

산학프로젝트

(신뢰 네트워크를 기반으로 평점 행렬을
채워나가는 Imputation 방법의 성능 검증 실험)

결과보고서

2015. 1. 20.

한양대학교

컴퓨터공학부 특성화사업단

신뢰 네트워크를 기반으로 평점 행렬을 채워나가는 Imputation 방법의 성능 검증 실험

데이터 및 지식 공학 (김상욱 교수님) 연구실 작성자: 차정민

1. 프로젝트 기간

2014.07.01.~2015.1.20

일련 번호	세부 내용	세부 추진 일정 (월)							비고
		1	2	3	4	5	6	7	
1	논문 서베이								
2	아이템 기반 추천 논문								
3	Trust network 기반 추천 논문								
4	Trust-based data imputation 구현 설계								
5	Trust-based data imputation 구현								
6	MyMediaLite 이해 및 적용								
7 8	실험 및 결과 도출								

2. 프로젝트 필요성 및 목적

추천 시스템은 전자상거래, 영화, 음악, 도서, 뉴스 등의 사이트에서 개별 사용자가 선호할 가능성이 높은 아이템을 제공하기 위한 기술이다. 이 기술은 사용자, 상품의 프로필 정보와 사용자가 상품에 대한 평가, 구매, 검색 기록 등을 분석하여 사용자가 과거에 평가하지 않은 상품에 대해 평가를 예측하거나, 상품에 대한 선호 또는 구매 할 가능 여부를 예측한다. 최근, 웹 사이트를 통해서 거래 또는 제공되는 아이템 및 콘텐츠의 종류와 수가 크게 증가함에 따라 개별 사용자의 취향에 적합한 아이템을 추천해주는 추천 시스템 기술이 큰 주목

을 받고 있다.

이러한 추천 시스템은 1990년 중반에 협업 필터링에 관한 논문이 처음 등장하면서 중요한 연구분야로서 자리잡게 되었다. 그 이후로, 산학계 전반에 걸쳐 추천 시스템에 대한 다양한 연구 활동이 진행되어 왔고, 이 분야에 대한 관심은 여전히 높은데, 이 연구분야에 해결해야 할 문제들이 다양하고, 과부화된 정보를 분석해 개인화된 추천, 콘텐츠, 서비스를 제공할 추천시스템에 실질적으로 적용시킬만한 방안이 많이 개발되지 않았기 때문이다.

산학계 전반에 걸쳐 추천 시스템 연구가 계속 진행되었지만, 현 세대의 추천 시스템은 실생활에 존재하는 넓은 분야에 적용시킬 수 있고, 더 효율적인 새로운 추천 방법을 찾는 것에 대한 개선을 필요로 한다. 이러한 개선들은 사용자의 행동이나 추천할 아이템에 대한 정보를 더 나은 방법으로 명시하는 것, 고급화된 추천 모델링 방법, 다양한 문맥상들의 정보들을 추천 프로세스에 통합시키는 것, 추천 시스템의 성능을 더 효율적으로 만들 측정 방법과 연관된 더 효율적인 추천 방법 등을 포함한다.

현 세대의 추천 시스템에서 사용되는 추천 방법은 크게 세가지 중요 카테고리로 분류가 되는데, 콘텐츠 기반, 협업 기반 그리고 하이브리드 추천 연구 방법이 있다. 콘텐츠 기반 추천은 ‘사용자는 자신이 과거에 선호했던 아이템과 유사한 아이템을 추천받는’ 방법이다. 협업 기반 추천은 ‘사용자는 선호와 취향이 유사한 사람들이 과거에 좋아했던 아이템들을 추천받는’ 방법이다. 그리고, 하이브리드 접근법은 협업 기반과 콘텐츠 기반 방법을 조합한 방법이다. 이들 각각은 한계점을 가지고 있다.

3가지의 추천 연구 방법 중에서 본 프로젝트의 관심이 되는 방법은 협업 기반 추천 방법이다. 협업 기반 추천 방법에는 3가지의 한계점이 있는데, ‘새로운 사용자 문제’, ‘새로운 아이템 문제’, ‘데이터 희소성 문제’가 있다. 이러한 한계점들을 해결해나가면서 PMF와 같은 기존의 추천 방법의 성능을 향상시킬 수 있도록 평점 행렬을 채워나가는 새로운 imputation 방법을 제시하는 것이 본 프로젝트의 목표이다.

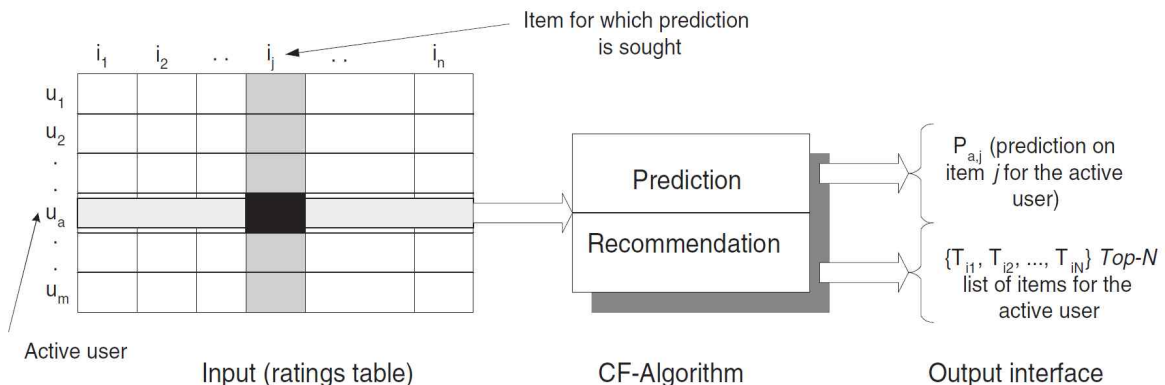
3. 프로젝트 내용 요약

현 세대의 추천 시스템 방법 중 협업 기반 추천 방법은 3가지의 한계점을 가지고 있는데, ‘새로운 사용자 문제’, ‘새로운 아이템 문제’, ‘데이터 희소성 문제’ 사용자의 선호와 취향을 이해하고 신뢰할만한 추천을 해주기 전에는, 그 사용자가 평가한 아이템의 수가 충분해야 한다. 하지만 새로운 사용자는 평가를 거의 하지 못한 경우가 많아서, 정확한 추천을 얻기가 어렵다는 점이 있는데, 이러한 문제를 ‘새로운 사용자 문제’ 혹은 cold-start user 문제라고도 한다.

새로운 아이템이 추천 시스템에 추가될 때도 이와 유사한 문제가 발생하게 된다. 협업 추천 시스템은 사용자의 선호를 기반으로 추천을 하게 되는데, 새로운 아이템이 충분한 사용자에게 평가되기 전에는, 추천 시스템이 그 아이템을 추천하기 어려울 수 있다. 이러한 문제를 ‘새로운 아이템 문제’라고 한다.

몇몇 추천 시스템에서는 얻어진 평점의 수가 필요한 양에 비해서 매우 적을 수도 있다. 영화 추천 시스템을 예를 들어보면, 그 시스템에는 매우 적은 수의 사람들이 평가한 영화가 매우 많이 있을 수 있다. 만약, 이들에 대한 평점이 높다고 하더라도, 이 영화들은 거의 추천되지 않을 것이다. 또, 다른 사람들에 비해 유별날 취향을 갖고 있는 사람들은 유사한 취향을 가진 사람들을 찾기가 어려워 좋은 추천을 받기가 어렵게 된다. 이들 문제는 결국 평점 행렬의 빈 공간의 수가 많아 발생하는 문제들로, 이러한 문제들을 ‘데이터 희소성 문제’라고 한다.

