

# 유비쿼터스 환경에서 상황 인지 정보를 이용한 적응형 추천 서비스 기법

## (An Adaptive Recommendation Service Scheme Using Context-Aware Information in Ubiquitous Environment)

최 정 환 <sup>†</sup>      류 상 현 <sup>\*\*</sup>      장 현 수 <sup>\*\*</sup>      엄 영 익 <sup>\*\*\*</sup>  
(Jung Hwan Choi)    (Sanghyun Ryu)    (Hyunsu Jang)    (Young Ik Eom)

**요약** 최근 유비쿼터스 시대의 도래와 함께 개인화된 서비스를 제공하기 위한 다양한 서비스 모델들이 제안되어 왔으며, 특히, 사용자에게 개인화된 서비스를 선용적으로 제공하기 위한 다양한 추천 서비스 기법들이 고안되었다. 그러나, 기존의 기법들은 수 많은 데이터를 여과 과정 없이 분석함으로써 추천의 효율성이 떨어지며, 한정된 상황 인지 정보만을 추천 요소로 고려하기 때문에 사용자에게 개인화된 서비스를 제공하기에 적합하지 않다. 본 논문에서는 유비쿼터스 환경에서 사용자의 현재 상황에 가장 적합한 서비스를 제공하는 적응형 추천 서비스 기법을 제안한다. 본 기법은 사용자의 선호도 예측을 위해 누적된 사용자와 장치 간의 상호작용 상황 정보들을 이용하며, 군집 및 협업 필터링 기법을 이용하여 사용자에게 현재 상황에 적응적인 서비스를 추천한다. 군집 기법을 통해 사용자의 현재 위치에 근접한 데이터만을 분석함으로써, 추천의 효율성을 높이며, 협업 필터링을 이용하여 누적된 정보들이 충분하지 않은 상황에서도 정확한 추천을 보장한다. 끝으로, 시뮬레이션을 통해 본 기법의 성능 및 신뢰성을 평가한다.

**키워드** : 추천자 시스템, 협업 필터링, 군집 기법, 상황 인지, 유비쿼터스 컴퓨팅

**Abstract** With the emergence of ubiquitous computing era, various models for providing personalized service have been proposed, and, especially, several recommendation service schemes have been proposed to give tailored services to users proactively. However, the previous recommendation service schemes utilize a wide range of data without any filtering and consider the limited context-aware information to predict user preferences so that they are not adequate to provide personalized service to users. In this paper, we propose an adaptive recommendation service scheme which proactively provides suitable services based on the current context. We use accumulated interaction contexts (IC) between users and devices for predicting the user's preferences and recommend adaptive service based on the current context by utilizing clustering and collaborative filtering. The clustering algorithm improves efficiency of the recommendation service by focusing and analyzing the data that is collected from the locations nearby the users. Collaborative filtering guarantees an accurate recommendation, even when the data is insufficient. Finally, we evaluate the performance and the reliability of the proposed scheme by simulations.

**Key words** : Recommender System, Collaborative Filtering, Clustering Algorithm, Context Awareness, Ubiquitous Computing

· 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음(NIPA-2010-(C1090-1021-0008))

논문접수 : 2008년 10월 2일  
심사완료 : 2010년 1월 4일

† 정 회 원 : 성균관대학교 휴대폰학과  
themars@ece.skku.ac.kr  
\*\* 학생회원 : 성균관대학교 전자전기컴퓨터공학과  
shryu@ece.skku.ac.kr  
jhs4071@ece.skku.ac.kr  
\*\*\* 종신회원 : 성균관대학교 정보통신공학부 교수  
yieom@ece.skku.ac.kr

Copyright©2010 한국정보과학회: 개인 목적이거나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.  
정보과학회논문지: 소프트웨어 및 응용 제37권 제3호(2010.3)

## 1. 서론

지금까지 추천 서비스는 전자상거래 환경 혹은 웹 환경에서 사용자의 선호도를 예측하여 사용자가 구매할 만한 상품이나, 사용자가 선호하는 뉴스, 음악 등의 정보를 추천해주는 서비스로 널리 이용되어 왔다. 개인화라는 관점에 비추어 볼 때, 고객의 필요와 욕구에 밀착된 서비스를 제공하는 추천 서비스는 유비쿼터스 컴퓨팅 환경에서 점차 필수적인 요소로 자리매김 하고 있다 [1-3]. 개인화된 서비스에 대한 요구의 증대와 함께, 여러 연구기관에서는 개인에게 편의를 제공하는 시스템들에 대한 연구가 활발히 진행 중이다[4-12]. 특히, 스마트 환경 내에서 수집되는 상황 인지 정보들을 활용하여 사용자에게 개인화된 서비스를 선용적으로 제공하는 다양한 추천자 시스템들이 제안되었다. 그 중 COMPASS(Context-aware Mobile Personal ASSistant)[13]와 CACFS(Context-Aware Collaborative Filtering System)[14]는 유비쿼터스 환경에서 개인화된 서비스를 제공하는 대표적인 추천자 시스템으로써 사용자 정보, 주변 환경 정보 등의 상황 정보를 고려하여 사용자에게 개인화된 서비스 제공을 보장한다. 그러나, 이러한 추천자 시스템들은 추천을 위한 선호도 연산 시 수많은 데이터들을 여과 과정 없이 분석함으로써 추천의 정확성이 떨어지고, 효율적이지 못한 단점을 가진다. 또한, 사용자의 위치 정보와 같은 특정한 상황 정보만을 추천을 위한 요소로 고려하기 때문에 개인화된 맞춤형 서비스를 제공하기엔 적합하지 않다.

