

자동화된 추천을 위한 협동적 필터링 에이전트 시스템의 개발

황병연, 김의찬
가톨릭대학교 컴퓨터공학과
e-mail : byhwang@www.cuk.ac.kr

Development of Collaborative Filtering Agent System for Automatic Recommendation

Byungyeon Hwang, Euichan Kim
Dept. of Computer Engineering, The Catholic University of Korea

요 약

최근 전자상거래에서 에이전트 기술들이 많이 나타나고 있는데, 주목해야 할 것은 패키지 형태로 내장될 수 있는 에이전트이다. 전자상거래 솔루션에 탑재되어 자동화시킨 에이전트로서 NetPerception의 GroupLens 엔진과 MacroMedia의 LikeMinds가 있는데 이들은 협동적 필터링을 구현한 것들이다. 현재 이러한 협동적 필터링 에이전트 시스템이 탑재된 전자상거래 솔루션들이 등장하고 있다. 하지만 add-on 성격이 부족하고, 실제 협동적 필터링 알고리즘에 의해 고객의 취향이나 기호에 맞는 아이템을 추천하는 진정한 의미의 에이전트 시스템은 찾아보기 힘들다. 그래서, 이러한 점을 보완한 MindReader 시스템을 개발하였다. 제안된 알고리즘은 기존의 GroupLens 알고리즘에 클러스터링을 접목시킨 알고리즘을 사용하였다.

1. 서론

오늘날 전자상거래의 판매자들은 증가하는 내적 경쟁에 직면하고 있다. 이들은 방문객을 빠르게 구매자로 변화시키고, 각각의 고객에게 구매할 기회를 최대화하여, 오랜 기간 자신들과의 관계가 지속되게 할 필요성을 강하게 느끼고 있는 것이다[1]. 불과 몇 년 전만 하더라도 전자상거래의 미래에 대한 이야기와 함께 전통적 상거래에서 전자상거래로 전환하는 것의 필요성과 효과에 대한 논의가 전자상거래의 화두였다. 그러나 지금은 이미 많은 성공적인 전자상거래 솔루션들이 등장했고, 또한 많은 전자상거래 사이트가 구축되어 있기에 현재의 주된 관심은 전자상거래 내에 서 벌어지는 경쟁에 그 초점이 맞춰져 있다.

전자상거래로의 전환이 중요했던 초기에는 전자상거래를 구축할 수 있는 커머스 서버의 진출에 따른 불안감을 해소할 수 있는 보안 솔루션 등이 주로 부각된 반면, 현재는 전자상거래 내의 경쟁력이 강조되고 있으며 고객에게 양질의 서비스를 제공하고, 이를 이익 창출로 연결시키는 데 도움이 되는 다양한 솔루션들이 부각되고 있다.

선들이 부각되고 있다.

아마존(amazon.com) 사이트만 하더라도 특정 책을 조회하면 이 책을 구매했던 사용자들이 많이 구매한 책의 리스트를 제시해주는 'Who bought'[2] 서비스와 고객이 특정 책들에 호감도를 평가한 평가지력을 바탕으로 그 고객이 좋아할 만한 책을 추천해주는 자동화된 추천서비스를 하고 있다. 이처럼 특화된 에이전트를 적용하여 고객의 취향과 기호에 맞는 아이템 추천기능을 수행하고 있는 여러 전자상거래 사이트가 큰 성공을 이루고 있다.

본 연구에서는 자동화된 추천을 위한 협동적 필터링 에이전트 시스템인 MindReader 시스템을 개발하였다. 본 연구에서 개발된 MindReader는 CORBA를 사용하였고, add-on 제품으로서의 기능을 발휘할 수 있도록 하였다. 또한 컴포넌트 기반의 구조로서 정확하고 빠른 추천을 할 수 있다.

2 장에서는 전자상거래 에이전트와 협동적 필터링 알고리즘이 어떤 것인지 간단히 살펴보고 3 장에서는 본 연구에서 제안된 MindReader 시스템에 대해 기술한다. 끝으로 4 장에서 결론을 맺도록 하겠다.