협업 기반 추천 방법을 포함한 다양한 추천 방법들은 데이터 희소성과 cold-start user 문제 때문에 결과의 정확도가 낮아지는 문제를 야기하곤 한다. 이러한 문제를 해결하고자, 본 프로젝트에서는 평가되지 않은 평점을 가능한 값으로 채워서 평점 행렬의 빈 부분을 채워나가는 imputation 방법을 제시한다. 협업 필터링 추천 방법은 아래와 같이 평점 행렬이 입력값으로 주어지면, Prediction 혹은 Recommendation 협업 필터링 알고리즘을 통해 예측한 평점 혹은 추천받을 아이템의 리스트가 반환되는 형태를 띄고 있다. 본 프로젝트에서 제안하는 방법은 평가되지 않은 평점을 예측하여 평점 행렬을 채워나가는 것이므로, 협업 필터링 관점에서는 Prediction 알고리즘을 돌리는 것으로 볼 수 있다. 여기서, 평가되지 않은 평점을 정확하게 추정하기 위해서 신뢰 네트워크(trust network)를 사용한다. Epinions 데이터 세트를 사용하여 실험을 진행하고, 본 프로젝트에서 제시하는 방법이 기존 방법들에 비해 더 좋은 추천 정확도를 제공할 수 있음을 보일 것이다.



▲ [그림 1] 협업 필터링 방법의 프로세스

많은 추천 방법들은 한 목표 사용자가 아직 평가하지 않은 아이템들의 평점을 예측하는 방법을 제시해왔다. 그들 중 PMF(Probabilistic Matrix Factorization) 모델은 꽤 효과적인 것으로 알려져 있는데, 이러한 모델들은 대부분의 사용자가 적은 수의 아이템에만 평가한 경우 (데이터 희소성 문제) 낮은 정확성을 보이는 단점이 있었다. 특히, 이러한 문제는 cold-start 사용자들의 평점을 정확히 예측해야 하는데 중요한 문제점이 되었다.

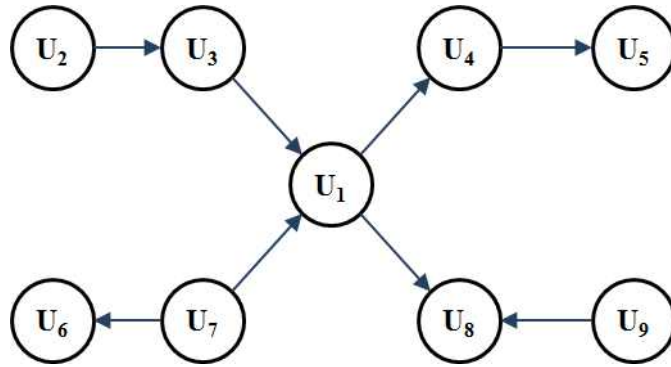
데이터 희소성과 cold-start 사용자 문제를 해결하기 위해서, 기존에 존재하는 imputation 방법들은 평가되지 않은 평점을 가능한 값으로 변경하였다. 하지만, 대부분의 방법들이 소셜 네트워크와 같은 추가적인 정보를 간과하였고, 그들 방법에만 적합하게 설계되어 있어서 PMF와 같은 다른 추천 방법 모델에 적용될 수 없다.

본 프로젝트에서 제안하는 imputation 방법은 다음과 같은 중요한 특징을 갖고 있다.

- (1) trust network를 추가 정보로 활용한다.
- (2) PMF를 포함한 기존에 존재하는 다른 추천 방법들에 적용될 수 있다.

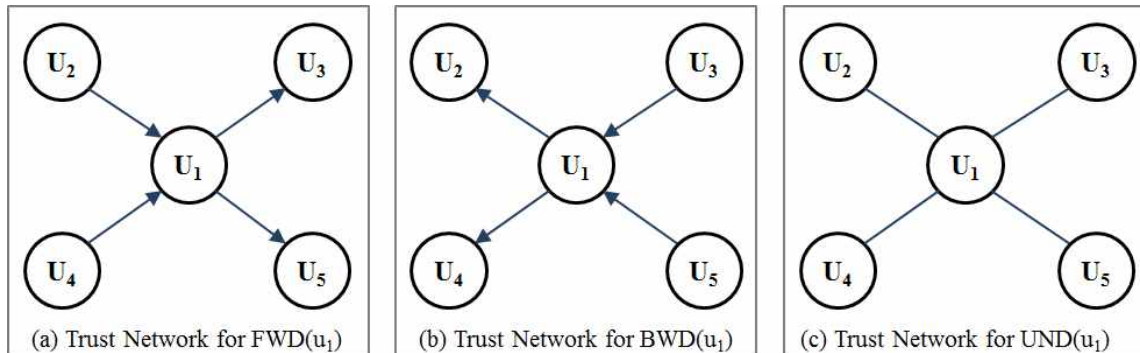
본 방법은 사용자에게 의해 평가되지 않은 아이템의 평점을 추정할 때 사용자의 신뢰할 수 있는 이웃(reliable neighbors)이 평가한 평점을 집계하여 추정한다. 신뢰할 수 있는 이웃을 찾기 위해서 신뢰 네트워크를 사용하는데, 신뢰 네트워크는 소셜 네트워크의 종류 중 하나로, 사람들 사이의 신뢰 관계를 표현한다. 신뢰할 수 있는 이웃의 평점을 집계하여 추정하는 이유는 신뢰 네트워크에서 연결된 사람들은 비슷한 선호를 가지는 경향이 있기 때문이다. 게다가, 본 방법은 충분한 수의 이웃으로부터 평점 행렬을 채운다. 본 방법 덕분에, 더욱 더 밀도가 높은 평점 행렬을 얻을 수 있고, 그로 인해 본 방법을 PMF를 포함한 대부분의 추천 방법에 적용시킴으로써 더 높은 정확도를 얻을 수 있다.

신뢰 네트워크에서 신뢰 관계란 trustor가 trustee를 신뢰하는 관계를 나타낸다. 신뢰 관계는 아래 그림과 같이 그래프로 표현할 수도 있는데, 각 마디는 사용자를 의미하고, 간선($U_1 \rightarrow U_4$)는 “사용자 U_1 이 사용자 U_4 를 신뢰한다”는 것을 표현한다. 본 방법에서 신뢰할만한 이웃은 사용자를 신뢰하는 trustor들과 사용자의 trustee들로 정의한다. 신뢰 네트워크를 사용하기 때문에 본 방법에서는 trustor와 trustee는 유사한 선호를 가지고 있을 거라는 가정을 하고 진행하게 된다.



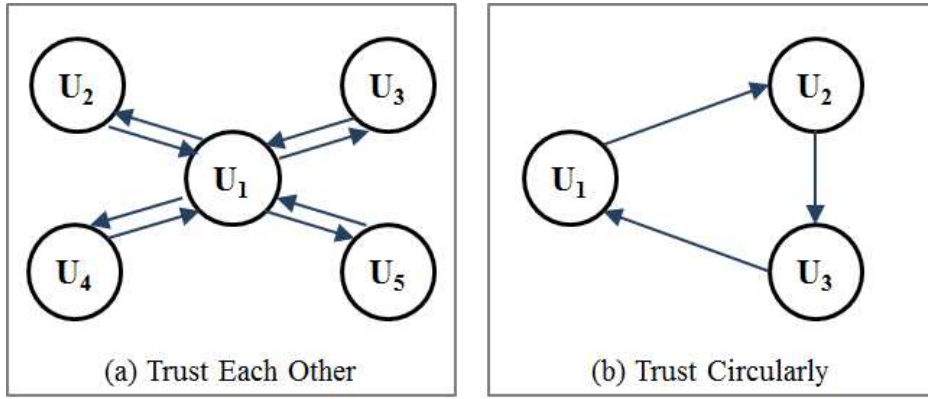
▲ [그림 2] 신뢰 네트워크의 한 예시

신뢰할만한 이웃을 찾아야 할 때, 특정 사용자 U_1 에 대해서 신뢰 네트워크를 간선의 앞 방향, 뒤 방향 또는 양 방향을 모두 살펴보는 방법이 있을 수 있다. 신뢰 네트워크에서 간선의 앞 방향으로 도달할 수 있는 사용자는 trustee가 되고, 간선의 뒤 방향으로 도달할 수 있는 사람은 trustor가 된다. 양 방향을 모두 살펴 보는 경우 trustor와 trustee를 모두 포함하여 이웃으로 설정하게 된다. 아래의 그림은 신뢰 네트워크에서 사용자 간의 신뢰 관계 간선 방향을 나타내고 있다.