본 논문에서는 유비쿼터스 환경에서 사용자와 장치 간의 상호작용 상황 정보들을 추적하고, 이를 바탕으로 사용자에게 현재 위치에서 가장 적합한 장치 및 서비스를 선용적으로 추천하는 적응형 추천 서비스 기법을 제안한다. 사용자와 장치 간의 상호작용 상황 정보들은 프로파일 내에 누적되며, 이 정보들은 군집 기법 및 협업 필터링을 이용하여 사용자에게 현재 상황에 적응적인 서비스를 추천한다. 본 기법에서는 기존의 K-means 군집 기법[15]보다 성능 면에서 뛰어난 DSA K-means 군집 기법을 새롭게 제안하여 적용하였으며, 이를 통해 모든 데이터를 분석하지 않고 사용자가 위치한 군집 내의 데이터만을 분석함으로써 추천의 정확성 및 효율성을 증대시킨다. 또한, 본인과 가장 유사성을 갖는 사용자의 데이터를 활용하는 협업 필터링[16-18]을 적용하여, 데이터가 불충분한 상황에서 부정확한 추천을 야기할 수 있는 문제점을 해결한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 추천자 시스템에 대해 알아보고, 3장에서는 본 논문에서 제안하는 적응형 추천 서비스 구조 및 추천 알고리즘에

대해 설명한다. 4장에서는 시뮬레이션을 통해 본 알고리즘의 성능 및 신뢰성을 평가하고, 5장을 통해 결론을 맺는다.

## 2. 유비쿼터스 환경의 추천자 시스템

본 장에서는 유비쿼터스 환경에 적합하게 설계된 개인화된 추천 서비스를 제공하는 추천자 시스템인 COMPASS(Context-aware Mobile Personal ASSistant) [13]와 CACFS(Context-Aware Collaborative Filtering System)[14]에 대해 설명한다.

COMPASS는 2004년 네덜란드 델레마티카 연구소의 M. V. Setten에 의해 고안되었으며, 사용자의 선호도와 현재 주변 컨텍스트 정보들을 고려하여 사용자의 요구에 부응하는 서비스를 제공하는 시스템이다. COMPASS에서는 사용자의 위치 컨텍스트를 서비스 제공의 주요 요소로 활용한다. COMPASS를 실행하면, 사용자의 현재 위치가 맵 상에 표시되는데, 여기에 사용자의 프로파일 및 사용자의 목적에 따라 주변에 위치한 건물들에 대한 상세 정보와 근거리에 위치한 친구들에 대한 정보 등이 표시된다. 사용자 위치 정보의 변화를 감지하면 그에 따른 업데이트된 화면을 제공한다. 또한, 위치 정보의 변화 폭에 따라 사용자의 속도를 감지하여, 사용자가 걷고 있거나, 차를 타고 이동하는 지를 판단하여, 자동으로 맵이 줌-아웃 되거나, 주변 정보들에 대한 잦은 업데이트를 피하는 등 상황 정보에 기반한 서비스를 제공한다. 그러나, 본 시스템은 판단 및 결정을 시스템 내부에서 담당하기 때문에, 사용자가 직접 생각하고 결정을 내리기 힘들다. 또한, 위치라는 한정된 상황 정보를 추천 요소로 고려하고 있기 때문에 복합적인 요구사항을 반영하지 못하는 단점이 있다.

2005년 스위스 취리히 IBM 연구소의 A. Chen은 유비쿼터스 컴퓨팅 환경에서 협업 필터링 기법을 이용한 사용자 선호도 예측 시스템인 CACFS를 설계하였다. CACFS는 기존에 전자상거래 분야에서 널리 이용되었던 협업 필터링 기법을 최초로 유비쿼터스 컴퓨팅 환경에 적용한 사례이다. 본 시스템은 특정한 상황에 대한 행동을 미리 정의하지 않더라도, 그 상황에 있었던 다른 사람들의 데이터를 참고하여 추천을 한다는 것을 기본 전제로 하고 있다. 따라서, 사전에 정의되지 않은 상황에서 사용자가 서비스를 요청하더라도, 적절한 추천 서비스를 제공받을 수 있다. 그러나, 이러한 추천 기법은 개개인의 세밀한 정보를 반영한 추천을 제공하기엔 어려움이 따른다. 즉, 선호도 예측을 위한 데이터가 여과 과정 없이 넓은 범위를 대상으로 분석하기 때문에 효율성이 떨어지며, 자신의 데이터를 이용한 추천 보다는 다른 사람의 데이터를 이용한 추천이기에 개개인의 특성

을 반영한 개인화된 서비스를 제공하기엔 부적합하다는 단점을 가진다.

### 3. 적응형 추천 서비스 설계

본 장에서는 본 논문에서 제안하는 적응형 추천 서비스 (Adaptive Recommendation Service, ARS) 구조에 대해 설명한다. 또한, 추천에 이용하는 상호작용 상황 정보가 생성되는 과정을 보이고, 빠르고 효율적인 처리를 위한 군집 단계, 그리고 선호도를 예측하여 사용자에게 서비스 추천을 수행하는 추천 단계에 대해 각각 설명한다.

#### 3.1 ARS 구조

ARS는 유비쿼터스 환경에서 누적된 사용자와 장치 간의 상호작용 상황 정보들을 분석하여, 현재 상황에서 사용자에게 적합한 장치 및 서비스를 선용적으로 추천한다. 추천은 사용자와 장치 간의 상호작용 상황 정보를 바탕으로 이루어지며, 누적된 상호작용 상황 정보들은 군집화 및 예측 알고리즘을 통해 사용자의 선호도를 연산하여, 선호도가 높은 순으로 사용자에게 현재 상황에 적합한 장치 및 서비스를 추천한다. 여기서 장치란 TV, 컴퓨터, 전등과 같이 유비쿼터스 환경 내에 사용자가 상호작용할 수 있는 모든 사물을 의미하며, 서비스란 TV의 볼륨조절 및 채널조정 등과 같은 각 장치 별로 가지는 동작들을 말한다.

