

# 내용 기반 추천기법의 TV 환경 적용에 관한 연구

유상원<sup>0</sup> 이홍래 이형동 김형주  
서울대학교 전기.컴퓨터공학부  
{swyoo<sup>0</sup>, hrlee, hdlee, hjk}@oopsla.snu.ac.kr

## Content based recommendation in TV environment

Sangwon Yoo<sup>0</sup> Hongrae Lee Hyungdong Lee Hyung-joo Kim  
School of Computer Science and Engineering, Seoul National University

### 요 약

다양한 분야를 대상으로 추천기법에 관한 연구 및 적용이 이루어지고 있다. 전자 상거래 분야에서 소비자가 선호할만한 상품을 추천하거나 영화 관련 사이트에서 불만찬 영화를 추천해주는 것들이 대표적인 예이다. TV 프로그램 또한 채널의 수가 수 백개 이상으로 늘어남에 따라 추천의 필요성이 제기되고 있다. 본 연구에서는 TV 프로그램들을 대상으로 하는 추천 시스템을 구현하였다. 추천 기법은 내용 기반 방식으로 이루어져 있으며 실험을 통해 내용기반 방식이 TV환경에서 가지는 효용성을 알아보고 적용 가능성을 타진해 보았다.

## 1. 서 론

TV환경에 새로운 기술들이 접목되면서 TV시청 패턴에도 변화가 불가피하게 되었다. 기존의 공중과 중심으로 이루어진 TV에서는 시청자가 원하는 프로그램을 시청하기 위해 특별히 고려해야 할 사항이 없었다. 케이블 TV를 포함해도 채널과 프로그램의 수가 그리 많지 않기 때문에 본인이 선호하는 채널의 번호를 암기하거나 특정 시간대에 원하는 프로그램을 찾아서 시청하는 일에 별다른 어려움이 없었다.

하지만 디지털 방송시대가 열리고 각 가정마다 디지털 방송이 일반화되면 채널의 수가 수백 개로 늘어난 상황에서 이전과 같은 채널과 프로그램의 선택은 불가능해진다. 예를 들어 시청 가능한 TV채널이 300개 정도라고 가정해 보자. 이런 상황에서 지금처럼 리모콘을 돌려가며 각 채널에서 원하는 프로그램을 찾으려 한다면 채널당 1초가 걸린다고 해도 300초, 즉 5분의 시간이 걸리게 되는 것이다.

이미 TV환경이 아닌 다른 환경에서의 추천내지 필터링 관련 시스템에 관한 연구는 많이 이루어져 왔다[1][2]. 인터넷과 더불어 온라인상의 정보가 기하급수적으로 증가하고 사용자가 원하는 정보를 찾기가 점점 어려워지면서 사용자가 원하는 정보를 직접 찾지 않아도 받아보는 필터링 방식이나 추천 방식이 필요하게 된 것이다.

현재 추천기법으로 쓰이는 양대 축은 내용 기반 추천(Content-based recommendation)과 협업 기반 추천(Collaborative recommendation)이다. 협업 기반 추천이 실제로 많이 쓰이고 있으며 두 가지 방식을 혼합하여 사용하기도 한다[3][4].

내용기반 방식은 정보 검색(Information Retrieval)분야에서 기본적인 아이디어를 가져온 것이라고 할 수 있다. 검색 시스템에서 사용자는 시스템에 질의를 던지고 시스템은 사용자에게 질의의 결과로 문서를 순위화하여 보여준다. 이때 순위를 매기는 기준은 사용자의 질의와 질의의 대상이 되는 문서가 얼마나 유사한가이다. 이때 유사도를 측정하는 방법은 여러 가지가 있지만 기본적으로 문서의 내용 즉 문서 안에 있는 단어들과 질의가 가진 단어들을 비교하는 것으로 이루어진다.

내용 기반 추천기법도 마찬가지로 비슷한 방식을 취한다. 차이점은 사용자가 원하는 정보를 찾기 위해 질의를 던지기 전에 시스템이 추천을 해야 하므로 사용자의 질의는 사용자 정보(user profile)로 대체된다. 사용자 정보는 개념적으로 장기간 동안 변하지 않는 질의라고 생각할 수 있다.