▲ [그림 3] 간선 방향에 따른 신뢰 네트워크

또한, 신뢰 네트워크에서는 두 사용자가 서로를 신뢰하는 관계에 있을 수도 있고, 사용자 간에 신뢰 관계에서 순환이 생길수도 있다. 이 경우, 한 사용자에서 다른 사용자까지의 거리가 여러 개로 결정될 수 있는데, 신뢰 네트워크의 신뢰 관계에서 순환이 생겨 거리를 결정하기 어려울 경우에는, 사용자와 사용자 간의 거리를 둘 사이의 가장 짧은 거리로 한다.



▲ [그림 4] 신뢰 네트워크에서 서로를 신뢰할 경우와 순환하게 신뢰하는 경우

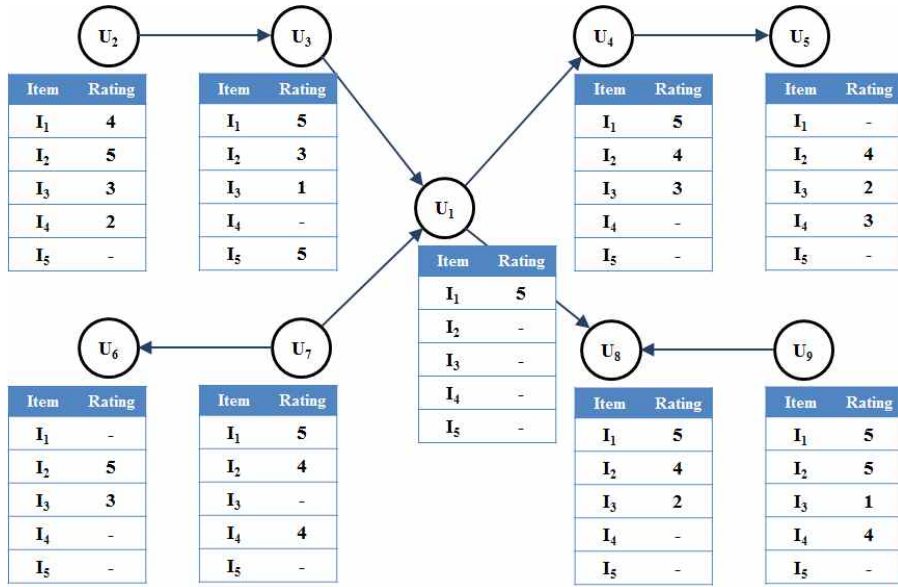
신뢰 이웃을 정할 때, 신뢰 네트워크 그래프에서 간선의 앞 방향으로 따지는 것만큼이나 뒤 방향으로 따지는 것 또한 중요하다. 사용자로부터 간선의 앞, 뒤 모든 방향으로 제한 거리 δ 안에서 접근할 수 있는 이웃의 집합을 $BID_\delta(u)$ 으로 표시하고, 앞 방향으로 제한 거리 δ 안에서 접근할 수 있는 이웃의 집합을 $FWD_\delta(u)$ 라고 표현한다. 위 그림 [그림 1] 에서 거리 threshold δ 를 2로 가정하면, 사용자 U_1 에 대해서, $FWD_2(U_1) = \{U_4, U_5, U_8\}$ 과 $BID_2(U_1) = \{U_2, U_3, U_4, U_5, U_6, U_7, U_8, U_9\}$ 를 얻을 수 있다.

사용자 u 의 아이템 i 에 대한 평점을 예측하기 위해서, 신뢰 네트워크를 통해 $BID_\delta(u)$ 를 찾고, 그들의 아이템 i 에 대한 평점을 이용하여 $r'_{u,i}$ 를 다음과 같이 추정한다.

$$r'_{u,i} = \bar{u} + \frac{\sum_{v \in \{v | r_{v,i} \neq 0, v \in BID_\delta(u)\}} w_{uv} (r_{vi} - \bar{v})}{\sum_{v \in \{v | r_{v,i} \neq 0, v \in BID_\delta(u)\}} w_{uv}}$$

▲ [공식 1] 신뢰 네트워크로부터 얻은 이웃을 활용하여 평가되지 않은 평점 추정

본 방법에서는 신뢰할만한 이웃이 평가한 평점의 수가 평점 추정의 정확도를 결정하므로, 적은 수의 이웃에 의한 추정값은 부정확할 수도 있다. 이러한 문제에 대한 사례가 다음 그림에 묘사되어 있다. 각각의 원은 사용자를 의미하고, 간선은 신뢰 관계를 표현한다. 원 아래의 표는 그 사용자가 평가한 아이템의 평점을 표시하고 있다. 제한 거리가 2일 때, 간선의 양쪽방향으로 얻은 이웃의 집합은 $BID_2(U_1) = \{U_2, U_3, U_4, U_5, U_6, U_7, U_8, U_9\}$ 이다. 아이템 I_2 에 대해서는 8명의 신뢰 이웃이 그 아이템에 대해 모두 평가하였지만, I_5 의 경우, 오직 U_3 만이 그 아이템에 대해 평가하였기 때문에 imputation을 위해 U_3 의 평점만 활용할 수 있고, 이는 믿음직하지 못한 정보가 된다.



▲ [그림 5] 이웃들의 평점을 이용해서 U_1 의 평점을 채워나가는 예시.

아이템 I_5 는 이웃들의 평점의 수가 부족해서 평점을 추정하여 채울 수 없다.

이를 해결하기 위해서, 본 방법에서는 imputation 프로세스를 진행하는 동안, 충분하지 않은 이웃에 의해 평가된 아이템의 평점을 채워나가는 것은 무시하도록 한다. 충분한 이웃에 의해 평가된 나머지 아이템들의 집합을 후보 아이템 집합(candidate item set)이라고 하며, 사용자 u 에 대한 후보 아이템 집합 $C_\theta(u)$ 를 만들기 위해서, 사용자 u 에 대한 이웃들이 아이템에 대해서 평가한 수를 기준으로 아이템을 정렬하여, 상위 θ 퍼센트의 아이템만을 선택한다.

제안된 imputation 방법은 사용자 u 의 아이템 i 에 대한 평점 $r'_{u,i}$ 을 다음과 같이 추정한다.

$$r'_{u,i \in \{i | i \in C_\theta(u)\}} = \bar{u} + \frac{\sum_{\{v \in v | r_{v,i} \neq 0, v \in BID_\delta(u)\}} w_{u,v} (r_{v,i} - \bar{v})}{\sum_{\{v \in v | r_{v,i} \neq 0, v \in BID_\delta(u)\}} w_{u,v}}$$

▲ [공식 2] [공식 1]에서 후보 아이템 집합을 고려한 공식

본 방법으로 더 밀도가 높은 평점 행렬 $R = (r_{u,i})_{M \times N} + (r'_{u,i})_{M \times N}$ 을 얻을 수 있다. M, N 은 각각 사용자와 아이템의 수를 의미하고, $r_{u,i}$ 는 기존 평점을, $r'_{u,i}$ 는 추정된 평점을 의미한다. 결과로 얻은 밀도행렬은 기존의 평점 행렬과 완전히 똑같은 형태를 띄고 있어 PMF 모델을 포함한 다른 추천 방법이 정확도를 향상시키기 위해 이 행렬을 사용할 수 있다.

4. 프로젝트 결과물

상세한 그림 또는 표를 활용하여 설명
필요한 경우, 결과물 수행 사진을 활용하여 설명

4.1 실험을 수행할 프로그램 구현 설계

신뢰 네트워크를 활용하여 평점 행렬을 채워나가는 역할을 수행할 프로그램을 설계, 구현을 하였는데 아래의 [그림 4]는 프로그램을 설계할 당시 작성한 UML 다이어그램을 나타내는 그림이다. 실험에 영향을 끼치는 중요한 주제들로는 크게 4가지가 있는데, 신뢰 네트워크, (신뢰할만한) 이웃 집합, 후보 아이템 집합, 그리고 평점 행렬이 있다. 이 4가지의 주제들이 있다는 점을 고려해서 각각의 기능과 역할을 제대로 수행할 수 있는 4가지 클래스를 작성하였다.