그림 1은 ARS의 전체 구조도를 나타낸다. ARS는 센싱 계층, 상황정보 생성 계층, 서비스 예측 계층, 응용 계층의 총 4계층으로 구성되며, 계층 별 순차적 처리를 통해 사용자에게 현재 상황에 적응적인 서비스를 추천한다. 센싱 계층에서는 유비쿼터스 환경 내에 구축된 위

치, 행동, 그리고 환경 등의 다양한 센서들로부터 상황 정보들을 취득한다. 또한, 사용자와 장치 간 상호작용을 습득한다. 센싱된 정보들은 상황정보 생성 계층의 상황 정보 생성 에이전트를 통해 하나의 상호작용 상황 정보 (Interaction Context, IC)로 생성되며, 이는 상호작용 프로파일 내에 누적된다. 서비스 예측 계층의 서비스 예측 에이전트에서는 상호작용 프로파일 및 다른 여러 프로파일들의 정보를 토대로 군집 및 예측의 두 단계를 거쳐 사용자에게 현재 상황에 가장 적합한 서비스를 예측한다. 군집 단계에서는 누적된 수많은 상호작용 상황 정보들을 위치 기반으로 군집화를 수행한다. 즉, 서로 가까이 있는 상황정보들끼리 군집을 형성한다. 그 후 예측 단계에서는 사용자 위치에 근접한 군집 내의 데이터를 선호도 연산 수식을 통해 사용자의 선호도를 계산하며, 우선 순위가 높은 장치 및 서비스들을 바탕으로 추천 리스트를 생성한다. 최종적으로, 응용 계층에서는 서비스 매니저에 의해 추천 리스트를 바탕으로 사용자에게 적합한 장치 및 서비스를 추천한다.

#### 3.2 상호작용 상황 정보 생성

상황정보 생성 에이전트에서는 각종 센서로부터 얻은 센싱 정보들을 취합하여 하나의 상호작용 상황 정보 (Interaction Context, IC)를 생성하고, 이를 상호작용 프로파일 내에 누적한다.

그림 2는 IC의 데이터 형식을 나타낸다. 각종 센서들로부터 받은 센싱 요소들( $s_1 \dots s_n$ )과 사용자와 장치 간 상호작용의 요소들( $i_1 \dots i_m$ )을 취합하여 하나의 IC를 생성한다. 센싱 요소에는 사용자의 위치를 나타내는 좌표값, 시간, 온도, 그리고 조도 등 각종 센서로부터 받는 주변

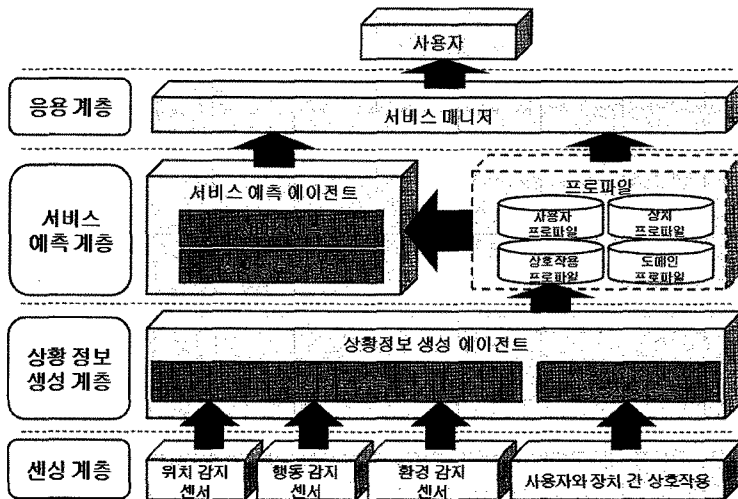


그림 1 ARS 구조

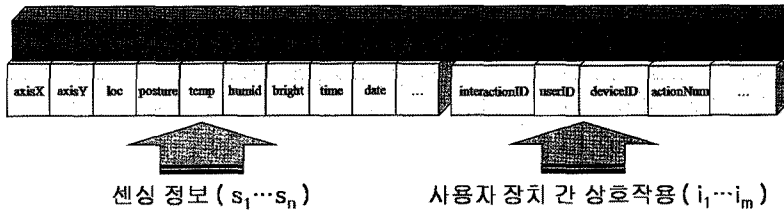


그림 2 상호작용 상황 정보(IC) 데이터 형식

상황 정보를 포함하며, 사용자와 장치 간 상호작용 요소들은 상호작용 식별자, 사용자 식별자, 장치 식별자, 그리고 장치의 행동 번호 등의 정보를 포함한다.

하나의 IC가 생성되는 시점은 사용자가 장치와 상호작용을 한 시점이며, 사용자가 특정 장치와 상호작용을 한 정보뿐만 아니라 그 시점의 상황 정보를 모두 포함함으로써, 적응형 추천 서비스를 위한 예측 데이터로 활용된다.

3.3 군집 단계

상황정보 생성 에이전트로부터 생성된 IC들은 상호작용 프로파일 내에 누적되어 서비스 예측에 활용된다. 그러나, 서비스 예측을 위해 수 많은 IC들을 모두 분석하면 효율성이나 정확성 측면에서 좋지 않다. 따라서, 군집 기법을 통해 서로 가까운 IC끼리 군집을 형성하고, 현재 사용자의 위치에 근접한 군집 내의 IC만을 분석하면 예측의 효율성 및 정확성을 높일 수 있다.

본 단계에서는 DSA K-means를 이용하여 군집화를 수행한다. DSA K-means는 군집 기법으로 널리 이용되고 있는 K-means 군집 기법의 전통적인 문제점으로 지적된 초기값 선정 과정을 개선하기 위해, 본 논문에서 새롭게 제안한 밀도기반 초기값 선정 기법(DSA: Density-based initial values Selection Algorithm)을 적용한 군집 기법으로써, 군집화의 정확성 및 효율성을 개선한다.