협업 기반 추천 기법은 취향이 비슷한 사람을 하나로 묶어 그 그룹 내에서 다른 사람들이 본 영화라면 나에게도 흥미가 있을 것이라는 가정에서 출발한 방식이다. 사람의 평가 또는 추천에 근거를 두고 있으므로 추천의 대상이 되는 아이템에 관한 정보가 부족하거나 기술하기 곤란한 경우에도 적용 가능하다. 마찬가지로 이유로 자신이 선호한다고 입력한 것과 내용 기반 방식에 의하면 거리가 있는 것들도 그룹 내의 다른 사람이 추천할 경우 나에게 추천될 수 있다.

TV프로그램을 위한 추천시스템의 경우 두 가지 기법을 택함에 있어 두 가지 기법이 가진 기본적인 장단점 이외에도 환경적인 요인을 고려해야만 한다. 협업 기반 방식이 우수함에도 불구하고 적용하기 곤란한 이유는 다음과 같다. 첫째, 협업 기반 방식의 경우 다른 사람이 추천한 프로그램을 내가 시청해야 하는데 어떤 프로그램을 시청한 후에 추천을 해 준다면 이미 그 프로그램은 더 이상 볼 수 없게 된다. 시간적인 문제가 생기게 되는 것이다. VOD 환경의 경우에도 영화나 드라마와 같은 특정 장르가 아니면 일반적으로 서비스가 되지 않고 일정시간이 지나면 뉴스와 같은 장르는 의미가 없어지기 때문에 적절하지 않다. 둘째, 협업 기반의 경우 많은 시청자들의 시청기록과 프로그램에 대한 평가들을 관리해 줄 중앙 서버가 필요하다. 반면 내용 기반 방식의 경우 현재 상용화 되어 있는 것처럼 TV프로그램 정보만 정기적으로 제공 받으면 쉽게 셋탑박스나 간단한 저장장치 만으로도 구현 가능하다.

본 논문에서 구현한 추천 시스템은 내용 기반 방식을 택하였다. 물론 추후에 여러 가지 환경적인 요인들이 바뀐다면 협업 추천 방식도 도입 가능할 것이다.

## 2. 추천 시스템의 구현

### 2.1 사용자 정보 모델링

추천을 하는데 있어서 비교의 대상이 되는 것은 TV프로그램이 가진 프로그램 정보와 사용자의 선호도에 관한 정보이다. 사용자의 선호도는 다음과 같이 모델링하였다.

- \* 장르, 세부장르  
예를 들어 장르는 뉴스, 세부장르는 스포츠 뉴스로 표현 가능하다. 장르는 형식적인 분류이며 세부장르는 내용적인 분류이다.
- \* 채널  
같은 영화채널이라도 방영되는 영화의 패턴이나 유료, 무료 등에 따라 선호도가 달라진다.
- \* 출연자  
출연자는 연예인, 운동선수, 스포츠 팀등 다양할 수 있다.
- \* 주제어  
방송 내용이 가진 주제를 나타내는 키워드가 있을 수 있다. 사랑, 우정과 같은 추상적인 선호도 표현이 가능하다.  
각각의 요소들은 가중치를 가질 수 있으며 추천시에 사용자의 선호도 패턴에 따라 반영되는 항목간의 비율이 달라질 수 있다. 이를 그림으로 나타내면 다음과 같다.

사용자 정보	가중치	반영 비율
장르	뉴스 2 엔터테인먼트 3	20%
세부장르	경제뉴스 4 토크쇼 2	40%
채널	NBC 3 ABC 5	10%
출연자	김철수 3 농구팀 1	15%
주제어	사랑 2 우정 3	15%

그림 1. 사용자 정보 모델링

## 2.2 시스템의 구조

우리가 구현한 시스템의 전체적인 구조는 그림 2와 같다. 양방향 TV를 시뮬레이션 하기 위해 웹 인터페이스를 사용하였으며 JAVA와 JSP를 이용하여 구현하였다. 사용자 정보의 저장은 데이터베이스의 테이블을 통해 이루어진다.

