

컴포넌트 검색에서 가우시안 함수를 이용한 사용자 피드백의 개선

한정수*, 김귀정**

*천안대학교 정보통신학부, **건양대학교 의공학과

e-mail : jshan@cheonan.ac.kr

Improvement of User Feedback using Gaussian Function in Component Retrieval

Jung-Soo Han*, Gui-Jug Kim**

*Division of Info. & Comm. Cheonan University

**Dept of Bio Medical Engineering, KonYang University

요 약

본 논문은 컴포넌트 검색 시스템의 성능을 향상시키기 위해 사용자 피드백을 효율적으로 수행하는 방법을 제안하고자 한다. 기존의 퍼지 기법이 적용된 삼각형 모양의 퍼지화 함수는 컴포넌트를 선택할 때마다 매번 4가지 경우의 그래프를 재구성해야 하는 어려움이 있다. 본 연구에서는 이러한 피드백의 단점을 극복하기 위하여 검색된 컴포넌트의 선택여부에 따라 동일한 함수이지만 학습률을 달리할 수 있는 가우시안 함수를 이용한 상호작용 함수를 제안한다. 가우시안 함수를 피드백 함수로 채택 시 함수의 파라미터에 따른 검색 성능을 비교하고, 이를 토대로 가장 효율적인 동적 상호작용 함수를 제안하여 소수의 컴포넌트로도 좋은 검색 결과가 가능한 검색 시스템을 구축하고자 한다.

1. 서론

본 논문은 컴포넌트의 효율적인 재사용을 위해 후보 컴포넌트들을 사용자 의견에 따라 우선순위로 검색할 수 있는 방법을 제안하였다. 사용자 피드백은 사용자 집단의 요구에 적응적으로 반응하기 위하여 시스템을 변화시킨다. 단순히 검색된 결과물의 사용자 선택횟수에 따라 우선순위를 정하거나[1], 특정 피드백 함수를 이용하여 시스템을 장기간에 걸쳐 서서히 변화시킴으로써 가능하다[2]. 전자는 웹 검색과 일반적인 문서 검색 등에서 가장 흔히 볼 수 있는 방법이며, 신뢰도나 유의값을 이용한 시소러스의 검색에서는 후자의 방법을 이용하여야 한다. 이에 따라 사용자 피드백이 효율적으로 이루어지는 검색 시스템을 구축하기 위해서는 피드백 함수의 사용이 무엇보다도 중요하다.

기존에 사용된 퍼지화 함수를 이용한 방법은 몇 가지 문제점을 가지고 있다[2,3]. 첫째, 파라미터의 선택에 따라 함수의 성능이 크게 좌우된다. 예를 들어 파라미터에 따른 함수 선택에 있어서 초기 함수

를 삼각형 모양으로 설정하면, 컴포넌트의 신뢰값이 증가하기 매우 어려운 경우가 발생한다. 초기 계산된 컴포넌트의 신뢰값이 사용자 피드백에 의해 수정이 어려워지게 된다. 즉, 파라미터들을 임의로 초기화 시켜줘야 하고, 그 값에 따라 시스템의 성능에 커다란 영향을 미친다. 둘째, 시스템의 크기가 어느 정도 이상 커야하며(최소 컴포넌트 130개 이상) 100회 이상의 시행이 이루어 져야 안정적인 결과를 기대할 수 있다. 셋째, 컴포넌트를 선택할 때마다 매번 4가지 경우의 그래프를 재구성해야 하는 어려움이 있다. 그러므로 가중치나 파라미터의 임의 설정을 최소화하고, 구축 절차를 간소화하여 사용자와 시스템 간에 적응적·동적으로 상호작용 할 수 있는 피드백 함수의 생성이 매우 요구된다.

본 논문에서는 위에서 열거한 기존의 피드백 단점을 극복하기 위하여 검색된 컴포넌트의 선택여부에 따라 동일한 함수이지만 학습률을 달리할 수 있는 가우시안 함수를 이용한 동적 상호작용 함수를 제안한다. 본 논문에서는 가우시안 함수를 피드백 함수

로 채택 시 함수의 파라메타에 따른 검색 성능을 비교하고, 이를 토대로 가장 효율적인 동적 상호작용 함수를 제안하여 소수의 컴포넌트로도 좋은 검색 결과가 가능한 검색 시스템을 구축하고자 한다.