그림 3은 DSA의 수도 코드이다. 기존의 초기값 선정 방식이 단순히 군집의 개수 k 개만큼의 초기값을 임의로 선정하는 방식인 반면, DSA K-means에서는 데이터가 많이 분포된 곳을 중심으로 초기값 선정이 이루어지기 때문에, 군집화를 이루기 위한 비용이 적어지고 정확성이 높아지는 장점을 가진다. 즉, 초기 값이 한쪽에

치우치게 선택될 수 있는 기존의 K-means의 단점을 극복한다. 초기값 선정 이후 군집화 과정은 기존의 K-means 군집 기법과 동일하다.

3.4 추천 단계

군집화가 이루어지면, 현재 사용자의 위치에 가장 근접한 군집을 찾아 해당 IC만을 대상으로 서비스 예측을 수행한다. 서비스 예측은 사용자가 위치한 군집 내의 IC의 수가 많고 적응에 따라 두 가지 방향으로 이루어진다. 여기서 데이터의 많고 적응은 미리 정해진 한계값(T)에 의해 판별한다.

Case 1. 사용자가 속한 군집 내의 IC의 개수가 한계값 이상인 경우 ( $IC \geq T$ )

$$P_{U,j} = \frac{I_{U,device[j]} * actionRatio_{device[j]}}{\sum_{i \in C} I_{U,device[j]}} \quad (1)$$

사용자가 속한 군집 내의 IC의 개수가 충분히 많은 경우에는 선호도 계산을 위해 식 (1)을 이용한다. 식 (1)에서  $P_{U,j}$ 는 사용자 U와 장치 j 간의 선호도를 나타내며, 선호도는 사용자가 속한 군집 C 안에서 이루어진 모든 상호작용들 중 장치 j와 상호작용을 한 비율로 계산한다. 여기서  $I_{U,device[j]}$ 는 사용자 U와 장치 j 간의 IC의 개수를 나타내며, 장치 j에 대한 actionRatio 값은 장치들 마다 상이한 동작 수를 같게 유지하기 위해 적용하는 값으로써, 서로 다른 동작 수를 가진 각각의 장치들의 동작 수를 같은 비율로 유지한다. 장치 j의 actionRatio 값은 식 (2)와 같이 동작 수에 대한 비율로 구하며, 장치 프로파일 내의 각각의 장치들에서 미리 정의되어 있다고 가정한다.

$$actionRatio_{device[j]} = \frac{1}{number\ of\ action_{device[j]}} \quad (2)$$

군집 내에 속한 모든 장치 별 선호도를 위의 식 (1)을 통해 계산하여, 사용자와 장치 간의 선호도들을 우선 순위 정렬하여 추천 리스트를 생성한다. 장치에 해당하는 서비스 역시 우선 순위 정렬하여 추천 리스트에 반영한다. 추천 리스트는 최종적으로 응용 계층의 서비스 매니저에게 전달되어 가장 높은 선호도를 가진 장치들을 기준으로 사용자에게 현재 상황에서 가장 적합한 장치 및 서비스를 추천하는데 이용된다.

**알고리즘:** 밀도기반 초기값 선정 기법  
**입력 값:** 군집 개수 k, 데이터 n, 도메인 상황정보  
**출력 값:** k개의 초기값  
**방법:**  
 (1) 도메인을 k\*k 수로 균등 분할한다.  
 (2) 분할된 파티션 내 데이터의 밀도를 계산한다.  
 (3) 군집 개수 k 만큼 밀도의 크기가 큰 파티션을 내림 차순으로 정렬한다.  
 (4) 해당 k개의 파티션에서 각 데이터들의 중심 값을 구한다.  
 (5) 각각의 중심 값을 기준으로 가장 근접한 값을 초기값들로 선정한다.

그림 3 밀도기반 초기값 선정 기법

**Case 2.** 사용자가 속한 군집 내의 IC의 개수가 한계 값 미만인 경우( $IC \geq T$ )

IC의 개수가 충분히 많지 않은 경우에서 식 (1)의 수식을 이용하여 사용자의 선호도를 예측하면 정확하지 않은 결과가 나타날 확률이 높다. 이 경우에는 사용자 자신과 가장 유사한 사용자를 찾아 그의 데이터를 서비스 추천에 활용하는 협업 필터링을 활용하는 것이 효과적이다. 협업 필터링을 활용하기에 앞서, 사용자  $U$ 와 가장 유사한 사용자  $V$ 를 찾는다. 유사 사용자  $V$ 를 찾기 위해 피어슨 상관계수[19]를 이용하여 사용자 별 유사성을 측정한다. 식 (3)의 피어슨 상관계수는 두 값 사이의 선형 관계 정도를 측정함으로써 얼마나 유사한 지에 대한 정도를 판별할 수 있는 기법으로, 각 사용자 프로파일 내에 정의 되어있는 장치 별 이용성향 요소인 *personalRatio*( $pR$ ) 값을 이에 적용하여 연산한다.

$$r_{UV} = \frac{\sum_{i \in device} (U_i - \bar{U})(V_i - \bar{V})}{\sqrt{\sum_{i \in device} (U_i - \bar{U})^2 \cdot \sum_{i \in device} (V_i - \bar{V})^2}}, \quad -1 \leq r_{UV} \leq 1 \quad (3)$$

이용성향 요소  $pR$ 은 사용자의 전체 장치( $1 \dots n$ ) 사용에 대한 각 장치 별 사용 비율을 나타낸다( $pR_1 \dots pR_n$ ). 즉, 이용성향 요소의 비율 값이 높을수록 해당 장치에 대한 선호도가 높다고 판단한다. 이들 요소를 식 (3)에 대입하여 이용성향이 유사한 사용자  $V$ 를 찾는다. 식 (3)에서  $U$ 는 사용자  $U$ 의 이용성향 요소  $pR_i$ 를 나타내고,  $\bar{U}$ 는 사용자  $U$ 의 이용성향 요소들( $pR_1 \dots pR_n$ )의 평균 값을 나타낸다. 사용자  $V$ 의 경우도 동일하다. 수식의 결과값은 서로 유사할수록 1에 가깝고, 유사하지 않을수록 0에 가까운 값을 나타낸다. 값이 서로 배타적인 경우에는 음수의 경우가 나타나지만, 본 기법에서는 유사성의 정도만을 고려하므로 음수는 고려하지 않는다. 모든 사용자를 대상으로 식 (3)을 적용하여 유사성이 가장 큰 사용자를 찾는다. 가장 큰 유사성을 가진 사용자  $V$ 는 그가 가지고 있는 정보들을 유사도 만큼의 비율로 활용하여, 데이터가 불충분한 상황에서도 보다 정확하게 사용자  $U$ 의 선호도를 예측한다.