전체적인 워크 플로우를 살펴보면 사용자 정보는 사용자 정보 관리자에 의해 관리된다. 사용자 정보 관리자는 시스템 사용자로부터 해당 사용자를 식별할 수 있는 아이디와 사용자의 초기 선호도를 얻는다. 사용자의 선호도는 사용자 정보 DB에서 관리되며 각 요소별로 별도의 테이블을 가진다. 사용자의 초기 선호도를 충분히 얻기 위해서는 사용자에게 많은 양의 정보를 요구해야만 한다. 이럴 경우 사용자가 추천기능 자체를 외면하기 쉽기 때문에 우리가 구현한 시스템은 사용자의 초기 입력을 최소로 하고 이후 사용자의 시스템 이용 기록을 토대로 사용자의 선호도를 수정해 나간다. 이용 기록에 해당하는 것은 시스템이 TV프로그램을 추천했을 경우 이에 대해 긍정과 부정으로 평가를 내리는 것과 시청을 원하는 프로그램을 검색하는 것이다. 또한 가장 중요한 것은 사용자가 시청한 TV프로그램의 요소들을 반영하는 것이다.

$$Q_{i+1} = \alpha Q_i + \beta \sum_{d \in R} v_d - \gamma \sum_{d \in NR} v_d \quad [5]$$

식 [5]에 의해 사용자 정보의 가중치는 증가하거나 감소하게 된다.

사용자 정보에 포함된 요소들 간의 반영 비율은 다음과 같은 가정에 의해 조정된다. 사용자에게 따라 여러 가지 선호 요소들 중 특정 요소에 더 많이 좌우될 수도 있다는 것이다. 예를 들어 사용자가 추천한 프로그램에 긍정을 표시하면 추천의 근거가 된 요소가 채널인 경우 해당 요소의 반영 비율을 올려주고 부정을 표시하면 줄여주게 된다. 일정기간 동안 사용자의

평가를 누적하였다가 이를 사용자 정보에 반영한다.

추천 엔진은 사용자 정보와 프로그램 정보를 검색하여 유사성을 비교한 후 프로그램에 순위를 부여하는 역할을 한다. 그림 1처럼 사용자는 2가지 종류의 선호도를 가지게 되는데 하나는 장르나 채널과 같은 각 요소의 값들에 대한 선호도이며 다른 하나는 자신이 여러 요소 중 어느 쪽에 더 영향을 많이 받는가에 대한 선호도이다.

사용자의 요소 값들에 대한 선호도는 양의 값과 음의 값의 일정한 범위 내에서 가지며 초기 정보 입력시 사용자로부터 얻은 값들은 1로 초기화된다. 사용자의 요소별 영향 비율에 대한 반영은 균등하게 반영되는 것으로 초기화 된다.

프로그램 정보 DB에서 사용자 정보에 해당하는 요소와 일치 여부를 검사하여 점수를 부여한다.

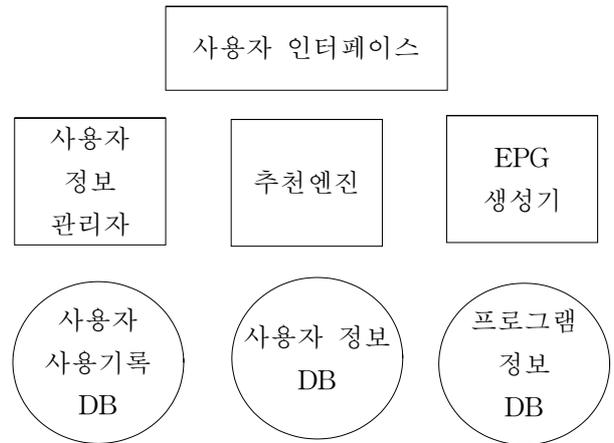


그림 2. 추천시스템의 구조

$$Score = \sum_{i=1}^k ratio_i * weight_i$$

각 프로그램의 점수는 위의 식에 의하여 계산된다. ratio는 해당 요소의 반영비율을 나타내며 weight는 해당 요소의 가중치를 나타낸다.