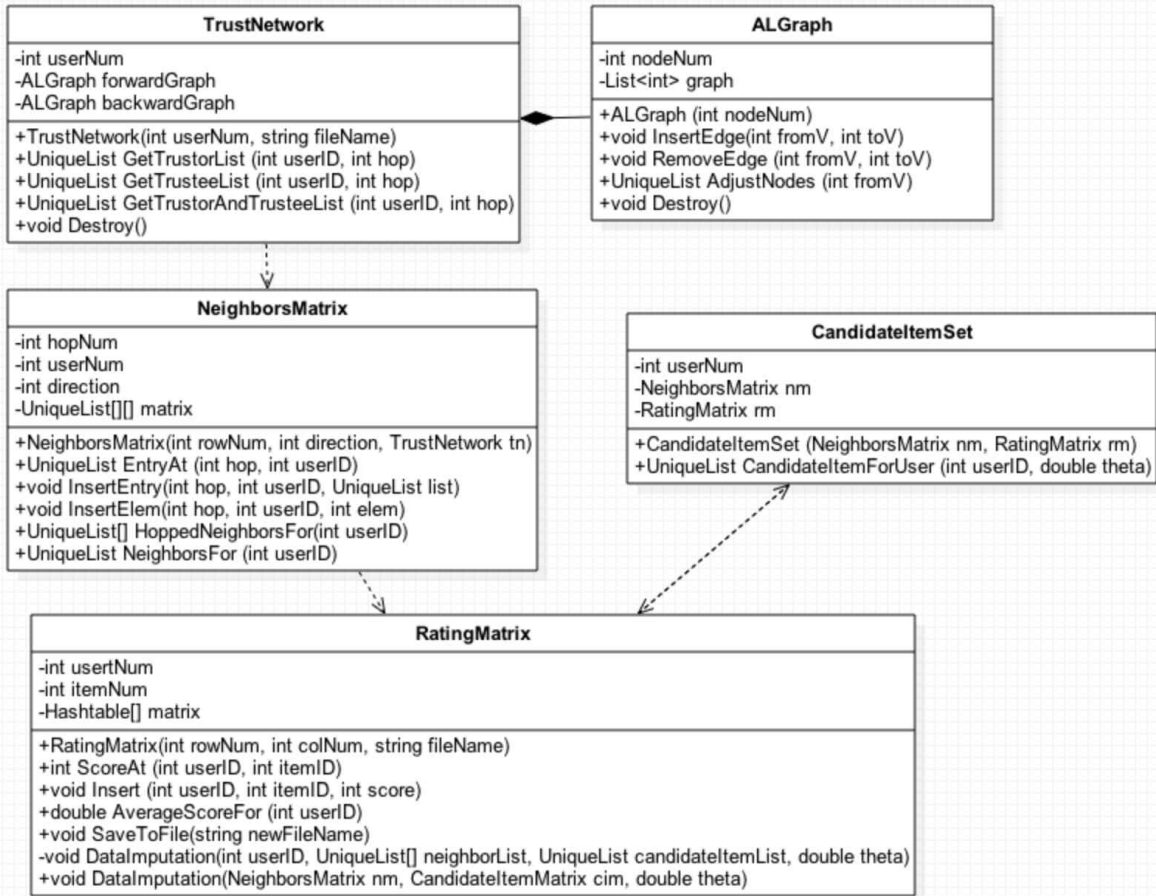
신뢰 네트워크(TrustNetwork) 클래스는 신뢰 관계 정보를 포함하고 있어야 하며, 이웃 집합을 만드는데 사용할 수 있어야 한다. 신뢰 네트워크는 그래프로 구성이 되어 있기 때문에, 인접 행렬 기반 그래프를 활용하여 앞 방향의 간선 정보를 담고 있는 그래프와 뒷 방향의 간선 정보를 담고 있는 그래프를 멤버 변수로 가지게 된다. 따라서, 실험에서 간선의 방향을 변경하여 이웃을 찾고자 할 때, 두 그래프 중 하나 혹은 동시에 두 개를 선택하여 사용할 수 있다. 신뢰 네트워크는 신뢰할만한 이웃 집합을 만들 때 사용해야 하므로, 특정 사용자로부터 특정 거리에 도달할 수 있는 사용자들의 리스트를 알려주는 역할을 해야 하므로 GetTrusterList, GetTrusteeList, GetTrusterAndTrusteeList와 같은 메서드를 구현하였다.

(신뢰할만한) 이웃 집합(NeighborsMatrix) 클래스는 신뢰 네트워크를 기반으로 각 사용자별 이웃 집합의 정보를 가지고 있어야 한다. 여기서, 각 사용자별로 그리고 거리 별로 그 이웃 집합의 정보를 가지고 있어야 하기 때문에 원소가 리스트형태인 2차원 배열을 멤버변수로 가지게 되었다. 그리고, 이웃 집합을 결정할 때, 간선의 방향을 정해야 신뢰 네트워크에서 그 간선 방향으로 이웃을 결정하게 되므로, 간선 방향 정보를 멤버 변수로 가지고, 클래스를 생성할 때에 입력받은 간선 방향 정보를 기준으로 신뢰 네트워크를 조사하여 이웃 집합을 결정하게 된다. 또한, 이웃 집합은 후보 아이템 집합을 결정할 때, 그리고 평점 행렬을 채워나갈 때 사용해야 하므로 특정 사용자에 대한 이웃들의 집합을 반환하는 역할을 하는 메서드인 HoppedNeighborsFor, NeighborsFor 메서드를 구현하였다.

후보 아이템 집합(CandidateItemSet) 클래스는 각 유저의 후보 아이템을 Candidate threshold θ 값에 맞추어 선정해야 한다. 후보 아이템은 신뢰 이웃에 의해 평가된 아이템들 중에서 선정이 되고, 신뢰할만한 이웃에 의해 평가되었는지를 확인하려면 평점 행렬을 확인해야 한다. 따라서, 이 클래스는 신뢰할만한 이웃 집합 클래스와 평점 행렬 클래스의 인스

턴스 포인터들을 멤버변수로 가지고 있다. $C_\theta(u)$ 를 구하기 위해서 두 가지 인자 (candidate threshold θ , 사용자 번호)를 받고 아이템 리스트를 반환하는 메서드인 CandidateItemFor 메서드를 구현하였다.

마지막으로, 평점 행렬 클래스는 사용자가 아이템에 대해 평가한 평점들의 정보를 갖고 있는 클래스이다. 사용자 수와 아이템 수가 많고, 평점 데이터가 희소하므로 이를 이차원 배열로 정보를 유지하는 것은 비합리적이다. 따라서, 이 클래스는 사용자 수를 길이로 하는 해쉬테이블의 배열을 이용하여 평점 데이터의 정보를 유지하고 있다. 또한, 후보 아이템 집합을 결정할 때, 평점 행렬을 확인해 평점 데이터가 존재하는지를 확인하려면 이 자료구조에 대한 검색 시간도 보장이 되어야 할 것이다. 배열은 인덱스에 대한 무작위 접근이 가능해 검색에 일정한 시간이 소요된다. 또한, 배열의 원소인 해쉬 테이블도 최적의 경우 검색 시간이 일정하게 되므로, 검색 성능도 어느 정도 확보하고, 무의미한 메모리 공간 낭비도 하지 않게 된다. 평점행렬 클래스에는 특정 사용자가 특정 아이템에 평가했는지, 평가했다면 몇점으로 평가했는지를 알려주는 역할인 ScoreAt 메서드를 구현하였다. 그리고 평점 행렬을 채우는 Data imputation 프로세스 중에 특정 사용자가 내린 평점들의 평균값을 빈번하게 알아야 하는 경우가 많으므로 ([공식 1],[공식 2] 참조) 그 평균값을 반환하는 AverageScoreFor 메서드도 구현이 되었다. 그리고, 이웃 집합과 후보 아이템 집합이 주어졌을 때, 평점 행렬에서 평가되지 않은 평점들을 채워나가는 역할인 DataImputation 메서드도 구현이 되어 있다. 이 메서드를 통해 평점 행렬이 평점 데이터를 유지하는 자료구조에 예측된 평점들이 채워지게 되고, 이 모든 평점 데이터들을 파일에 저장하는 메서드인 SaveToFile 메서드도 역시 구현이 되어 있다.