식 (4)는 사용자가 속한 군집 내의 IC가 충분하지 않은 경우 선호도 계산을 위해 사용한다. 사용자  $U$ 의 선호도를 계산하기 위해 유사 사용자  $V$ 의 IC를 활용하여 연산한다. 유사 사용자  $V$ 의 IC는 사용자  $U$ 와의 유사도 만큼의 비율로 적용된다.

$$P_{U,j} = \left\{ \frac{I_{U,device_{|j}} * actionRatio_{device_{|j}}}{\sum_{i \in C} I_{U,device_{|j}}} \right\} + r_{UV} * \left\{ \frac{I_{V,device_{|j}} * actionRatio_{device_{|j}}}{\sum_{i \in C} I_{V,device_{|j}}} \right\} \quad (4)$$

사용자  $U$ 의 IC가 서비스 추천을 하기에 충분하지 않은 상태에서는 유사한 사용자  $V$ 의 IC를 활용함으로써 사용자에게 보다 정확한 서비스 추천이 가능하다. 이에 대한 검증은 4장에서 시뮬레이션을 통해 보인다.

선호도 연산 후 Case 1과 같이 사용자와 장치 간의 선호도들을 우선 순위 정렬하여 추천 리스트를 생성하고, 서비스 매니저에 의해 가장 높은 선호도를 가진 장치들을 기준으로 사용자에게 현재 상황에서 가장 적합한 장치 및 서비스를 추천한다.

#### 4. 시뮬레이션 및 성능 평가

본 장에서는 시뮬레이션을 통해 제안하는 알고리즘의 성능 및 신뢰성을 평가한다. 먼저 본 논문에서 새롭게 제안한 DSA K-means 군집 기법의 성능을 기존의 K-means 군집 기법과 비교하여 평가하고 기존의 밀도 기반 군집화 기법들과도 비교 분석한다. 또한, 협업 필터링을 이용한 것과 그렇지 않은 것을 비교하여 본 논문에서 제안한 협업 필터링을 적용한 추천 알고리즘의 신뢰성을 평가한다. 본 시뮬레이션은 펜티엄 2.4GHz CPU를 탑재하고, 1GB의 메모리를 가진 IBM PC 상에서 수행되었으며, 시뮬레이션 프로그램은 C언어로 작성되었다.

시뮬레이션 프로그램은 일정한 규칙으로 생성된 가상의 상호작용 상황 정보(IC)를 이용한다. 실제 생활에서 누적인 상황 정보를 이용하기에는 현실적으로 제약이 있으므로, 가상의 데이터를 통해 제안 기법의 성능 및 신뢰성을 검증한다. 시뮬레이션을 위한 IC는 성능과 신뢰성을 검증할 수 있는 최소한의 요소들로 구성되며, 상호작용이 이루어진 좌표값, 생성된 순번, 그리고 상호작용한 장치 정보만을 포함한다. 이와 같은 IC를 통해 사용자가 어떠한 장치와 어느 위치에서 상호작용이 이루어졌는지에 대한 판단이 가능하며, 현재 위치에서 적합한 장치의 추천 여부를 검증한다.

$$ratio_{device_A} = actionRatio_{device_A} * personalRatio_{device_A}$$

$$IC_{device_A} = totalIC * \frac{ratio_{device_A}}{\sum_{i=0}^n ratio_{device_i}} \quad (5)$$

생성하고자 하는 총 IC의 수를 *totalIC*라고 할 때, 사용자의 각 장치별 IC를 IC장치별 동작횟수 비율인 *actionRatio*와 장치별 선호도 비율인 *personalRatio*를 반영하여 식 (5)에서 정의한 방법으로 각 장치별 IC의 수를 결정하여 생성한다.

그림 4는 시뮬레이션 화면을 나타낸다. 왼쪽은 DSA K-means이고, 오른쪽은 기존의 K-means이며, 색깔을 통해 각 군집을 구분한다. 여기서 각 점은 하나의 IC를 나타내고, 임의의 점을 기준으로 일정 반경 범위 내에서 식 (5)를 통해 결정된 장치별 IC의 수만큼 뿌려지게 된

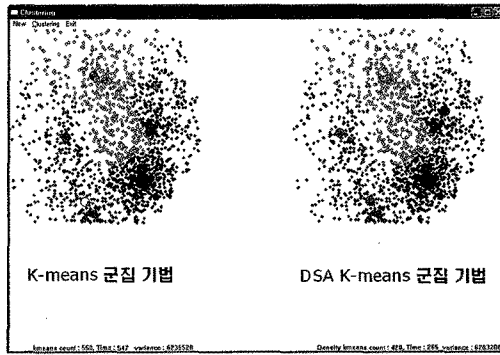


그림 4 시뮬레이션 화면

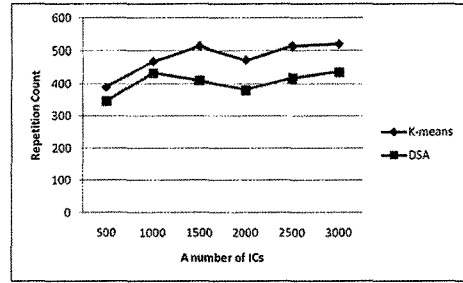
다. 각 IC들은 생성된 순번을 가짐으로써 이를 기반으로 신뢰성을 검증한다.

