사용자 피드백을 지원하는 적응형 검색 시스템

김귀정*
*건양대학교 컴퓨터학과
e-mail:gjkim@konyang.ac.kr

Adaptive Retrieval System Supporting User Feedback

Gui-Jug Kim*
*Dept of Computer Science, KonYang University

요 약

본 연구는 컴포넌트 재사용을 효과적으로 수행하기 위해 사용자 피드백을 지원할 수 있는 검색 시스템을 제안하였다. 컴포넌트 검색을 위해 퍼지 함수를 이용한 신뢰값을 사용하였으며, 사용자 집단의 요구에 능동적으로 반응할 수 있도록 퍼지 함수를 변화시켜 컴포넌트의 검색 우선순위를 변경시키는 방법을 제안하였다. 컴포넌트의 행위적 특성에 따른 검색은 응용 도메인에 따른 소프트웨어의 재사용에 매우 효과적이다. 본 연구는 후보 컴포넌트들 중 사용자가 어떤 컴포넌트를 선택하느냐에 따라 시스템이 유연하게 반응할 수 있는 적응형 검색 방법이다.

1. 서론

효율적인 소프트웨어 재사용은 재사용을 편리하게 해주는 컴포넌트 분류 방법과 검색 기술이 필요하다. 이에 따라 본 연구에서는 도큐멘트 정보로부터 구축된 컴포넌트를 분류하는 방법과 시소러스를 이용한 검색 기술을 제안하고, 후보 컴포넌트들을 사용자 의견에 따라 우선 순위로 검색할 수 있는 방법을 제안하였다.

컴포넌트는 패싯분류의 개념을 이용하여 하나 이상의 범주(CCC)로 분류하였으며, 컴포넌트 내의 각 클래스와의 가중치를 계산하였다[1]. 이 가중치는 컴포넌트를 검색하기 위하여 질의와 컴포넌트의 신뢰값을 계산하는데 사용된다. 신뢰값 계산은 시소러스에 의해 구축된 유사도와 퍼지 함수를 이용한다. 신뢰값은 컴포넌트와 질의의 일치정도를 퍼지로 표현하며, 퍼지화 함수는 사용자의 지속적이고 장기적인 컴포넌트 선택여부에 따라 그 값이 변하게 된다. 즉, 사용자 검색 환경에 따른 컴포넌트의 검색순위 재구성이 이루어지게 된다. 초기 퍼지화 함수는 삼각형 모양으로 설정하였으며, 퍼지화 함수는 사용자가 어떤 컴포넌트를 선택하느냐에 따라서 그 모양이 수정될 수 있도록 하였다[2,3]. 이 방법은 단일 질의에 대한 최적화보다는 시스템의 전반적인 향상을 목적으로 한다. 본 연구의 사용자 피드백 방법은 시스템을 장기간에 걸쳐 서서히 변화시킴으로써 사용자 집단의 요구에 검색 시스템이 적응적으로 반응할 수 있도록 하였다.

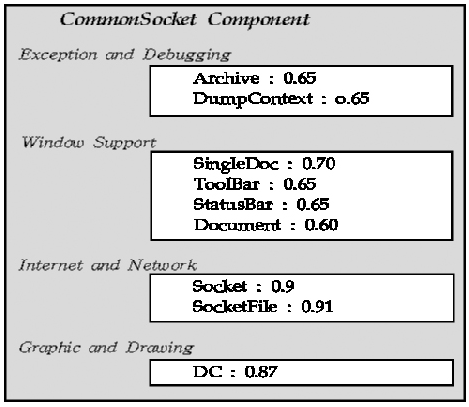
본 논문은 다음과 같이 구성된다. 2장에서는 컴포넌트 분류와 검색 모델에 대해 설명하고, 3장에서는 검색 시스템의 사용자 피드백에 대해 설명한다. 4장에서는 적응형 검색 시스템의 실험결과에 대해 설명하고, 끝으로 5장에서 결론을 맺는다.