▲ [그림 6] 신뢰 네트워크를 이용한 Data Imputation 실험을 수행한 프로그램의 설계 UML 다이어그램

4.2 실험 환경과 실험 계획

실험을 위해 실세계의 데이터 세트들 3가지를 사용하였다. 그 데이터 중 2개는 Epinions로부터, 마지막 하나는 Ciao로부터 수집하였다. Epinions1 데이터 세트는 다른 데이터 세트에 비해서 더 많은 유저를 포함하고 있고, Epinions2 데이터 세트는 다른 데이터 세트들보다 아이템과 평점을 더 많이 포함하고 있다. 신뢰 관계의 수는 Epinions1 데이터 세트가 가장 많았다. Ciao 데이터 세트의 크기는 다른 두 개의 데이터 세트보다 작다. 아래의 표에서는 cold-start 사용자를 5개의 아이템보다 적게 평가한 사용자로 정의하였다. 많은 영역에서 대부분의 사용자들이 cold-start 사용자임에도 불구하고, Epinions2와 Ciao 데이터 세트는 적은 수의 cold start 사용자만을 포함하고 있는데, 이는 이 데이터 세트들을 사용한 이전의 연구들은 cold start 사용자를 포함하지 않았을 것이기 때문이다. 희소성은 평점 행렬에서 비어있지 않은 원소의 수의 비율을 의미한다. 아래의 표에서 희소성 항목을 통해 데이터 세트들 모두가 매우 희소하다는 것을 알 수 있다.

Statistics	Data set		
	Epinions1	Epinions2	Ciao
Users	49,289	22,166	7,375
Items	139,738	296,277	106,797
Ratings	664,824	916,085	282,619
Trusts	487,002	355,813	111,781
Cold Users	16,910	21	6
Sparsity	0.0001	0.0004	0.0001

▲ [표 1] 데이터 세트의 통계 수치

본 실험에서 제시하는 방법에서는 각 사용자의 신뢰할만한 이웃들을 찾을 필요가 있다. 목표 사용자의 신뢰 이웃들은 신뢰 네트워크에서 거리 threshold 안에서 접근할 수 있는 유저들을 의미한다. 또한, 본 방법은 후보 threshold θ 으로 후보 아이템 집합을 결정한다. 제안된 추천 방법에서 거리와 후보 threshold를 결정하는 2개의 인자는 평점을 채워나가는 것의 정확도를 결정할 뿐만 아니라 추천 예측 결과에도 영향을 끼친다. 이러한 이유로, 두 개의 인자값을 다양하게 만들어 다른 인자 환경에 따른 정확도를 비교하여 가장 최적의 설정값을 얻었다.

예측 정확도를 측정하기 위해서 mean absolute error (MAE) 와 root mean square error (RMSE)를 본 실험의 측정 기준으로 사용하였다. MAE는 예측된 평점들과 실제 평점들의 차이점의 평균값을 계산하는 측정 방법인 반면, RMSE는 큰 오류에 더 중점을 많이 두는 측정 방법이다. 본 실험에서는 5-fold cross validation을 실행하였다. 각각의 fold에는 80%의 평가 데이터를 학습 집합(training set), 20%의 평가 데이터를 검증 집합(test set)으로 사용하였다. 다른 모든 실험에서도 이와 같은 설정을 사용하였다.

MAE, RMSE를 측정하기 위해서 MyMediaLite 라이브러리를 사용하였다. MyMediaLite 라이브러리는 추천 시스템 알고리즘의 라이브러리 중 하나로, MyMediaLite 라이브러리는 평점 예측과 아이템 추천에 대해서 수량적인 측정 수치 값을 얻을 수 있는 루틴을 제공해 주고 있는데, MAE, RMSE, CBD, CBD, prec@N, MAP, NDCG 등을 측정할 수 있다. 또, 간단한 텍스트 기반 입력 포맷을 받아들여서 평점 예측과 아이템 추천 두 가지 작업에 대한 커맨드 라인 툴도 제공한다. 따라서, 제안된 방법으로 평점 행렬을 새로 채워나가는 프로그램을 작성한 후에, MyMediaLite 라이브러리를 사용하여 RMSE, MAE 값을 측정하고 그 값을 토대로 본 제안된 방법에서 정확도 측면에서의 향상이 있었는지를 확인할 것이다.

목적 사용자의 평점을 채워나가기 위해서, 제안된 방법은 간선의 양방향을 거쳐 도달할 수 있는 사용자들을 신뢰할만한 이웃으로 선택한다. 이 방법과는 다르게, 기존의 신뢰 기반 추천 방법은 간선의 앞 방향만을 통해 도달할 수 있는 사용자들만을 고려하여 평점을 추정한다. 신뢰 이웃을 찾을 때, 간선의 앞 방향뿐만 아니라 양 방향을 통해 찾는 것의 효율성

을 확인하기 위해서, 본 실험에서는 두 개의 신뢰 이웃을 비교하는 실험을 수행하였는데, 하나는 양 방향을 통해 얻은 이웃들을 포함하고, 다른 하나는 간선의 앞 방향을 통해 얻은 이웃들을 포함했다. 그리고, cold start 사용자들만을 대상으로 동일한 실험을 수행하였는데, 그들에 대한 평점 예측은 다른 사용자들에 비해 덜 정확할 것이기 때문이었다.

두 인자 값의 최적의 조합을 찾은 후에, 신뢰 네트워크의 사용이 정확한 imputation을 하는데 도움이 되는 것을 보이는 실험을 했다. imputation의 결과가 정확한 것을 보이기 위해서, imputation의 결과로 얻은 imputed rating 값을 사용하여 예측의 정확도를 확인하였다. 이 실험에서는 SimVote라고 불리는 신뢰 네트워크를 고려하지 않는 간단한 imputation 방법과 본 실험에서 제시하는 방법을 비교해보았다. SimVote 방법은 평가되지 않은 평점을 채워나갈 때, 기존 imputation 방법 중 하나인 default voting approach를 기반으로 채워나간다. 본 실험에서 제시하는 방법과 비슷한 수의 평가되지 않은 평점을 채워나가기 위해서, 평가되지 않은 평점 중 무작위로 \hat{I}_{ij} 퍼센트를 뽑았고 SimVote에 의한 평점들만을 채워나갔다. 평점을 예측하기 위해서, SimVote 방법으로 채워진 평점 행렬에 PMF 모델을 적용 시켰다.

마지막으로, 본 실험에서 제안하는 imputation 방법을 사용하는 추천 방법과 기존에 존재하는 추천 방법들과 비교하는 실험을 진행하였다. 이 실험에서는, 비교할 네 개의 기존 추천 방법들이 있는데, 사용자 기반 추천, AutAI, PMF, 그리고 SocialMF가 있다. 사용자 기반 추천 방법을 선택한 이유는 가장 잘 알려져 있으면서 간단한 협업 필터링 모델이기 때문이다. 또한, AutAI은 최근에 개발된 imputation 기반 추천 방법 중 하나이기 때문에 선택되었다. AutAI 방법에 대해서 사용자와 아이템 관점 imputation의 비율을 의미하는 인자인 λ 값을 0.4로 설정한다. 제안한 추천 방법은 평점을 예측하는데 PMF 모델을 사용하므로, 기존의 PMF 모델을 비교 대상 중 하나로 삼았다. 그리고, SocialMF는 행렬 인수분해를 활용하고, 신뢰 네트워크를 사용하여 데이터 희소성을 극복한다는 점 때문에 본 실험에서 제안하는 방법과 유사한 점이 많아서 선택하게 되었다. SocialMF 방법에 대해서 평점을 예측하는데 소셜 네트워크의 영향 정도를 결정하는 인자인 λT 는 0.5로 설정하였다. PMF, SocialMF, 그리고 본 프로젝트에서 제안하는 방법에 5차원 행렬 인수분해를 수행하며 이 방법들에 대해서 인자 λU 와 λV 값을 0.01로 설정한다. 모든 사용자들에 실험을 수행하는 한편, 기존 추천 시스템에 심각한 문제인 cold-start 사용자만 따로 동일한 실험을 수행한다.