4.1 군집 기법 성능 평가 및 비교

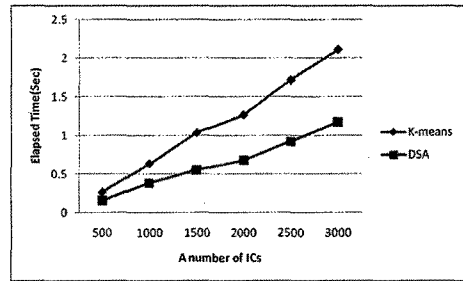
본 논문에서 제안한 DSA K-means 군집 기법을 기존의 K-means 군집 기법과 동일한  $k$  값을 갖는 환경 내에서, 각각 500, 1,000, 1,500, 2,000, 2,500, 그리고 3,000개의 데이터를 대상으로 30회씩 클러스터링을 수행한 후 그들의 평균값을 측정하여 성능을 평가하였다. 측정 항목은 군집화를 완성하는 데 걸린 반복 횟수, 소요 시간, 그리고 분산이다. 반복 횟수와 소요시간이 적을수록 보다 높은 성능을 내는 군집 기법임을 입증하며, 서로 간의 분산을 비교하여 군집화가 동일한 조건으로 수행하였는지를 확인한다.

시뮬레이션 프로그램을 통해 두 군집 기법을 비교한 결과는 그림 5와 같다. 이는 K-means 군집 기법과 DSA K-means 군집 기법의 반복 횟수, 수행시간, 그리고 분산을 비교한 차트이다. 본 차트를 통해 시뮬레이션 수행 결과 기존의 K-means 군집 기법보다 본 논문에서 제안한 DSA K-means 군집 기법이 모든 부분에서 나은 성능을 나타냄을 확인할 수 있다. 특히, 분산은 두 기법 모두 비슷한 수준을 나타내고 있는데, 이는 두 기법이 서로 비슷한 형태로 군집을 형성함을 의미한다. 즉, 두 기법 모두 비슷한 수준의 군집을 형성하면서, 반복횟수와 수행시간은 각각 15.88%와 44.62% 정도로 상당량이 감소함을 보였다. 이는 제안하는 DSA K-means 군집 기법이 기존의 K-means 군집 기법보다 우수한 성능을 나타냄을 입증한다. 이는 추천에서 선행 요소로 요구되는 군집 단계에서 성능을 끌어올림으로써, 결과적으로 전체 추천 기법의 성능을 향상시킴을 나타낸다.

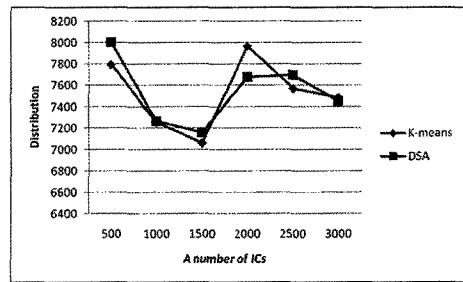
본 논문에서 제안하고 있는 DSA K-means 군집 기법을 기존의 밀도 기반 군집화 기법인 DBSCAN, PAM 및 K-means 군집화의 초기값 문제를 해결한 K-means++ 과 비교하면 표 1과 같다.



(a) 반복 횟수 비교



(b) 수행 시간 비교



(c) 분산 비교

그림 5 K-means 군집 기법과 DSA K-means 군집 기법의 성능 비교

표 1에서 군집화 기법들 중 DBSCAN과 PAM은 임의의 초기값을 선정하는 반면, 제안기법의 DSA K-means 기법은 공간을 밀도기반으로 정렬하고 밀도가 높은 곳을 초기값으로 선정하여 군집화 정확도를 향상시키고 계산 복잡도를 감소시킨다. 이와 유사하게 K-means 군집화의 초기값 선정을 공간 내의 거리 값을 이용해 계산하는 K-means++ 기법의 경우 계산복잡도가 제안 기법보다 빠르지만 도메인 내 점들의 거리를 기반으로 초기값을 선정하기 때문에 응집도가 떨어지는 점에 대해서는 효과적이지 못한 군집화를 보여준다.

4.2 추천 알고리즘 신뢰성 평가

추천 알고리즘의 신뢰성 평가 방법은 다음과 같다. 먼저 시뮬레이션을 통해 데이터들을 순차적으로 생성한 후, 본 논문에서 제안하는 수식을 적용하여 선호도를 계

표 1 DSA K-means 군집화 기법과 다른 군집화 기법들의 비교

	K-means++	DBSCAN	PAM	DSA K-means
초기값	공간 내의 점과 중심점들로부터의 거리들을 계산하여 초기값 선정	임의의 초기값	임의의 초기값	공간을 k개의 초기값으로 나누어 밀도 기반으로 초기값을 선정
계산복잡도	$O(\log k)$	$O(n \cdot \log n)$	$O(n \cdot k)$	$O(n)$
효과적인 자료형태	소형자료	대형자료	소형자료	대형 자료
군집화 방식	거리기반 초기값 선정 후 K-means 군집화 수행	거리기반 군집화	밀도 기반 군집화	밀도 기반 초기값 선정 후 K-means 군집화 수행

산한다. 계산을 통해 예측된 선호도는 순차적으로 생성되는 데이터들을 분석하여, 실제로 예측한 선호도와 얼마나 적중하였는지를 판단함으로써 본 알고리즘의 신뢰성을 평가한다.

본 알고리즘은 데이터가 충분한 상황과 충분하지 않은 상황에서 각각 다른 방식으로 예측을 수행한다. 군집 내의 데이터가 한계값 이상인 경우에는 자신의 데이터만을 사용하여 예측을 수행하는 반면, 한계값 미만에서는 자신과 유사한 사용자의 데이터를 참조하는 협업 필터링 기법을 활용하여 예측을 수행한다. 본 시물레이션에서는 두 상황에 대한 각각의 신뢰도를 평가한다.