프로그램 정보DB는 TVAnytime표준[6]을 따르는 XML 데이터를 파싱하여 DB테이블의 형태로 관리한다.

EPG는 Electronic Program Guide의 약자로써 오프라인 상의 편성표를 화면으로 옮겨놓은 것을 말한다. 시스템은 프로그램 정보를 이용하여 사용자에게 자신이 원하는 프로그램에 관한 정보를 얻을 수 있는 인터페이스를 제공한다.

## 3. 실험 및 결과

TV프로그램 추천을 위한 실험은 환경의 제약으로 인해 8일본 공중파 5개 채널(KBS1, KBS2, MBC, SBS, EBS) 데이터를 이용하여 이루어졌다. 본 논문에서 가정한 환경을 직접 실험하기 위해서는 수백 개 채널의 수개월 분에 해당하는 데이터를 필요로 하지만 현재 이를 제공할 수 있는 기관이 없어 8일본의 데이터만으로 가능성을 보이도록 하였다.

실험에 사용한 프로그램의 총수는 1036개이며 8명의 사람이 실험에 참여하였다. 실험 참여 방법은 본 시스템에 접속하여 프로그램 정보를 보고 날짜별로 본인이 시청할 프로그램을 선택하고 시스템이 추천해 준 프로그램에 대해 긍정과 부정 두 가지 평가를 내리는 것으로 이루어졌다. 사용자의 평가 스케일은 일반적으로 3가지 스케일이나(긍정, 부정, 보통) 5가지 스케일

일(아주 좋음, 좋음, 보통, 나쁨, 매우 나쁨)도 사용하며 추천 결과를 더 좋게 만들 수 있다. 그러나 실제 TV에서 사용자에게 자세한 평가를 요구하기는 어렵다고 보고 긍정과 부정 두 가지 스케일로 사용자의 평가를 입력 받았다.

채널수가 적고 공중파의 특성 때문에 채널에 대한 선호도는 제외하였고 프로그램 수가 적어 검색기능이 이용되지 않아 이 기록도 배제하였다.

시스템의 성능 척도는 coverage를 사용한다. coverage는 일자별로 사용자가 시청한 프로그램들이 얼마나 많이 추천 리스트에 포함되어 있는가에 관한 척도이다[7]. 추천 시스템을 평가하는 척도로 영화와 관련된 분야에서는 MAE(Mean Absolute Error)가 많이 사용된다[8]. 하지만 TV와 같이 사용자 입장에서 프로그램을 본다 안본다의 두 가지로만 나누는 경우는 IR의 precision[7]과 유사한 coverage가 더 적절하다고 볼 수 있다.

$$coverage\ ratio = \frac{n(R \cap W)}{n(W)} \quad [7]$$

R: Recommended list, W: Watching list

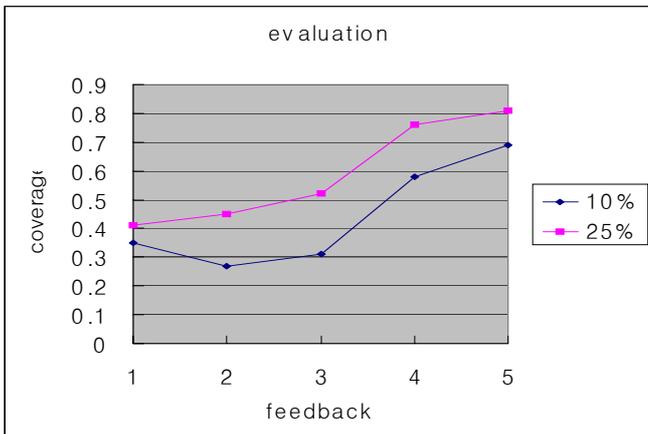


그림 3. 실험 결과

8명의 사용자는 7일 분의 프로그램에 대해 시스템의 추천에 대한 평가를 내렸으며 8일 분의 방송에 대해 시청한 프로그램을 기록하였다.