4.3 실험 결과

아래의 표들은 다른 인자 설정 아래에서 제안하는 방법을 사용하였을 때, 예측에 대한 정확도를 보여주고 있다. 한 인자는 신뢰할만한 이웃을 결정하는 인자인 거리 threshold δ 값

을 의미하고, 다른 인자는 평가되지 않은 평점들 중에서 추정할 평점의 비율을 결정하는 후보 threshold θ 값이다. 실험은 Epinions1, Epinions2, Ciao 데이터 세트들에 대해서 수행되었고, 각각의 표는 각각의 데이터 세트에서의 실험 결과를 포함하고 있다. 표에서 어두운 셀은 RMSE와 MAE 관점에서 보았을 때 두 threshold 인자 값이 최적의 조건인 경우를 말한다.

첫 번째(Epinions1), 세 번째(Ciao) 실험 결과 표에는, 후보 threshold θ 값이 20%일 때 RMSE값이 최소이고, θ 값이 30%일 때 MAE값이 최소임을 확인할 수 있었다. 반면, 두 번째 실험(Epinions2) 결과 표에서는 θ 값이 10%일 때 RMSE값이, θ 값이 20%일 때 MAE값이 최소임을 확인할 수 있었다. 이것은 Epinions2 데이터 세트가 Epinions1과 Ciao 데이터 세트에 비해서 평점의 수가 더 많기 때문인 것으로 보인다. 대부분의 경우에, 본 실험에서 제안한 방법은 3개의 데이터 세트에서 거리 threshold δ 값이 3 또는 4으로 설정되었을 때, RMSE 혹은 MAE값이 최소가 되었다. 이러한 결과는 거리 threshold δ 값은 3 혹은 4이고, 후보 threshold θ 값은 20%으로 설정하는 것이 대부분의 데이터 세트에서 최고의 정확도를 보이는 것임을 보여준다.

측정값	Candidate Threshold(θ)	Distance Threshold (δ)					
		1	2	3	4	5	6
RMSE	10%	1.1089	1.0941	1.0959	1.0966	1.0968	1.0968
	20%	1.1073	1.0886	1.0876	1.0877	1.0878	1.0879
	30%	1.1072	1.0934	1.0899	1.0891	1.0892	1.0893
	40%	1.1062	1.0962	1.0935	1.0953	1.0950	1.0950
	50%	1.1051	1.1003	1.1001	1.1013	1.1015	1.1023
MAE	10%	0.8666	0.8466	0.8401	0.8392	0.8395	0.8395
	20%	0.8626	0.8370	0.8310	0.8295	0.8292	0.8291
	30%	0.8616	0.8433	0.8271	0.8232	0.8227	0.8226
	40%	0.8626	0.8437	0.8296	0.8279	0.8281	0.8280
	50%	0.8637	0.8450	0.8310	0.8294	0.8284	0.8284

▲ [표 2] Epinions1 데이터 세트에서 Distance Threshold δ 와 Candidate Threshold θ 의 값에 따른 결과

측정값	Candidate Threshold(θ)	Distance Threshold (δ)					
		1	2	3	4	5	6
RMSE	10%	1.1041	1.0900	1.0843	1.0851	1.0855	1.0860
	20%	1.1030	1.0923	1.0865	1.0907	1.0913	1.0916
	30%	1.1013	1.0944	1.0942	1.1018	1.1025	1.1029
	40%	1.0994	1.0969	1.1102	1.1204	1.1210	1.1249
	50%	1.0977	1.1023	1.1519	1.1886	1.1984	1.1995
MAE	10%	0.8631	0.8404	0.8293	0.8281	0.8282	0.8294
	20%	0.8618	0.8460	0.8285	0.8281	0.8287	0.8300
	30%	0.8605	0.8465	0.8321	0.8329	0.8334	0.8353
	40%	0.8590	0.8469	0.8395	0.8421	0.8420	0.8456
	50%	0.8576	0.8501	0.8602	0.8798	0.8794	0.8752

▲ [표 3] Epinions2 데이터 세트에서 Distance Threshold δ 와 Candidate Threshold θ 의 값에 따른 결과

측정값	Candidate Threshold(θ)	Distance Threshold (δ)					
		1	2	3	4	5	6
RMSE	10%	1.0088	0.9899	0.9863	0.9881	0.9893	0.9899
	20%	1.0052	0.9864	0.9813	0.9811	0.9812	0.9816
	30%	1.0045	0.9867	0.9816	0.9826	0.9827	0.9831
	40%	1.0039	0.9895	0.9830	0.9834	0.9880	0.9934
	50%	1.0036	0.9907	0.9884	0.9884	0.9980	0.9982
MAE	10%	0.7843	0.7571	0.7505	0.7521	0.7556	0.7563
	20%	0.7781	0.7485	0.7379	0.7377	0.7380	0.7407
	30%	0.7759	0.7476	0.7339	0.7348	0.7348	0.7355
	40%	0.7811	0.7599	0.7516	0.7493	0.7484	0.7497
	50%	0.7743	0.7465	0.7325	0.7338	0.7521	0.7525

▲ [표 4] Ciao 데이터 세트에서 Distance Threshold δ 와 Candidate Threshold θ 의 값에 따른 결과

아래의 표는 RMSE 관점에서 거리 threshold δ 값과 간선의 방향을 다양하게 하여 신뢰할만한 이웃 집합을 찾고, 그 이웃들 집합으로부터 얻은 결과를 나타내고 있다. 이 실험에서는 θ 값을 위 실험에서 얻은 최적의 값으로 설정한다. 아래의 표에서 $FWD_{\delta}(u)$ 는 거리 δ 안에서, 간선의 앞 방향으로 도달할 수 있는 이웃들의 집합을 의미하는 반면, $BID_{\delta}(u)$ 는 간선의 양 방향을 통해 도달할 수 있는 이웃들의 집합을 의미한다. 양 방향을 통해 얻은 이웃의 집합이 모든 데이터 세트에서 더 높은 정확도를 보였다. 특히 Epinions2 데이터 세트에서는 양 방향을 통해 얻은 이웃 집합을 사용하는 것이 다른 데이터 세트에 비해서 정확도가 눈에 띄게 향상되었다.

데이터 세트	간선 방향	Distance Threshold (δ)					
		1	2	3	4	5	6
Epinions1 ($\theta=20\%$)	BID(u)	1.1073	1.0886	1.0876	1.0877	1.0878	1.0879
	FWD(u)	1.1100	1.0968	1.0895	1.0877	1.0880	1.0880
Epinions2 ($\theta=10\%$)	BID(u)	1.1041	1.0900	1.0843	1.0851	1.0855	1.0860
	FWD(u)	1.1059	1.1025	1.0929	1.0905	1.0912	1.0915
Ciao ($\theta=20\%$)	BID(u)	1.0052	0.9864	0.9813	0.9811	0.9812	0.9816
	FWD(u)	1.0070	0.9919	0.9843	0.9822	0.9822	0.9825