4.2.1 한계값 이상의 데이터 선호도 예측

시물레이션 프로그램을 통해 데이터를 순차적으로 2,500개를 생성한 후, 2,400번째까지 누적된 데이터들을 대상으로 본 기법을 적용하여 선호도를 예측한다. 그 후 2,401번째부터 2,500번째까지의 데이터들을 분석하여 얼마나 적중하였는지를 판단한다.

표 2는 시물레이션 후 데이터가 충분한 상황에서의 선호도 예측 및 결과를 나타낸다. 사용자가 위치한 군집 내의 총 405개의 충분히 많은 IC들을 대상으로 식 (1)의 알고리즘을 적용하여 선호도를 계산하였다. 그 결과, 장치3, 장치0 순으로 선호도를 가진다고 예측하였다. 정확한 예측이 수행되었는지를 검증하기 위해 2,401번째부터 2,500번째까지 생성된 사용자가 위치한 군집 내의 13개의 IC들을 대상으로 선호도를 살펴본 결과, 실제로 장치3, 장치0 순으로 사용자가 수행하였다는 결과가 나타났다. 즉, 예측한 선호도와 실제로 나타난 결과가 일치함은 본 알고리즘이 데이터가 충분한 상황에서 신뢰성을 가짐을 입증한다.

표 2 한계값 이상의 데이터 선호도 예측 및 결과

	1 <sup>st</sup> ~ 2400 <sup>th</sup> (예측)		2401 <sup>st</sup> ~ 2500 <sup>th</sup> (결과)	
	상호작용	선호도	상호작용	선호도
장치0	122	30.12%	5	38.46%
장치1	6	1.48%	0	0%
장치2	0	0%	0	0%
장치3	273	67.41%	8	61.54%
장치4	4	0.99%	0	0%

4.2.2 한계값 미만의 데이터 선호도 예측

사용자가 위치한 군집 내의 데이터가 충분하지 않은 상황에서는 비교적 정확한 예측을 수행하기 어렵다. 표 3은 시물레이션 후 데이터가 부족할 때 본인의 데이터만을 가지고 예측을 수행한 결과이다. 이 경우 부족한 데이터를 통해 예측을 수행하였기 때문에, 그 정확성이 현저히 떨어짐을 알 수 있다. 그러나, 본 알고리즘에서 제안하는 협업 필터링을 이용하면, 보다 풍부한 데이터를 가진 자신과 유사한 사용자의 데이터를 선호도 예측에 활용함으로써 보다 정확한 예측을 수행할 수 있다.

표 4는 사용자의 데이터가 부족할 때 협업 필터링을 이용하여 예측된 선호도와 그 결과를 나타낸다. 유사 사용자인 사용자2를 피어슨의 상관계수를 통해 구한 후, 식 (4)에 사용자2의 IC들을 적용하여 각 장치별 선호도를 계산한다. 그 결과, 표 4에서와 같이 선호도 예측 결

표 3 한계값 미만의 데이터 선호도 예측 결과

	1 <sup>st</sup> ~ 50 <sup>th</sup> (예측)		51 <sup>st</sup> ~ 2500 <sup>th</sup> (결과)	
	상호작용	선호도	상호작용	선호도
장치0	4	36.36%	77	11.39%
장치1	0	0%	0	0%
장치2	0	0%	1	0.15%
장치3	3	27.27%	65	9.61%
장치4	4	36.36%	533	78.85%

표 4 협업 필터링을 통한 선호도 예측 결과

	1 <sup>st</sup> ~ 50 <sup>th</sup> (사용자1)		1 <sup>st</sup> ~ 2500 <sup>th</sup> (사용자2)	
	상호작용	선호도	상호작용	선호도
장치0	4	36.36%	21	3.32%
장치1	0	0%	1	0.16%
장치2	0	0%	3	0.47%
장치3	3	27.27%	101	15.95%
장치4	4	36.36%	507	80.09%
	협업 필터링 이용		51 <sup>st</sup> ~ 2500 <sup>th</sup> (결과)	
	연산	선호도	상호작용	선호도
장치0	0.3927	20.91%	77	11.39%
장치1	0.0014	0.07%	0	0%
장치2	0.0041	0.21%	1	0.15%
장치3	0.4127	21.98%	65	9.61%
장치4	1.0661	56.8%	533	78.85%

과 장치4가 가장 높은 선호도를 가지고 있음을 알 수 있다. 실제로 사용자1의 이후 나타난 IC들을 분석했을 때 역시 장치4가 가장 높은 선호도를 가지고 있는 것으로 나타났다. 이러한 결과는 사용자의 데이터가 충분하지 않은 상황에서 협업 필터링을 이용하는 본 알고리즘이 보다 신뢰성 있는 추천을 수행함을 입증한다.

## 5. 결론

본 논문에서는 유비쿼터스 환경 내에서 사용자와 장치 간의 상호작용 상황 정보를 바탕으로 현재 상황에 가장 적합한 서비스를 추천하는 적응형 추천 서비스 기법을 제안하였다. 추천을 위해 사용자와 장치들 간의 상호작용으로 생성되는 상호작용 상황 정보를 이용함으로써 보다 개인화된 서비스 제공하였다. 누적된 대량의 정보들을 모두 분석하는 대신에 군집 기법을 통해 현재 사용자의 위치에 근접한 군집 내의 데이터만을 이용함으로써 예측의 효율성 및 정확성의 향상을 보였다. 또한, 유사 사용자의 데이터를 활용하는 협업 필터링을 이용함으로써 데이터가 충분하지 않은 상황에서도 보다 정확한 추천을 가능하게 하였다.

본 논문에서는 제안된 알고리즘의 성능 및 신뢰성 평가를 위해 시뮬레이션을 수행하였다. 그 결과, 본 논문에서 제안한 DSA K-means군집 기법은 기존의 K-means 군집 기법보다 반복 횟수와 수행 시간을 각각 15.88%와 44.62% 감소 시킴으로써 전체적인 성능 향상을 보였다. 또한, 가상의 데이터를 생성하여 본 논문에서 제안하는 추천 알고리즘을 검증한 결과, 신뢰성 있는 추천을 수행함을 입증하였고, 특히, 데이터가 충분하지 않은 상황에서도 협업 필터링 기법을 적용한 본 기법이 정확한 추천을 보장함을 확인하였다.