2. 기존 연구

퍼지화 함수는 사용자가 어떤 컴포넌트를 선택하느냐에 따라서 그 모양이 수정될 수 있다. 각 컴포넌트는 제 자신 고유의 퍼지화 함수를 가지고 있으며, 사용자의 지속적이고 장기적인 컴포넌트 선택여부에 따라 함수값이 변하게 된다[2,3]. 즉, 사용자 검색 환경에 따른 컴포넌트의 검색순위 재구성이 이루어지게 된다.

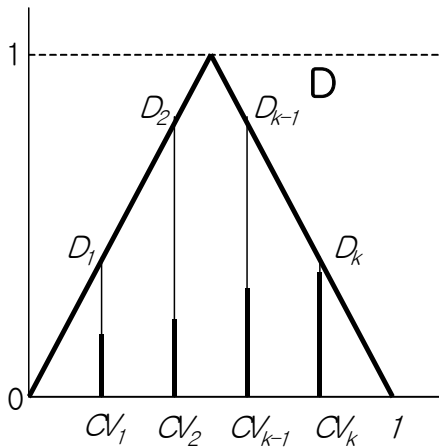


그림 1. 삼각형 퍼지화 함수 D

그림 1은 사용자 피드백이 이루어질 때 일반적으로 사용되는 퍼지화 수정방법인 삼각형 함수를 나타낸다. 그림에서 CV₁, CV₂, ..., CV_k는 유사집합(Sim)의 요소를 오름차순으로 정렬한 값이다. 유사도는 질의와 클래스의 만족도에 컴포넌트의 가중치 벡터를 적용함으로써 최종적인 질의와 컴포넌트 간의 유사정도를 계산한 것이다[4,5]. 이와 같은 퍼지화 함수를 이용하여 신뢰값(CV)은 다음과 같이 구해진다. 여기에서 10은 지나치게 작은 값이 나올 경우 그 값을 증폭시키는 역할을 한다.

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

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.

$$D_{corr}(t) = (1 + \gamma)t - \gamma, \quad t \leq 0.5$$

$$D_{corr}(t) = 2(1 + \gamma)(1 - t) - \gamma, \quad t > 0.5$$

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

$$D_{corr}(t) = 2(1 - \gamma)t + \gamma, \quad t \leq 0.5$$

$$D_{corr}(t) = (1 - \gamma)(1 - t) + \gamma, \quad t > 0.5$$

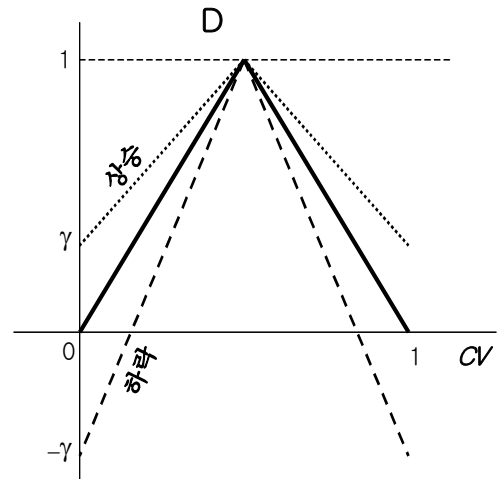


그림 2. 퍼지화 함수의 변화

그림 2는 사용자가 선택한 컴포넌트에 따라 변화하는 퍼지화 함수의 모양을 나타낸 것이다. 파라메타 γ 은 애플리케이션 엔지니어에 의해 조절되며, 시스템의 변화정도를 나타낸다. γ 의 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 β 는 사용자 등급에 따라 시스템 반영 정도를 다르게 해주는 역할을 한다. 그러나 γ 에 따라 값의 변동이 너무 크게 나타나고, 컴포넌트를 선택할 때마다 매번

4가지 경우의 그래프를 재구성해야 하는 어려움이 있다.

3. 가우시안 함수를 이용한 피드백

본 연구는 사용자 집단의 요구에 동적으로 반응할 수 있는 피드백 함수를 제안하고자 한다. 본 연구에서 제안하는 피드백은 학습 과정을 거쳐 시스템의 컴포넌트 신뢰도를 수정하는 방법이며, 단일 질의에 대한 최적화보다는 검색 시스템의 전반적인 향상을 목적으로 한다.