▲ [표 5] 모든 사용자에게 대해 간선의 방향과 Distance Threshold δ 의 영향

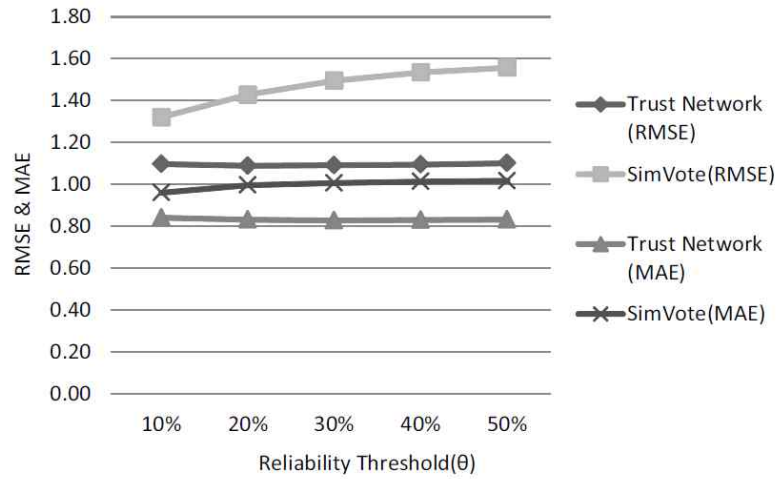
아래의 표는 cold-start 사용자들만을 대상으로 위 실험과 유사한 실험을 한 결과를 보여주고 있다. Epinions2와 Ciao 데이터세트에는 cold-start 사용자가 거의 없기 때문에, Epinions1 데이터 세트에 대해서만 실험을 수행하였다. 정확한 예측을 하기에는 정보가 부족한 cold-start 사용자를 대상으로 하였기 때문에 모든 조건에서 위 실험보다 안 좋은 결과값(높은 RMSE값)을 보이고 있다. cold-start 사용자는 아이템에 대한 평가도 적게 했을 뿐만 아니라 신뢰 관계도 다른 사용자에 비해 적게 가졌다. 그러므로, 양 방향을 통해 얻은 이웃을 사용한 경우 더 많은 수의 신뢰할만한 이웃을 얻게 되었고, 더 높은 정확도를 얻을 수 있었다.

데이터 세트	간선방향	Distance Threshold(δ)					
		1	2	3	4	5	6
Epinions1	BOTH(u)	1.195	1.1698	1.1492	1.1427	1.1415	1.1418
	FWD(u)	1.2027	1.1911	1.1737	1.1581	1.1539	1.1541

▲ [표 6] cold-start 사용자에게 대해 간선의 방향과 Distance Threshold δ 의 영향

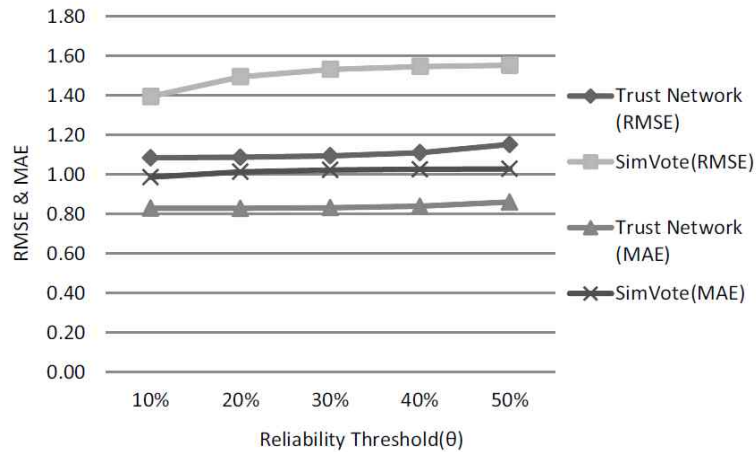
아래 그래프는 두 개의 imputation 방법을 사용했을 때의 예측 정확도를 보여주고 있다. 여기서 사용하는 두 개의 imputation은 본 프로젝트에서 제안하는 방법과 무작위로 비평가된 평점을 선택하고, 그들을 default voting approach로 채워나가는 SimVote 방법이다. 그래프에서 x축은 후보 threshold θ 값을 의미하고, y축은 MAE 또는 RMSE 값을 의미한다. 작은 그래프들은 각각의 데이터 세트에 대응되는 것들이다. 아래의 그래프들에서 볼 수 있듯이, SimVote를 사용하는 것보다 제안한 imputation 방법을 사용하는 것이 θ 값이나 측정 방법에 상관없이 더 정확한 것을 알 수 있다. 이유는 제안한 방법에서는 정확하게 추정할 수 있는 평점들에 대해서만 추정하여 채워나간 반면, SimVote는 그런 것을 고려하지 않고 무작위로 채울 평점을 선택했기 때문에, 심지어 채워낸 평점의 수가 적을 경우에도 본 실험에서 제안한 방법이 더 높은 정확도를 보이게 된다. 아래 그래프들에는 흥미로운 점이 있는데, 채워진 평점이 많을수록, 정확도가 더 안 좋아진다는 점이다. 그 이유는

imputation 방법이 정확히 추정할 수 있는 평점들에 대해서만 채워나갈 때, 행렬 인수분해를 통한 방법들이 더 정확하게 평점들을 예측해낼 수 있기 때문이다. 쉽게 말하자면, 부정확하게 채워진 평점이 많아질수록 예측의 정확도가 떨어진다.



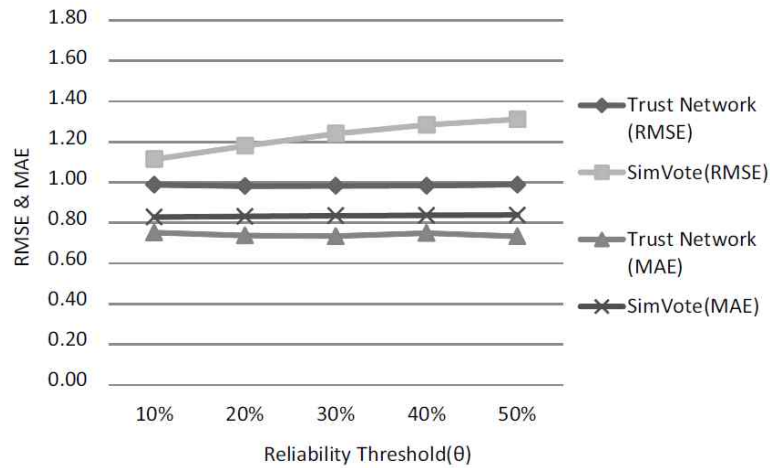
(a) Epinions1

▲ [그림 7] Epinions1 데이터 세트에서의 제안한 Imputation 방법과 SimVote의 비교



(b) Epinions2

▲ [그림 8] Epinions2 데이터 세트에서의 제안한 Imputation 방법과 SimVote의 비교



(c) Ciao

▲ [그림 9] Ciao 데이터 세트에서의 제안한 Imputation 방법과 SimVote의 비교

아래의 표는 기존 추천 방법과 제안된 imputation 방법의 정확도를 측정한 결과를 보여주고 있다. 본 프로젝트에서 제안한 방법을 적용한 PMF 모델은 다른 방법들에 비해 더 정확한 것을 확인할 수 있었고 특히 Ciao 데이터 세트에서 그 차이가 두드러졌다. 또한, 제안한 방법을 적용한 PMF모델과 SocialMF 방법이 기존의 PMF 모델보다 더 성능이 좋은 것을 확인할 수 있었는데, 이는 신뢰 네트워크를 사용하는 것이 추천 정확도를 향상시키는데 효과적이라는 것을 가리킨다. 게다가, 제안된 방법은 SocialMF보다 더 정확도가 높는데, 여기에는 두 가지 이유가 있다. 하나는 본 방법이 trustee 뿐만 아니라 trustor를 활용하기 때문이고, 다른 하나는 거리 threshold 안에서 도달할 수 있는 사용자들은 더욱 신뢰할만하기 때문에 더 좋은 성능을 보인 것이다.