2. 컴포넌트 분류와 검색 모델

2.1 컴포넌트 분류

본 연구에서는 패싯분류의 개념을 사용하여 컴포넌트를 하나 이상의 범주(CCC)로 분류하였다[1]. CCC는 컴포넌트와 도큐멘트 정보로부터 구축되며, 구성단위는 컴포넌트의 기능을 포괄적이고 모듈별로 제공해주는 클래스가 된다. 각 클래스는 컴포넌트와

의 가중치를 가지고 있다. 컴포넌트의 구조는 다음 그림 1과 같다.



클래스의 가중치는 컴포넌트와 클래스간의 관련 정도를 나타내는 척도이다. 한 컴포넌트의 행위와 더 관련이 깊고 유일하게 나타나는 클래스 일수록 가중치는 높은 값을 가지게 된다. i번째 컴포넌트의 k번째 클래스 가중치는 다음과 같은 식에 의해서 계산된다.

$$w_{i,k}=\frac{v_{i,k}\log(\frac{N}{n_k})}{\sqrt{\sum_{z=1}^{F}(v_{i,z}\log(\frac{N}{n_z}))^2}}$$

$w_{i,k}$: i 번째 컴포넌트에 대한 k 번째 클래스 가중치
$v_{i,k}$: i 번째 컴포넌트에 나타나는 k 번째 클래스 빈도
N : 전체 컴포넌트의 수
n_k : k 번째 특징이 나타난 컴포넌트의 수

2.2 시소러스에 의한 검색

컴포넌트를 검색하기 위하여 질의와 컴포넌트들의 유사도를 계산한다. 유사도 계산은 시소러스에 의해 구축된 유의값을 이용한다[4].

먼저 질의와 컴포넌트간의 동치관계를 계산한다. 이 식은 질의에 나타난 클래스와 컴포넌트에 있는 각 클래스 간의 유의값을 반환한다.

$$Eq(Query(u), Comp(v)) = SYNON(Query(u), Comp(v))$$

u : 질의에 있는 질의어 갯수
v : 컴포넌트에 있는 클래스 갯수

다음은 질의와 컴포넌트간의 함축관계를 계산한다. 함축관계식에 의해서, 질의에 설정한 질의 중요도가 함축관계에 의해 계산되어진 값보다 작거나 같을 경우에 질의와 컴포넌트의 각 클래스에 대한 교환이 이루어진다.

$$Imp(Query(u), Comp(v)) = \max\{w(Query(u)), w(Comp(v))\}\ [Eq(u,v)]$$

질의와 컴포넌트 클래스의 만족도를 계산한다. 만족도(satisfaction value)는 질의어와 컴포넌트의 각 클래스가 얼마나 호환성이 있는가를 말해준다.

$$Sat(Query, Comp(v)) = \frac{[\sum_{u=1}^{U} Imp(u,v)\times Eq(u,v)]}{U}$$

마지막으로 질의와 컴포넌트간의 유사도를 계산한다. 위에서 만들어진 만족집합에 컴포넌트의 가중치 벡터를 적용함으로써 최종적인 질의와 컴포넌트 간의 유사도를 계산한다.

$$Sim(Query, Comp) = Sat\times W$$

검색된 모든 후보 컴포넌트에 대해서 위와 같은 방법으로 유사도를 계산하여 우선순위 검색이 가능하도록 한다.

3. 사용자 피드백

3.1 신뢰값

사용자에 의해 주워진 질의에 대한 최적의 컴포넌트를 검색하기 위해서 본 연구에서는 신뢰값(confidence value:*CV*)을 이용하였다[4]. 신뢰값은 컴포넌트와 질의의 일치정도를 퍼지로 표현하며, 2.2에서 계산된 유사도(*Sim*)와 퍼지화 함수(defuzzification function:*D*)를 이용하여 계산되어 진다[3]. 각 컴포넌트는 제 자신 고유의 퍼지화 함수를 가지고 있으며, 사용자의 지속적이고 장기적인 컴포넌트 선택여부에 따라 함수값이 변하게 된다. 즉, 사용자 검색 환경에 따른 컴포넌트의 검색순위 재구성이 이루어지게 된다. 본 연구에서는 초기 퍼지화 함수를 삼각형 모양의 함수로 설정하였다[5]. 이는 모든 컴포넌트에 초기 함수로 설정된다. 그림 2는 초기의 삼각형 함수를 나타낸다. 그림에서 CV_1, CV_2, ..., CV_k는 유사집합(*Sim*)의 요소를 오름차순으로 정렬한 값이다. 이 퍼지화 함수를 이용하여 신뢰값(*CV*)은 다음과 같이 구해진다.