본 연구에서는 기존의 피드백 단점을 극복하기 위하여 검색된 컴포넌트의 선택여부에 따라 동일한 함수이지만 학습률을 달리할 수 있는 가우시안 함수를 이용한 동적 상호작용 함수를 제안한다.

가우시안 함수를 피드백 함수로 채택한 이유는 다음 세 가지의 장점을 가우시안 함수가 가지고 있기 때문이다.

- 좌우 대칭인 함수를 이용함으로써 매번 여러 가지의 그래프를 재구성하는 번거로움을 피할 수 있다.
- 1부터 0 사이의 값으로 한정지을 수 있다.
- 최대값(1)과 최소값(0)은 변하지 않으면서 폭을 조절함으로써 서로 다른 값을 생성할 수 있다.

본 연구의 사용자 피드백은 클래스 가중치를 변화시키는 것이 아니라 함수의 모양(기울기)을 변화시킴으로써 이루어진다. 즉, 기존의 방법은 경우에 따라 매번 다른 함수를 이용하거나, 가중치를 직접 변경시키는 방법이지만, 제안한 방법은 하나의 함수를 이용하여 다양한 값을 생성할 수 있다.

우선순위에 따라 검색된 컴포넌트 중 사용자가 첫 번째 컴포넌트를 선택하지 않고 k번째 컴포넌트를 선택했을 때, 첫 번째부터 k번째 컴포넌트에 피드백 함수를 적용시켜 컴포넌트의 신뢰값을 수정한다. 상호작용 함수로 사용되는 가우시안 함수의 폭을 감소·증가시킴으로써 각기 다른 학습률을 컴포넌트에 적용시킬 수 있도록 하였다. 첫 번째부터 (k-1)번째 컴포넌트의 경우에는 가우시안 함수의 폭을 줄여 기존의 값보다 작은 값을 부여하고, 선택된 k번째 컴포넌트의 경우에는 가우시안 함수의 폭을 늘려 더 높은 값을 부여한다. 제안한 가우시안 함수는 컴포넌트 선택여부에 따라 폭이 조절되는 동적 성질과 각기 다른 학습률을 갖게 되는 상호작용 함수로써

이용된다.

본 연구에서 제안한 피드백 함수식은 다음과 같다.

$$D_{corr}(t) = e^{-\frac{(H+(-1)^n \frac{C_i}{T}) \cdot t^2}{2}}$$

T : 누적 검색 횟수

C_i : i 번째 컴포넌트가 검색된 횟수

H : 함수의 폭

n : 0 or 1

파라메타 H 에 의해 함수의 폭이 결정된다. H 가 커질수록 폭은 작아지고, $D_{corr}(t)$ 값은 감소하며, H 가 작아질수록 폭은 커지고, $D_{corr}(t)$ 값은 증가한다. 이 식은 가우시안 함수의 폭 H 를 감소·증대시켜 각기 다른 학습률을 컴포넌트에 적용시키기 위함이다. 사용자가 k번째 컴포넌트를 선택했을 때, 첫 번째부터 (k-1)번째 컴포넌트에는 기존의 값보다 작은 값을 부여하기 위해 $n=0$ 을 설정하여 폭을 감소시킨다. 또한 선택된 k번째 컴포넌트의 경우에는 더 높은 신뢰값을 얻기 위하여 $n=1$ 을 설정하여 폭을 증대시킨다.

가우시안 함수는 파라메타 H 에 따라 성능에 커다란 차이를 보인다. 본 연구에서는 실험을 통해 파라메타에 따른 성능을 비교·분석하고, 이를 토대로 가장 효율적인 동적 상호작용 함수를 생성하였다. 그림 3은 사용자가 선택한 컴포넌트에 따라 변화하는 가우시안 함수의 모양을 나타낸 것이다. 예를 들어 $H=7$ 로 했을 때, T 와 C_i 에 의해 변화하는 함수의 모양이다.