시스템의 성능에 대한 평가는 마지막 8일째 방송에 대한 시스템의 추천과 사용자의 시청기록을 비교하는 것으로 이루어졌다. 그림3은 시스템의 성능을 그래프로 나타낸 것이다. 사용자의 프로그램에 대한 평가가 본다와 안본다 두 가지이기 때문에 프로그램에 순위를 매기는 시스템의 입장에서 보면 순위 50%안에 시청한 프로그램이 얼마나 들어 있는가를 계산하여야 하겠지만 이는 현실성이 떨어지기 때문에 순위10%와 25%인 경우를 비교하였다. 그림3을 보면 사용자의 초기 정보에서 사용자가 시스템에 추천에 대한 평가와 시청기록을 통해 추가 정보를 제공함으로써 추천의 질이 향상됨을 확인할 수 있다. feedback 2는 사용자의 추천에 대한 평가가 4일분이 반영된 경우이며 feedback 3은 7일분이 반영된 경우이다. feedback 4는 시청기록 4일분이 추가로 반영된 것이며 feedback 5는 시청기록 7일 분이 반영되었음을 나타낸다.

coverage가 떨어졌다가 다시 올라가는 현상은 사용자가 자신이 입력한 초기 정보가 어떤 형태로 반영이 될지 정확히 알 수가 없기 때문에 이에 적응해 나가는 기간이라고 분석할 수 있다.

실험 결과 내용 기반 방식만으로도 TV환경에서 일정 품질 이상의 추천결과를 얻어 낼 수 있었다.

#### 4. 결론

본 논문은 TV환경 하에서의 추천 시스템을 구현하고 실험을 통해 이의 효용성을 보였다. 충분한 데이터와 사용자를 확보할 수 있다면 좀 더 여러 가지 기능적인 요소들과 추천 기법을 추가할 수 있을 것이다. 추천 시스템 중 내용 기반 방식을 사용하는 시스템은 항상 추천의 대상이 되는 도메인에 여러 의 존적인 특징들을 가지고 있다. 사용자의 성향 분석과 충분한 프로그램 정보와 같은 요인들은 시스템을 좀 더 정교하게 만드는 데 많은 도움을 줄 수 있을 것이다. 또한 추천 시스템의 성능을 객관적으로 평가하기 위해서는 정보검색 분야와 같은 테스트 컬렉션이 필요하다. 현재 사용되고 있는 영화데이터 만으로 다른 도메인에 대한 추천기법을 평가하는 것이 곤란하므로 별도의 테스트 컬렉션이 마련되어야 할 것이다.

TV에는 TV만이 가진 여러 가지 특징이 있다. 프로그램의 스케줄이 일정 주기로 반복되므로 이를 추천에 이용할 수 있을 것이고 실제 시청자의 시청기록을 얻을 수 있으면 훨씬 더 나은 결과를 얻을 수 있을 것이다. 또한 협업 기반 방식이 가진 장점을 살리기 위해 연령별, 성별, 직업별 시청률 데이터 같은 것 등을 활용하여 추천 리스트에 포함시킨다면 내용 기반 방식이 가진 단점을 상당 부분 메워나갈 수 있을 것이다.

#### 참고문헌

- [1] Paul Resnick and Hal R. Varian, Recommender Systems, CACM 40(3) p56-58, 1997
- [2] Peter W. Foltz and Susan T. Dumais, Personalized Information Delivery: An Analysis of Information Filtering Methods, CACM 35(12) p51-60, 1992
- [3] Resnick, P., Iacovou, N., Sushak, M., Bergstrom, P., and Riedl, J. GroupLens: An open architecture for collaborative filtering of netnews. Proceedings of the 1994 Computer Supported Collaborative Work Conference. 1994
- [4] Marko Balabnovic and Yova Shoham, Fab: Content-Based, Collaborative Recommendation, CACM 40(3) p66-72, 1997
- [5] J.J.Rocchio Relevance feedback in information retrieval. In The Smart System-Experiments in Automatic Document Processing, p 313-323, 1971
- [6] <http://www.tv-anytime.org/>
- [7]R. Korfhage. Information Storage and Retrieval. John Wiley, New York, 1997.
- [8]Herlocker, J. Understanding and Improving Automated Collaborative Filtering Systems. Ph.D. Thesis, Computer Science Dept., University of Minnesota, 2000