$$CV = 10\sum_{i=1}^{k} D_i CV_i$$

여기에서 10은 지나치게 작은 값이 나올 경우 그 값

을 증폭시키는 역할을 한다. 예로서 K=3이고 유사도가 Sim={0.016, 0.043, 0.05} 일 경우, CV는 다음과 같이 계산되어 진다.

$$\begin{aligned} CV &= 10\times(D_1CV_1 + D_2CV_2 + D_3CV_3) \\ &= 10\times(0.5\times0.016 + 1\times0.043 + 0.5\times0.05) \\ &= 0.76 \end{aligned}$$

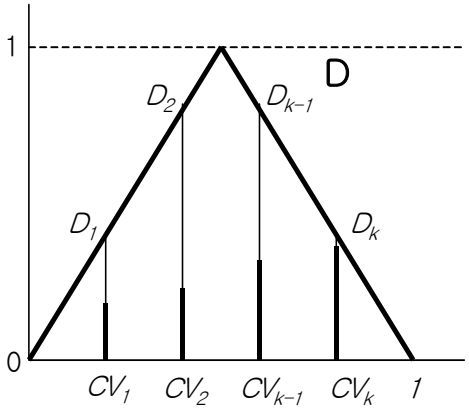


그림 2. 삼각형 퍼지화 함수 D

3.2 퍼지 함수에 의한 사용자 피드백

본 연구는 사용자 피드백에 반응할 수 있는 재사용 컴포넌트 검색 시스템을 제안한다. 사용자 피드백은 사용자 집단의 요구에 적응적으로 반응하기 위하여 시스템을 장기간에 걸쳐 서서히 변화시킴으로써 가능하다. 이 방법은 단일 질의에 대한 최적화보다는 시스템의 전반적인 향상을 목적으로 한다. 본 연구의 사용자 피드백은 클래스 가중치를 변화시키는 것이 아니라 퍼지화 함수의 모양(기울기)을 변화시킴으로써 이루어진다. 퍼지화 함수는 사용자가 어떤 컴포넌트를 선택하느냐에 따라서 그 모양이 수정될 수 있도록 하였다.

퍼지화 함수의 수정 방법은 다음과 같다. 우선순위에 따라 검색된 컴포넌트 중 사용자가 첫 번째 컴포넌트를 선택하지 않고 k번째 컴포넌트를 선택했을 때, 첫 번째부터 k번째 컴포넌트의 퍼지화 함수는 아래와 같은 식에 의해 변환된다.

$$D_{new}(t) = (1-\beta)D_{old}(t) + \beta D_{corr}(t)$$

여기에서, 첫 번째부터 (k-1)번째 컴포넌트의 경우에 $D_{corr}(t)$는 다음과 같이 수정된다.

$$\begin{aligned} D_{corr}(t) &= (1+\gamma)t - \gamma \quad , & t \le 0.5 \\ D_{corr}(t) &= 2(1+\gamma)(1-t) - \gamma \quad , & t > 0.5 \end{aligned}$$

그리고, 사용자가 선택한 k번째 컴포넌트의 $D_{corr}(t)$는 다음과 같이 수정된다.

$$\begin{aligned} D_{corr}(t) &= 2(1-\gamma)t + \gamma \quad , & t \le 0.5 \\ D_{corr}(t) &= (1-\gamma)(1-t) + \gamma \quad , & t > 0.5 \end{aligned}$$