데이터 세트	방법	RMSE	MAE
Epinions1	사용자 기반	1.185	0.894
	AutAI	1.097	0.838
	PMF	1.118	0.880
	SocialMF	1.115	0.879
	제안된 방법	1.088	0.823
Epinions2	사용자 기반	1.136	0.837
	AutAI	1.090	0.835
	PMF	1.109	0.868
	SocialMF	1.106	0.870
	제안된 방법	1.084	0.828
Ciao	사용자 기반	1.179	0.849
	AutAI	1.006	0.736
	PMF	1.033	0.810
	SocialMF	1.030	0.808
	제안된 방법	0.981	0.734

▲ [표 7] 전체 사용자에게 대한 추천 방법들의 정확도

cold-start 사용자의 경우, 모든 방법들이 전체 사용자에게 비교해 낮은 정확도를 보여주었는데, 이는 cold-start 사용자들의 정보가 적기 때문이다. cold-start 사용자에게 대해서 정확도를 따질 때에도, 다른 추천 방법에 비해 더 좋은 정확도를 보여주고 있다. 게다가, 전체 사용자의 경우보다 cold-start 사용자만을 따져 정확도를 비교할 때 더 큰 차이를 보여주고 있다. AutAI의 경우, 위 표와는 다르게 PMF, SocialMF 보다 더 큰 RMSE 값을 보이게 되는데, cold-start 사용자들의 정보가 부족한 탓에 AutAI 방법에서의 imputation이 신뢰할만한 평점 값을 채워나가지 못하기 때문이다. 하지만, 제안된 방법 신뢰 네트워크라는 추가적인 정보들을 활용하여 비평가된 평점을 채워나가기 때문에 더 정확한 평점을 예측할 수 있다.

데이터 세트	방법	RMSE	MAE
Epinions1	사용자 기반	1.350	0.993
	AutAI	1.288	0.934
	PMF	1.210	0.962
	SocialMF	1.209	0.968
	제안된 방법	1.142	0.888

▲ [표 8] cold-start 사용자에게 대한 추천 방법들의 정확도

여러 가지 추가 실험을 진행하면서, 신뢰 이웃을 찾기 위해서 가장 최적의 인자 값 (distance threshold, candidate threshold) 을 가장 먼저 찾고, 제안된 imputation 방법이 평점 추측의 정확도를 향상시키는데 큰 도움이 됨을 보였다. 게다가, 신뢰 네트워크를 사용하는 본 imputation 방법을 적용시킨 추천 방법이 신뢰 네트워크를 고려하지 않는 단순한 imputation 방법을 사용하는 추천 방법에 비해서 더 좋은 정확성을 보인다는 것도 알 수 있었다. 마지막으로, 특히 cold-start 사용자에게 대해서 기존 추천 방법보다 제안된 방법이 RMSE, MAE 관점에서 더 좋은 성능을 보이는 것도 보였다.

4.4 실험 결과에 대한 결론

한 사용자의 신뢰할만한 이웃들은 신뢰 네트워크에서 거리 threshold 값 안에서 간선의 양 방향으로 도달할 수 있는 사용자들을 의미하는데, 신뢰할만한 이웃들을 정의하는 접근 방법이 기존의 다른 방법들과 달랐다. (기존 방법들은 단 방향으로 접근하여 도달할 수 있는 사용자들을 신뢰할만한 이웃들로 설정하였다.) 게다가, 본 imputation 방법은 충분한 수의 이웃에게 평가된 아이템의 평점만을 고려하였는데, 적은 수의 사용자들의 의견은 신뢰도가 떨어졌기 때문이다. 또, 실험은 3가지 현실 세계의 데이터 세트에서 수행되었는데, 실험 결과는 제안된 접근 방법이 정확도를 향상시키는데 효과적임을 보여주었다. 신뢰 네트워크라는 추가적인 정보를 활용하기 때문에 특히 cold-start 사용자에게 대해서 실험을 진행할

때, 정확도의 향상이 더 두드러지게 나타났다.

5. 프로젝트 효과 및 활용방안

우선, 본 프로젝트에서 제안한 방법을 이용하면 평점 행렬의 평점 데이터들을 채워서 더 높은 밀도의 평점 행렬을 얻을 수 있다. 여전히 평점 행렬의 형태는 변하지 않았기 때문에, 평점 행렬을 사용하는 대부분의 추천 시스템에서 사용할 수 있다. 본 프로젝트에서는 PMF 모델과 결합하여 더 좋은 성능을 보인다는 것을 보였으나, PMF 모델 뿐만 아니라 평점 행렬을 활용하는 추천 방법은 무수히 많으므로, 다른 추천 방법들에도 적용시켜서 기존 추천 시스템의 정확도 측면에서의 성능을 높이는 데에 큰 기여를 할 수 있을 것으로 보인다.

또한, SocialMF 방법의 성능도 기존 PMF에 비해서 낫다는 것을 확인하게 됨으로써 신뢰 네트워크와 같은 소셜 네트워크를 활용하는 것이 cold-start 사용자 문제 등을 해결하고 기존의 추천 시스템이나 새로운 추천 시스템들의 정확도 성능을 높이기 위한 추가 정보로서 활용될 수 있음을 알게 되었다. 본 실험이 추천 시스템의 문제점을 해결하거나 성능을 높이는 데에 있어 소셜 네트워크의 중요성을 다시 한 번 제고해줌에 따라 다른 프로젝트나 논문이 이 점을 참고하여 신뢰 네트워크를 포함한 소셜 네트워크를 활용하여 새로운 실험 혹은 프로젝트를 진행할 기회를 제공해 주었다.

이러한 효과들은 결국 추천 시스템 분야 자체의 발전을 야기한다. 실제 사회에는 처리해야 할 데이터가 기하급수적으로 많아지고 있고, 사용자의 취향을 정확히 분석하여 개인화된 콘텐츠, 서비스, 추천을 제공해야 할 필요성이 더욱 많아지고 있다. 이러한 맥락에서 추천 시스템은 실제 생활과 떨어질 수 없는 중요한 기술 중 하나로 자리 잡을 것이며, 많은 데이터를 다루는 거의 모든 분야에서 적용될 수 있는 기술이므로 본 프로젝트는 기존 추천 시스템의 성능 향상에도 도움이 되고, 새롭게 등장할 추천 시스템 연구들에게 목표의식을 심어줄 수 있을 것이다.

추천 시스템 분야의 전반적인 발전을 도모할 뿐만 아니라, 본 프로젝트에서 제안한 방법은 실험 결과가 입증된 만큼, 현실 세계의 추천 시스템에 직접 적용될 수도 있을 것이다. 많은 IT 서비스들이 자동 추천 시스템을 사용하여 매출 효과를 극대화하고 있기 때문에 (예를 들어, 아마존의 경우 매출의 35%가 추천 시스템에 의해 발생하고 있다.) 기존의 추천 시스템을 추가로 사용하고 있는 서비스들이나 추천 시스템이 사업의 주요 기능이 되는 서비스들에게 있어서 더 정확한 추천은 더 큰 매출 효과를 불러일으킬 수 있을 것이다.

또한, 추천 시스템을 활용하는 새로운 사업 서비스를 생각해볼 수도 있을 것이다. 본 프

로젝트에서 제안하는 imputation 방법은 신뢰 네트워크와 같은 소셜 네트워크를 사용하므로, 사용자의 소셜 네트워크 계정을 분석해서 신뢰 네트워크 형태로 추가적인 정보를 구축한 후에, 그 정보를 기반으로 더 정확한 추천 시스템을 구현할 수 있을 것이다. 그리고 그 추천 시스템을 기반으로 하는 서비스를 생각해볼 수 있을 것이다. 기존의 추천 시스템보다 더 정확성을 높이고, 정보가 부족한 cold-start 사용자들에 대해서도 비교적 정확한 평점 예측과 아이템 추천을 해줄 수 있기 때문에 추천 시스템을 활용하는 다양한 서비스들 사이에서도 좋은 강점이 될 수 있다. 예를 들어, 영화 추천 서비스 중 하나인 '왓챠(Watcha)'의 경우, 새로 시작하는 사용자에게 30개의 영화 평점을 매기도록 요구하고 있는데, 소셜 네트워크와 같은 추가적인 정보를 활용하면 새로운 사용자에게 많은 정보를 요구할 필요 없이 정확도 높은 추천을 해줄 수 있다. 새로운 사용자에게 정보를 요구하지 않는 점은 사용자가 서비스로 들어올 수 있는 진입장벽을 많이 낮추도록 만들기 때문에, 더 매력적인 추천 서비스로 거듭날 수 있을 것이다.