.7과 정확도에 있어서는 별 차이가 나지 않지만 재현율은 상당히 떨어짐을 알 수가 있다. 따라서 본 연구에서는 정확도를 유지하면서 재현율을 최대한 보장할 수 있는 범위인 임계값 0.7 이상을 질의 확장의 범위로 설정하였다.

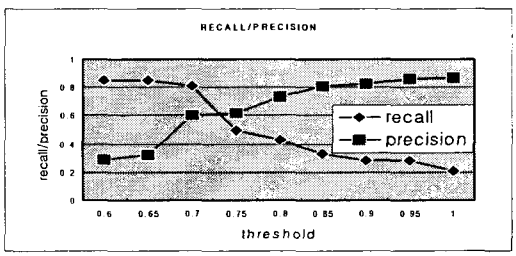

그림 4. 임계값에 따른 정확도/재현율

3. 질의 확장에 따른 성능 평가

임계값 0.7이상의 질의 확장과 이에 따른 시소러스를 성능 평가하였다. 방법은 하나의 질의에 대하여 5번의 확장을 시행한 후, 정확도와 재현율의 변화를 비교하는 것이다. 임계값

0.7 이상인 모든 클래스를 질의에 포함시키고, 새롭게 질의에 포함된 클래스에 대해서 다시 0.7 이상인 클래스를 질의에 포함시키는 방법으로 총 5번의 질의 확장을 시행한다. 이는 확장 과정에 따라 질의에 대한 재현율과 정확도의 변화를 알기 위함이다. 표 4에서 처럼 "Window"를 5번 확장한 결과 모두 8 개의 질의로 확장되었다. 10개의 클래스에 대해 시행한 결과의 정확도와 재현율 비교가 표 5와 그림 5에서 보여준다. 확장이 진행될수록 정확도가 떨어지고, 재현율이 향상됨을 알 수 있다. 그러나 확장이 한번도 이루어지지 않은 경우에는 정확도는 높지만, 재현율이 낮아 검색 효율이 떨어지며, 확장이 3번 이상 이루어진 경우에는 재현율에 있어서는 별 차이가 나지 않지만 정확도가 상당히 떨어짐을 알 수가 있다.

표 4. "Window" 클래스 확장 과정

확장 과정	질의
Q1	Window
Q2	Window View : 0.84 AnimateCtrl : 0.79
Q3	Window View FrameView : 0.93 ScrollView : 0.77 CtrlView : 0.85 AnimateCtrl
Q4	Window View FrameView CtrlView ScrollView RecordView : 0.80 AnimateCtrl
Q5	Window View FrameView CtrlView ScrollView RecordView DaoRecordView : 0.92 AnimateCtrl

표 5. 확장과정의 검색 효율

확장 과정	검색 효율	
	정확도	재현율
Q1	0.821	0.265
Q2	0.598	0.793
Q3	0.234	0.834
Q4	0.205	0.849
Q5	0.156	0.856