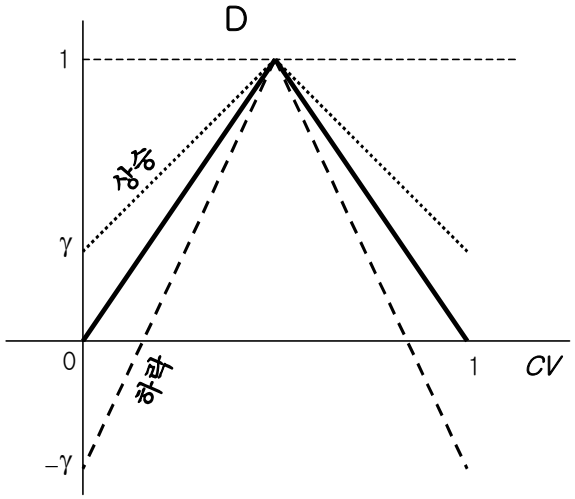


그림 3. 퍼지화 함수의 변화

그림 3은 사용자가 선택한 컴포넌트에 따라 변화하는 퍼지화 함수의 모양을 나타낸 것이다. 파라메타 γ은 애플리케이션 엔지니어에 의해 조절되며, 시스템의 변화정도를 나타낸다. γ의 값이 크면 클수록, 퍼지화 함수의 모양이 더 급격히 변하게 되어 사용자 피드백의 결과가 시스템에 더 빨리 반영되게 된다. 퍼지화 함수 $D_{new}(t)$에서의 파라메타 β는 사용자 등급에 따라 시스템 반영 정도를 다르게 해주는 역할을 한다. 함수 변화는 컴포넌트의 지속적이고 장기적인 선택의 결과로써 이루어지기 때문에 $\beta \ll 1$이며, 사용자 피드백에 의한 퍼지화 함수 모양의 수정은 매우 서서히 이루어지게 된다. 사용자 등급을 표현하는 β의 값을 정의하기 위해서 사용자 프로필을 이용하였다[5]. 이 프로필은 벡터값 $\{\alpha_1, \alpha_2, \alpha_3\}$으로 표현되고 0에서 1사이의 값을 갖는다. 각 벡터 요소는 사용자의 도메인 기술(domain skill), 작업 기술(task skill), 그리고 전략 기술(strategy skill)을 나타낸다. 이 사용자 프로필에 각 기술 가중치를 적용함으로써 사용자 등급 β값를 구할 수 있다. 다음은 파라미타 β를 구하는 식이다.

$$\beta = H\sum_{i=1}^{3} w_i\alpha_i$$

H는 $\beta \ll 1$가 되기 위한 상수값이다.

4. 실험결과

본 연구는 사용자 피드백에 의해 컴포넌트가 적응적으로 검색되는 적응형 검색시스템을 제안하였다. 이를 위해 "Document" 컴포넌트, "OleDocument" 컴포넌트, "ClientDocument" 컴포넌트에 대하여 같은 질의를 100번 수행하고 그 CV값의 변화를 실험하였다. 사용자 등급은 매니저, 전문가, 초급자로 나누었으며, 각 등급에 따른 기술 벡터값 $\{\alpha_1, \alpha_2, \alpha_3\}$과 이에 따른 β는 표 1과 같다. 벡터에 적용된 가중치는 $w=\{0.34, 0.23, 0.42\}$로 설정하였고, H=1이다[5]. 사용자 등급이 높을수록, β값이 커져 시스템 반영정도가 높아짐을 알 수 있다.