2. 관련 연구

2.1 전자상거래 에이전트

인터넷 전자상거래에서 판매자와 구매자가 꾸준히 증가하면서, 보다 빠르고 쉽게 자신의 목적을 달성하도록 하기 위해서 기존 에이전트 기술들이 전자상거래 분야에 적용되고 있다. 두드러지게 나타나는 에이전트들은 검색 에이전트, 협상 에이전트, 추천 에이전트들이 있다. 검색 에이전트는 여러 사이트에 있는 상품을 비교, 검색하거나(Firefly), 자연어 인식을 통한 검색(empas.com)을 지원한다. 협상 에이전트는 구매자나 판매자의 요구조건을 받아들여서 각각의 사용자를 대리하여 물건을 사고 파는 에이전트이다. 대표적인 경우가 경매 대행 에이전트로서 AuctionBot 이 있다. 이 시스템에서 구매하거나 판매하려는 사람들은 가격 정책, 구매 결정 시한 등을 설정하여 자신의 에이전트가 활동할 수 있는 기본 규칙을 정의할 수 있다. 그렇게 하면 AuctionBot 은 사용자의 개입 없이도 사용자가 설정한 목적과 행동의 가이드라인을 지키면서 고객을 대신하여 다른 에이전트와 협상을 진행하고, 결과를 사용자 자신에게 알려주게 된다. 추천 에이전트는 다양하게 분류될 수 있지만, 공통의 목적은 방대한 정보를 사용자에게 빠르게 매칭 시켜주는 데에 있다.

여기서 주목해야 할 것은 전자상거래 솔루션에 패키지 형태로 내장될 수 있는 에이전트이다. 전자상거래 솔루션에 탑재되는 에이전트가 갖추어야 할 조건은 가급적 자동화될 것, 그리고 기능이 특화된 것이 요구된다. 대표적인 경우가 NetPerception 의 GroupLens[3, 4] 엔진과 MacroMedia 의 LikeMinds[5]이다. 이들은 모두 협동적 필터링을 구현한 것이다. 자동화되지 않고, 특화되지 않았다고 하면, 개별 사이트가 가지는 성격에 따라 많은 커스터마이징이 필요하게 되고, 그럴 경우 하나의 특화된 에이전트 솔루션이 되기 힘들다. 이처럼 전자상거래 시장의 동향과 에이전트 기술의 동향은 특화된 솔루션이 경쟁력을 가질 수 있다는 것을 의미하고 있으며, 특히 이미 구축되어 있는 사이트에 통합되어 추천 기능을 수행하는 자동화된 에이전트 시스템의 개발 및 상품화는 특화된 솔루션이 요구되는 상황에서 큰 경쟁력을 가지게 되는 것이다. 이러한 상황에서 협동적 필터링 에이전트 시스템이 탑재된 전자상거래 솔루션이 계속 등장하고 있으며, 이를 이용한 사이트 수도 늘어나고 있다. 하지만 진정한 의미의 협동적 필터링 에이전트라 말하기 어려운 이름뿐인 경우가 대부분이며 협동적 필터링 에이전트 시스템이 구축되어 있는 전자상거래 사이트에 통합하여 사용할 수 있는 add-on 성격의 제품은 거의 없는 상태이다. 실제 협동적 필터링 알고리즘에 의해서 어떤 사람의 취향이나 기호에 맞는 아이템을 추천하고 있는 사이트는 찾아보기 힘들다. 따라서 평가이력을 이용하여 고객의 취향이나 기호에 맞는 상품을 추천해 주는 진정한 의미의 협동적 필터링 에이전트 시스템의 필요성이 대두되고 있으며, 이와 더불어 Who bought 기능 등을 가미한 에이전트 시스템은 큰 시장성과 경쟁력을 가질 것으로 보여진다.

2.2 협동적 필터링 알고리즘

누가 무엇을 구입했는지에 대한 정보를 통해 앞으로 누가 무엇을 사게 될 것인지를 예측할 수 있다. 또한 과거 유저들의 아이템에 대한 흥미도를 이용하여 미래에 누가 무엇에 관심을 가지게 될지 예측할 수도 있다. 이처럼 과거 유저들의 흥미도에 의해 특정 유저의 아이템에 대한 흥미도를 예측하여 추천하는 것을 협동적 필터링이라고 한다. 자동화된 협동적 필터링은 주어진 도메인에서 아이템에 대한 판단을 수집하고 같은 취향을 가지거나 또는 같은 정보 요구를 공유하는 사람들을 매칭시키는 작업으로 이루어진다. 협동적 필터링 시스템 사용자들은 그들이 구입한 아이템에 대한 분석적 판단 그리고 의견들을 공유한다. 이렇게 함으로써 다른 유저가 어떤 아이템을 선택할 지에 대한 보다 좋은 결정을 가능하게 해준다. 협동적 필터링 시스템은 또한 개별화된 아이템 추천을 가능하게 해준다. 협동적 필터링 알고리즘으로는 Tapestry 알고리즘[6], GroupLens 알고리즘, Best N 알고리즘, 그리고 MindReader 시스템에서 사용한 GroupLens 알고리즘과 클러스터링[7]을 접목시켜 만든 알고리즘[8]이 있다.