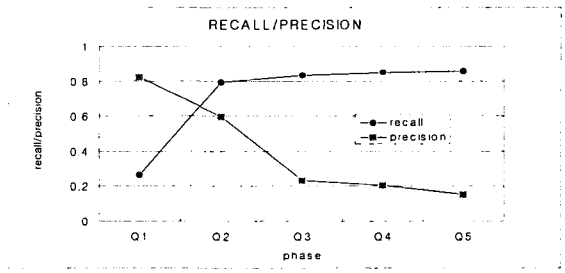

그림 5. 확장에 따른 정확도/재현율

V. 결론

본 논문은 사용자 질의가 가지는 특정 클래스로부터 개념적으로 서로 연관있는 컴포넌트를 검색하기 위하여 퍼지 시소러스를 통한 질의 평가 방법을 이용하였다. 퍼지 불리언 형태로 표현되는 사용자 질의는 퍼지 시소러스를 통해 확장되어 후보 컴포넌트가 선택된다. 이때, 질의로 사용된 클래스와 임계치 이상의 유의값을 가지는 클래스가 질의에 확장되어 포함된다. 사용자 의도에 부합하는 질의를 형성하기 위해 특정 질의를 강조할 수 있는 불리언 형태의 질의를 사용하였는데, 이는 사용자의 요구사항을 정확히 표현할 수 있도록 도와준다. 질의 확장에 의한 컴포넌트 검색은 사용자 질의와 정확히 일치하는 컴포넌트뿐 아니라 개념적으로 서로 유사한 의미를 가지는 컴포넌트까지 검색할 수 있게 하여 용어 불일치 문제를 해결함으로써 검색에 대한 일정한 정확도를 보장하면서 재현율을 향상시킬 수 있었다. 또한 검색 노이즈의 감소를 위해서 본 연구에서는 질의 확장의 임계치를 조절하여 효율성을 시뮬레이션한 결과, 유의값 0.7이상, 그리고 확장 과정 1회(Q2) 인 경우에 가장 좋은 결과를 나타냄을 알 수 있었다. 유의값 0.7이상인 경우에는 재현율과 정확도에 있어서 평균 81.1%, 60.3%이상을 보여줬으며, 질의 확장 과정이 1회인 경우에는 79.3%, 59.8%의 값을 보여줌으로써 시뮬레이션 결과 가장 좋은 임계치를 설정할 수 있었다. 따라서 본 연구는 객체지향 컴포넌트의 검색을 효율적으로 수행할 수 있는 퍼지 시소러스 기반 질의 확장 방법을 제안하였으며, 시뮬레이션을 통하여 그 유용성을 입증하였다.

참 고 문 헌

[1] R. Rada, H. Mili, E. Bickenell and M. Blettner, "Development and Application of a Metric on Semantic

Nets," IEEE Transaction on System, Mand Cybernetics Vol.19, No.1, pp.17-30. 1989.

[2] H. Chen, T. Tim and D. Fye "Automatic Thesaurus Generation for an Electronic Community System," Journal of the American Society for Information Science, Vol. 46, No. 3, pp. 175-193, 1995.

[3] ACM, "The Full Computing Reviews Classification System," ACM, New York, 1992.

[4] 최재훈, 김지숙, 조기환, "문제 은행에서 연상학습을 지원하는 퍼지 검색 시스템", 한국정보과학회 논문지, Vol. 29, No. 4, pp. 278-288. Apr. 2002.

[5] P. Subtil, N. Mouaddib and O. Foucaut, "A Fuzzy Information Retrieval and Management System and Its Applications," The Proceeding of the ACM Symposium on Applied Computing, pp. 537-541. Feb. 1996.

[6] E. Damiani, M. G. Fugini and C. Bellettini, "Aware Approach to Faceted Classification of Object-Oriented Component," ACM Transaction on Software Engineering and Methodology, Vol. 8, No. 4, pp. 425-472. Oct. 1999.

[7] 김귀정, "효율적인 컴포넌트 검색을 위한 퍼지 논리 기반 시소러스 구축", 경희대학교, 박사학위논문 pp. 129, 2003.

[8] E. Damiani, M. G. Fugini and C. Bellettini, "Aware Approach to Faceted Classification of Object-Oriented Component," ACM Transaction on Software Engineering and Methodology, Vol. 8, No. 4, pp. 425-472. Oct. 1999.

김 귀 정(Gui-Jung Kim) 정회원

1994년 2월 : 한남대학교 전자계산공학과 (공학사)
1996년 2월 : 한남대학교 전자계산공학과 (공학석사)
2003년 2월 : 경희대학교 전자계산공학과 (공학박사)
2001년 ~ 현재 : 건양대학교 컴퓨터학과 교수
<관심분야> : S/W 재사용, CASE 도구, 컴포넌트 검색

한 정 수(Jung-Soo Han) 정회원

1990년 2월 : 경희대학교 전자계산공학과(공학사)
1992년 2월 : 경희대학교 전자계산공학과(공학석사)
2000년 2월 : 경희대학교 전자계산공학과(공학박사)
2001년 ~ 현재 : 천안대학교 정보통신학부학부 교수
<관심분야> : CBD, 컴포넌트 관리, CASE 도구