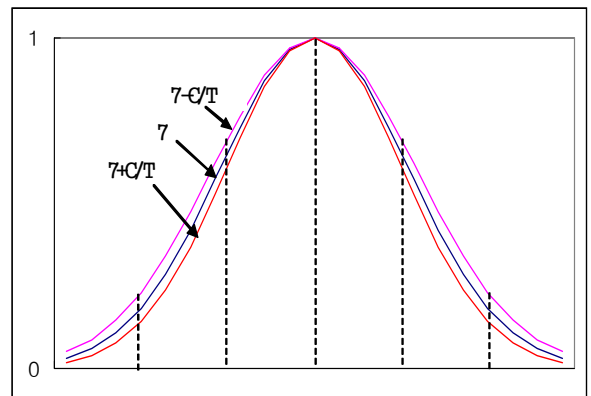


그림 3. 가우시안 피드백 함수

그림 4는 가우시안 함수를 피드백 함수로 이용했을 때 컴포넌트의 신뢰값 수정방법을 보여준다. 그림에서 CV_1, CV_2, \dots, CV_k 는 유의값의 요소를 오름차순으로 정렬한 값이다. 유의값의 가장 큰 값과 가장 작은 값의 영향을 최소화하고 함수값을 균형있게 적용하기 위함이다. 이는 좌우대칭을 이루는 가우시안 함수를 사용함으로써 그 효과를 증대시킬 수 있다. 최종적으로, 그림 4에서 얻어진 값을 이용하여 신뢰값을 수정할 수 있다.

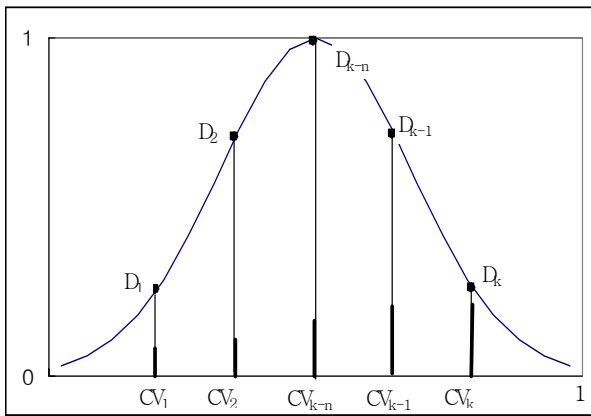


그림 4. 피드백 함수를 이용한 신뢰값 수정

이와 같이 제안한 가우시안 함수는 컴포넌트 선택 여부에 따라 폭이 자동 조절되는 동적 성질과 사용자 피드백이 가능한 상호작용 함수로써, 가우시안 함수를 피드백 함수로 채택 시 매우 효율적인 검색 시스템을 구축될 수 있을 것이다.

4. 결론

본 논문은 후보 컴포넌트들을 사용자 의견에 따라 우선순위로 검색할 수 있는 방법을 제안하였다. 피드백 함수로 가우시안 함수를 이용하는데, 가우시안 함수의 폭을 감소·증가시킴으로써 각기 다른 학습률을 컴포넌트에 적용시킬 수 있도록 하였다. 사용자가 k 번째 컴포넌트를 선택했을 경우, 첫 번째부터 $(k-1)$ 번째 컴포넌트에는 가우시안 함수의 폭을 줄여 기존의 값보다 작은 값을 부여한다. 그리고 선택된 k 번째 컴포넌트의 경우에는 가우시안 함수의 폭을 늘여 더 높은 값을 부여하도록 하였다. 함수의 폭 조절은 누적 검색횟수와 해당 컴포넌트를 선택한 횟수에 의해 자동 계산된다. 제안한 가우시안 함수는 컴포넌트 선택여부에 따라 폭이 조절되는 동적 성질과 각기 다른 학습률을 갖게 되는 상호작용 함수로써 이용된다.

이와 같이 제안한 가우시안 함수는 사용자와의 동적 상호작용 함수로써, 가우시안 함수를 피드백 함수로 채택 시 매우 효율적인 검색 시스템을 구축할 수 있으리라 기대된다. 향후에는 제안한 피드백 함수를 사용하여 소수의 컴포넌트로도 좋은 검색 결과가 가능한 검색 시스템을 구축하고자 한다.

참고문헌

- [1] T.Kohonen, "The Self-Organizing Map," Proceeding of the IEEE, Vol.78, No.9, pp.1464-1480, 1990.
- [2] E. Damini, M.G.Fugini, "Fuzzy Techniques for Software Reuse", In Proceedings of ACM SIG-APP Conference on Applied Computing, pp.552-557, Feb. 1996.
- [3] 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, pp.425-472, Oct., 1999.
- [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. pp.146-170, Apr., 1995.
- [5] 김귀정, 한정수, 송영재, "컴포넌트 검색을 지원하는 퍼지 기반 시소러스 구축," 한국정보처리학회 논문지, 제10-D권 제5호, pp.753-762, 8, 2003.