3. MindReader System

3.1 기능 및 특징

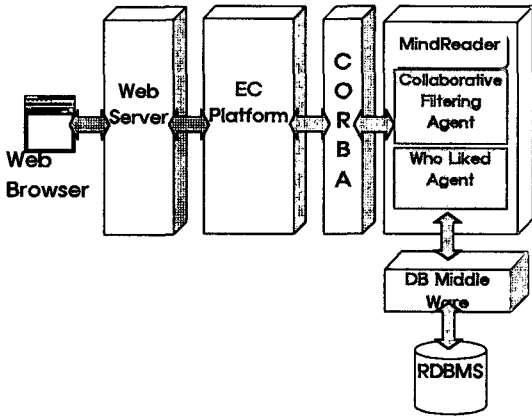
MindReader 시스템은 사이트를 방문한 고객의 취향이나 성향을 파악하여 이에 맞는 상품을 추천해주는 자동화된 추천 에이전트 시스템으로서 협동적 필터링 엔진과 WhoLiked 엔진으로 구성되어 있다. 먼저 협동적 필터링 엔진에서는 각 사용자들의 구매 이력이나 평가이력, 액션 등으로 그 사용자의 취향이나 기호를 파악하여 그에 맞는 아이템들을 자동으로 추천해준다. 추천을 요구할 때마다 다른 아이템들을 추천하도록 동적 변화기능을 갖추었고 현재 추천된 아이템의 예상되는 평가 점수를 통해서 호감도 예측을 할 수 있도록 하였다. 또한 특정 카테고리에 속한 아이템들을 추천할 수 있도록 하였다. WhoLiked 엔진에서는 현재 검색한 아이템에 관심을 가진 사용자들이 가장 많이 관심있어한 다른 아이템을 추천할 수 있도록 하였고 같은 저자가 쓴 다른 책과 같이 현재 아이템과 관련이 있는 아이템을 추천할 수 있도록 하였다.

이 시스템의 특징으로는 첫째, 정확하고 빠른 예측 및 추천을 할 수 있다. [8]에서 제안한 개선된 협동적 필터링 알고리즘을 이용하여 정확하고 빠른 예측 및 추천을 가능하게 하였다. 둘째, 주요 플랫폼들과의 통합을 용이하게 하였다. 셋째, CORBA 를 사용하여 확장성을 강화하고 동시 사용자 수에 상관없이 빠른 성능을 보이도록 하였고, 넷째로 컴포넌트 기반의 구조이기 때문에 신속하면서도 간단하게 시스템을 구현할 수 있고 이로 인해 소요시간과 비용을 절감할 수 있다.

3.2 시스템 구조 및 모듈

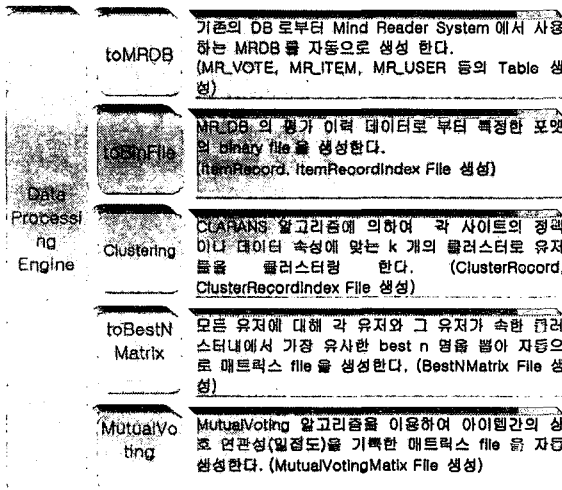
(그림 1)과 같이 MindReader 시스템은 add-on 기능을

갖도록 하기위해 CORBA 와 DB Middle Ware 를 사용한 개방형 구조를 채택하였다. Server 는 NT 플랫폼과 UNIX 플랫폼 2 개의 버전으로 이루어져 있다. CORBA 기반의 오브젝트 서버를 사용하여 분산 구조와 확장성을 갖도록 하였고, Orbix 와 Rogue Wave 등의 Middle Ware 를 사용하여 어떤 시스템이든 통합을 용이하게 하였다.