## 참고 문헌

- [1] W. Lee, "Towards Agent-based Decision making in the electronic market place: Interactive recommendation and automated negotiation," *Elsevier Expert Systems with Applications*, vol.27, pp.665-679, 2004.
- [2] O. Kwon, K. Yoo, and E. Suh, "UbiDSS: A Proactive Intelligent Decision Support System as An Expert System Deploying Ubiquitous Computing Technologies," *Elsevier Expert Systems with Applications*, vol.28, pp.149-161, 2005.
- [3] DW. McDonald, "Ubiquitous Recommendation Systems," *IEEE invisible computing*, vol.36, pp.111-112, 2003.
- [4] J. J. Barton and T. Kindberg, "The Cooltown User Experience," *Proc. of CHI2001*, pp.149-156, 2001.
- [5] D. Garlan, D. P. Siewiorek, A. Smailagic, and P. Steenkiste, "Project Aura: Toward Distraction-Free Pervasive Computing," *IEEE Pervasive Computing*, pp.22-31, 2002.
- [6] Y. Takeuchi and M. Sugimoto, "CityVoyager: An Outdoor Recommendation on User Location History," *Proc. of Ubiquitous Intelligence and Computing (UIC2006)*, pp.625-636, 2006.
- [7] J. K. Kim, H. K. Kim, and Y. H. Cho, "A User-oriented Contents Recommendation System in Peer-to-peer Architecture," *Elsevier Expert Systems with Applications*, vol.34, pp.300-312, 2008.
- [8] F. Ricci, and Q.N. Nguyen, "Acquiring and Revising Preferences in A Critique-based Mobile Recommender System," *IEEE Intelligent systems*, vol.22, pp.22-29, 2007.
- [9] M. C. Chen, L. S. Chen, F. H. Hsu, Y. Hsu, and H. Y. Chou, "HPRS: A Profitability Based Recommender System," *Proc. of IEEE International Conference on Industrial Engineering and Engineering Management*, pp.219-223, 2007.
- [10] W. Woerndl, and G. Groh, "Utilizing Physical and Social Context to Improve Recommender Systems," *Proc. of IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology Workshops*, pp.123-128, 2007.
- [11] C. Ono, M. Kurokawa, Y. Motomura, and H. Asoh, "A Context-aware Movie Preference Model Using A Bayesian Network for Recommendation and Promotion," *LECTURE NOTES IN COMPUTER SCIENCE*, vol.4511, pp.247-257, 2007.
- [12] K. Oku, S. Nakajima, J. Miyazaki, S. Uemura, "Context-Aware SVM for Context-Dependent Information Recommendation," *Proc. of IEEE Proceedings of the 7th International Conference on Mobile Data Management*, pp.109-112, 2006.
- [13] M. V. Setten, S. Pokraev, and J. Koolwaaij, "Context-Aware Recommendations in the Mobile Tourist Application COMPASS," *Proc. of AH2004, Lecture Notes In Computer Science*, vol.3137, Springer-Verlag, pp.235-244, 2004.
- [14] A. Chen, "Context-Aware Collaborative Filtering System: Predicting the User's Preference in the Ubiquitous Computing Environment," *Proc. of LoCA2005, Lecture Notes In Computer Science*, vol.3479, Springer-Verlag, pp.244-253, 2005.
- [15] Q. He, "A Review of Clustering Algorithms as Applied in IR," *Proc. of UIUCLIS1999*, pp.1-33, 1999.
- [16] Schafer, J.B. and Frankowski, D. and Herlocker, J. and Sen, S., "Collaborative filtering recommender systems," *Lecture Notes In Computer Science*, Springer-Verlag, vol.4321, pp.291-324, 2007.
- [17] M. O'Connor, J. Herlocker, "Clustering items for collaborative filtering," the *Proc. of SIGIR-2001 Workshop on Recommender Systems*, 2001.
- [18] M. Balabnovic and Y. Shoham, "Content-based, Collaborative Recommendation," *Communications of*



*the ACM*, vol.40, pp.66-72, 1997.

- [19] D. Billsus and M. J. Pazzani, "Learning Collaborative Information Filters," *Proc. of ICML*, pp.46-53, 1998.



**최 정 환**

2007년 성균관대학교 컴퓨터공학과(학사)  
 2009년 성균관대학교 휴대론학과(공학 석사). 2009년~현재 삼성전자 DMC 연구소  
 관심분야는 추천자 시스템, 상황 인지,  
 미들웨어, 접근 제어 등



**류 상 현**

2009년 성균관대학교 컴퓨터공학과(학사)  
 2009년~현재 성균관대학교 전자전기컴퓨터공학과(석사과정). 관심분야는 상황  
 인지, 미들웨어, 분산 컴퓨팅 등



**장 현 수**

2002년 성균관대학교 전자전기컴퓨터공학과(학사). 2005년 성균관대학교 대학원 전자전기컴퓨터공학과(공학석사). 2006년~현재 성균관대학교 전자전기컴퓨터공학과(박사과정). 관심분야는 유비쿼터스 컴퓨팅, 이동 에이전트, HCI, 미들웨어 등



**엄 영 익**

1983년 서울대학교 계산통계학과(학사)  
 1985년 서울대학교 전산학과(이학석사)  
 1991년 서울대학교 전산학과(이학박사)  
 2000년~2001년 Dept. of Info. and Comm. Science at UCI 방문교수. 1993년~현재 성균관대학교 정보통신공학부 교수. 2007년~현재 성균관대학교 정보통신처 처장. 2008년~현재 한국정보과학회 학회지 편집부위원장. 관심분야는 분산 컴퓨팅, 미들웨어, 운영체제, 시스템 보안, 내장형 시스템 등