사용자등급 / 파라메타	매니저	전문가	초급자
사용자기술값 α	$\alpha_1=0.8$ $\alpha_2=0.1$ $\alpha_3=0.8$	$\alpha_1=0.8$ $\alpha_2=0.6$ $\alpha_3=0.2$	$\alpha_1=0.1$ $\alpha_2=0.8$ $\alpha_3=0.1$
가중치 w	$w_1=0.34$ $w_2=0.23$ $w_3=0.42$	$w_1=0.34$ $w_2=0.23$ $w_3=0.42$	$w_1=0.34$ $w_2=0.23$ $w_3=0.42$
H	1	1	1
β	0.631	0.494	0.260

표 1. 사용자 등급에 따른 파라메타 값

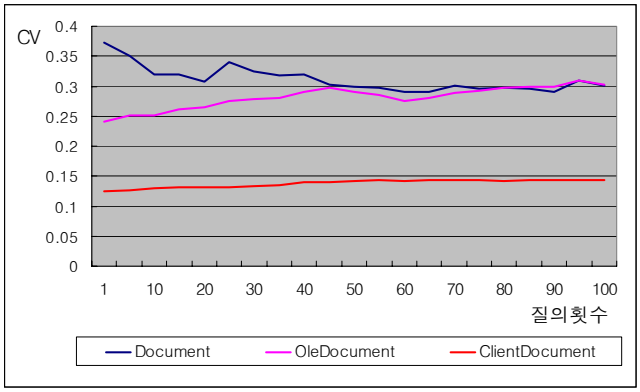


그림 4. 사용자 피드백에 따른 CV의 변화

실험에서 100번 질의 중 38번은 매니저에 의해 수행되었으며, 30번은 전문가에 의해 수행되었고 32번은 초급자에 의해 수행되었다. 질의에 대해 사용자에 의해 선택된 컴포넌트의 비율은 각각 "Document" 컴포넌트가 10%, "OleDocument" 컴포넌트가 80%, 그리고 "ClientDocument" 컴포넌트가 10%를 차지하였다. 그림 4는 각 질의에 대한 컴포넌트 선택에 따른 CV의 변화를 나타낸 것이다. 사용자가 "OleDocument" 컴포넌트를 많이 선택함에 따라 "OleDocument" 컴포넌트의 CV값이 점차로 증가함을 볼 수 있다. 이는 초기 CV값은 "Document" 컴포넌트가 더 높지만, 사용자가 "OleDocument" 컴포넌트를 많이 선택함으로 인해 검색 우선순위가 더 높아짐을 의미한다.

5. 결론

본 연구는 사용자 피드백을 지원할 수 있는 적응형 재사용 컴포넌트 검색 시스템을 제안하였다. 도큐멘트 정보를 이용하여 컴포넌트를 범주(CCC)로 분류하였으며, 시소러스에 의해 구축된 유사도와 퍼지 함수를 이용하여 신뢰값을 계산하고 컴포넌트를 검색하였다. 퍼지 함수는 사용자 요구에 맞는 컴포넌트를 우선적으로 검색해 줄 수 있도록 시스템을 적응적으로 변화시키는 역할을 수행한다. 이에 본 시스템은 사용자 피드백을 이용하여 사용자 집단의 요구에 능동적으로 반응할 수 있는 효율적인 검색 시스템이다.

참고문헌

[1] 김귀정, 한정수, 송영재, "컴포넌트 검색을 지원하는 퍼지 기반 시소러스 구축," 한국정보처리학회 논문지, 제10-D권 제5호, pp. 753-762, 8. 2003.

[2] E. Damini, M.G.Fugini," Fuzzy Techniques for Software Reuse", In Proceedings of ACM SIG-APP Conference on Applied Computing, Feb. 1996, 552-557.

[3] E. Damini, M.G.Fugini," Automatic thesaurus construction supporting Fuzzy Retrieval of Reusable Components", In Proceedings of ACM SIG-APP Conference on Applied Computing, Feb. 1995, 542-547.

[4] A. M. Zaremski, J. M. Wing, "Signature Matching: A Tool for Using Software Libraries," ACM Transaction Software Engineering and Methodology, Vol. 4, No. 2, Apr. 1995. 146-170.

[5] E. Damini, M.G.Fugini, C. Bellettini, "A Hierarchy-Aware Approach to Faceted Classification of Object-Oriented Components", The ACM Transaction on Software Engineering and Methodology, Vol.8, No.4, Oct. 1999, 425-472.