(그림 1) MindReader 시스템 구성

모듈로는 핵심엔진 모듈과 데이터 처리 모듈로 나누어 볼 수 있다. 핵심엔진 모듈에는 협동적 필터링 엔진과 WhoLiked 엔진이 있다. 데이터 처리 모듈로는 (그림 2)와 같이 데이터 처리 엔진이 있다.



(그림 2) 데이터 처리 모듈

3.3 추천과정 및 인터페이스

MindReader 시스템에서 추천하는 과정을 살펴보면 다음과 같다.

- 1) 사용자 로그인
- 2) 아이템에 대한 직접적인 평가 이력 및 액션 정

- 보수집
- 3) 평가 이력이 유사한 사용자 그룹 선택
 - 4) 선택된 그룹 내 사용자와의 유사성 계산
 - 5) 높은 호감도를 보일 것으로 예측된 아이젠크 추천
 - 6) 이후 로그인 시 자동으로 매번 다른 추천 수행
 - 7) 관련 아이젠크 추천
- 여기서 3), 4), 5)는 협동적 필터링을 통해 수행되고 7)은 WhoLiked 를 통해 수행되게 된다.

```
interface CollaborativeFilteringAgent {
    long predict(in unsigned long user_id, in unsigned long item_id, out float predicted, out string errMsg);

    long recommend(in unsigned long user_id, out long resultCnt, out MRItemList itemList, out string errMsg);

    long recommendByCategory(in unsigned long user_id, in unsigned short category_id, out long resultCnt, out MRItemList itemList, out string errMsg);

    long registerNewUser(in unsigned long user_id, in string login_id, out string errMsg);

    long saveRatingHistory(in unsigned long user_id, in unsigned short item_id, in unsigned short score, out string errMsg);

    long updateRatingHistory(in unsigned long user_id, in unsigned short item_id, in unsigned short score, out string errMsg);

    long getRatingHistory(in unsigned long user_id, in long check, out MRItemRecords items, out long total, out string errMsg);

    long getItemList(in unsigned long user_id, in long maxCnt, out long resultCnt, out LongList itemList, out string errMsg);
};

interface WhoLikedAgent {
    long whoLiked(in unsigned long id, in long key, out long resultCnt, out LongList itemList, out string errMsg);

    long getRatingAvg(in unsigned long item_id, out float ratingAvg, out string errMsg);
};
```

(그림 3) 협동적 필터링 에이전트와 WhoLiked 에이전트의 인터페이스 코드부분

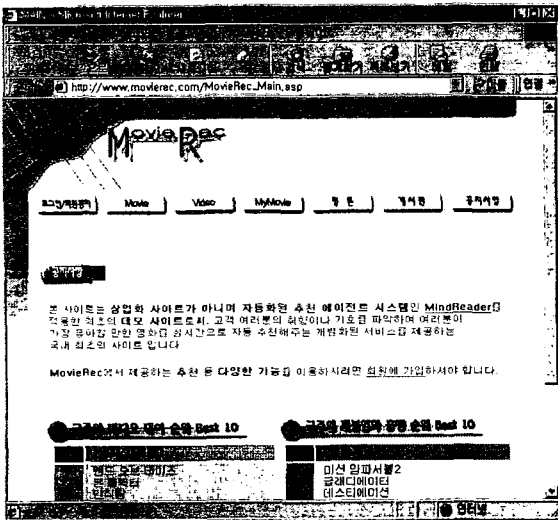
기존의 GroupLens 알고리즘과 Clustering 기법을 접목하여 만든 협동적 필터링 알고리즘을 사용한 인터페이스 코드부분과 아이젠크간의 상호관계를 쉽게 추출해 낼 수 있는 WhoLiked(MutualVoting) 알고리즘을 사용한 인터페이스 코드부분을 살펴보면 (그림 3)과 같다.

협동적 필터링 에이전트 인터페이스에서 Predict()

부분에서는 특정 사용자(user_id)의 특정 아이템(item_id)에 대한 예측 호감도를 리턴하게 되고, recommend() 부분에서는 사용자(user_id)가 좋아할 만한 아이템들을 리턴한다. RecommendByCategory() 부분에서는 사용자(user_id)가 특정 카테고리(category_id)에 속한 아이템 중에서 좋아할 만한 아이템들을 리턴한다. RegisterNewUser()는 사용자(user_id)를 mr_user 에 등록시킨다. SaveRatingHistory() 부분은 특정 사용자(user_id)가 특정 아이템(item_id)에 대해 평가한 점수를 mr_vote 에 입력한다. UpdateRatingHistory()는 특정 사용자(user_id)가 특정 아이템(item_id)에 대해 평가한 점수를 mr_vote 에 업데이트한다. GetRatingHistory()는 특정 사용자(user_id)의 평가 이력을 리턴한다. getItemList()는 특정 사용자(user_id)가 평가하지 않은 특정 수(maxCnt)의 아이템들을 임의로 뽑아 그 리스트를 리턴한다.

다음으로 WhoLiked 에이전트 인터페이스에서 whoLiked()는 특정 아이템(item_id)와 관련된 아이템 리스트를 리턴하고, getRatingAvg() 부분은 특정 아이템(item_id)에 대한 사용자들의 평균 평가치를 리턴한다.

이상을 바탕으로 MindReader 시스템을 실제 구현한 데모사이트(www.movieRec.com)는 (그림 4)와 같다.



(그림 4) MindReader 데모 사이트

4. 결론

MindReader 시스템은 사이트를 방문한 고객의 취향이나 성향을 파악하여 이에 맞는 상품을 추천해주는 자동화된 추천 에이전트 시스템으로서 펄동적 필터링 엔진과 WhoLiked 엔진으로 구성되어 있다. 이 시스템에서 사용한 알고리즘으로는 기존의 GroupLens 알고리즘과 Clustering 기법을 접목시켜 만든 알고리즘을 사용하였다. 좀 더 정확하고 빠른 예측 및 추천을 할 수 있도록 하였으며 add-on 기능과 컴포넌트 기반의 구조라는 장점을 통해 다른 플랫폼들과의 통합을 용이하게 하였다.

앞으로의 과제로는 현재 C++로 되어있는 server 를 java 로 포팅하여 좀 더 유연하고, 쉽게 통합할 수 있도록 하고, 알고리즘에 대한 지속적인 연구와 테스트를 통해 추천의 질과 반응 시간 모두를 향상시키도록 해야 한다. 또한 웹에서 사용자가 행한 행동에 수집함에 있어서 각 행동에 대한 가중치 부여 등에 대한 최적화 연구를 통해 추천의 질을 향상시킬 수 있도록 해야 하겠고, 전자상거래 사이트뿐만 아니라 다른 도메인(Call Centers, Marketing Campaign 등)에서 적용할 수 있는 제품에 대한 연구도 요구된다.

참고문헌

- [1] Oracle, Oracle iMarketing Release 3i, <http://www.oracle.com>
- [2] J. B. Schafer, J. Konstan, and J. Riedl, "Recommender Systems in E-Commerce," Proceedings of the ACM Conference on Electronic Commerce, November 3-5, 1999.
- [3] P. Resnic, N. Iacocou, M. Sushak, P. Bergstrom, and J. Riedle, "GroupLens : An Open Architecture for Collaborative Filtering of Netnews," Proceedings of the Computer Supported Collaborative Work Conference, 1994.
- [4] J. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, and J. Riedl, "GroupLens : Applying Collaborative Filtering to Usenet News," Communications of the ACM Vol. 40, No. 3, pp. 77-87, 1997.
- [5] Macromedia eBusiness <http://www.likeminds.com>
- [6] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using Collaborative Filtering to Weaves an Information TAPESTRY," Communications of the ACM, Vol. 35, No. 12, pp. 61-70, 1992.
- [7] R. T. Ng and J. Han, "Efficient and Effective Clustering Methods for Spatial Data Mining," Proceedings of the Int. Conf. On Very Large Database, Santiago, Chile, pp. 144-155, 1994.
- [8] 황병연, "개선된 추천을 위해 클러스터링을 이용한 협동적 필터링 에이전트 시스템의 성능," 한국정보처리학회 논문지, 제 7 권, 제 5S 호, pp. 1599-1608, 2000.