사용자 기반의 협력필터링 시스템을 위한 유사도 측정의 최적화

이수정†

요 약

협력 필터링 기반의 추천시스템에서 유사도 측정은 시스템의 성능에 큰 영향을 미치는데, 이는 유사한 다른 사용자들로부터 항목을 추천받기 때문이다. 본 연구에서는 전통적인 유사도 측정 방법의 가장 큰 문제인 데이터 희소성을 극복하기 위해, 기존의 유사도 측정값과 공통평가항목수의 반영값을 최적으로 결합하는 새로운 유사도 측정방식을 제안한다. 제안 방식의 성능 평가를 위해 다양한 조건으로 실험한 결과 기존 방식들보다 우수한 예측 정확도를 나타냈으며, 구체적으로 전통적인 피어슨 상관보다 최대 약 7%, 코사인 유사도보다는 최대 약 4% 향상된 결과를 보였다.

주제어 : 추천 시스템, 협력 필터링, 유사도 척도, 사용자 기반 협력필터링

Optimization of the Similarity Measure for User-based Collaborative Filtering Systems

Soojung Lee†

ABSTRACT

Measuring similarity in collaborative filtering-based recommender systems greatly affects system performance. This is because items are recommended from other similar users. In order to overcome the biggest problem of traditional similarity measures, i.e., data sparsity problem, this study suggests a new similarity measure that is the optimal combination of previous similarity and the value reflecting the number of co-rated items. We conducted experiments with various conditions to evaluate performance of the proposed measure. As a result, the proposed measure yielded much better performance than previous ones in terms of prediction qualities, specifically the maximum of about 7% improvement over the traditional Pearson correlation and about 4% over the cosine similarity.

Keywords : Recommender System, Collaborative Filtering, Similarity Measure, User-based Collaborative Filtering

† 정 회 원: 경인교육대학교 교수
논문접수: 2015년 10월 13일, 심사완료: 2016년 1월 19일, 게재확정: 2016년 1월 20일

1. 서 론

정보 과부하는 인터넷 사용자나 콘텐츠 제공업자들에게 매우 큰 부담이 되어왔다. 이러한 문제를 해결하는 효율적인 방법으로서 추천 시스템이 연구되어 유용하게 상용화되었다. 현재까지 개발된 추천시스템들은 크게 인구통계학적 필터링(demographic filtering), 내용기반 필터링(content-based filtering), 그리고 협력 필터링(collaborative filtering, CF)으로 구분할 수 있다[1][2]. 이들 중 가장 널리 사용되는 CF 시스템은 현 사용자와 가장 유사한 선호취향을 가진 다른 사용자들로부터 정보를 획득해, 현 사용자가 선호할만한 항목들을 추천하는 것이다. 이 방법은 다양한 시스템에서 구현되었는데, Tapestry, GroupLens, Video Recommender, Amazon 등이 그 예이다[2].

이와 같이 CF 시스템의 성공은 얼마나 유사한 선호의 사용자들을 파악할 수 있는가에 달려있다고 해도 과언이 아닌데, 따라서 여러 종류의 유사도 측정 방식이 개발되어왔다. 그러나, 만약 현 사용자가 시스템의 신규 사용자이거나 시스템을 거의 사용하지 않는 경우에는 유사한 선호를 가진 다른 사용자들을 구하기 어려운 문제가 발생한다. 왜냐하면 유사성은 공통으로 평가한 항목들이 있을 때 산출할 수 있는데, 이러한 경우엔 공통 평가항목이 없거나 매우 적을 수밖에 없기 때문이다. 이러한 데이터 희소성 문제(data sparsity problem)는 CF 시스템의 인기에도 불구하고 매우 심각한 문제로 대두되어 왔다.

데이터 희소성 문제를 고려한 CF 시스템을 개발하고자 김지혜와 박두순은 연관규칙의 적용에 있어서 상관관계가 높은 항목들을 추천하는 방법을 제안하였다[3]. 그러나 희소성 문제를 해결하는 대표적인 방법으로서 공통평가항목수를 반영하여 유사도를 측정한 연구결과들이 존재한다[4][5][6]. 본 연구에서는 기존의 유사도 측정값과 공통평가항목수를 결합하는데 있어서 최적의 가중치를 구하고자 한다. 이를 위해 유전 알고리즘(genetic algorithm, GA)을 활용한다. GA는 추천시스템에서 클러스터링이나 하이브리드 사용자 모델을 구하기 위해 사용되어 왔으며, Bobadilla 등은 최적의 유사도 측정 공식을 얻기 위해 사용하였지만 그들의 연구에서는 공통평가항목수는 고려하지 않아 제한점이 있다[7].

논문의 구성은 다음과 같다. 2절에서는 관련 지식을 기술하고, 3절에서 본 연구에서 제안하는 방법을 설명하며 4절에서 실험을 통한 성능을 입증하고 5절에서 논문의 결론을 맺는다.

2. 연구 배경

2.1 유사도 측정 관련 연구

유사도 계산은 CF 시스템의 성능을 좌우하는 매우 중요한 요소이다. 사용자 기반의 시스템에서는 두 사용자가 공통으로 평가한 항목들이 존재할 때 유사도 산출이 가능하다. 기존의 유사도의 주요 산출 방법은 크게 상관도 기반과 벡터 기반의 두 종류로 나뉜다[2].

상관도 기반 유사도 산출을 위한 대표적 방법인 피어슨 상관도는 두 사용자가 선형적으로 관계된 정도를 측정한다[8]. 피어슨 상관도를 활용하는 CF 시스템은 대표적인 시스템으로서 CF 연구에서 널리 활용되고 있다. 상관도 기반의 여러 다른 변형된 유사도 측정방식도 개발되었는데, 제한 피어슨 상관도(Constrained Pearson Correlation)는 평균 대신 중간값을 사용하는 특징이 있으며, 스피어만 순위 상관도(Spearman Rank Correlation)는 평가등급 대신에 순위를 활용하는 차이점이 있다. 또한 켄달의 타우(Kendall's τ)상관도는 스피어만 상관도와 유사하나 상대적 순위를 기준으로 하여 산출한다[9].

벡터 기반의 유사도 산출 방법에서는 각 사용자의 평가등급들을 벡터로 간주한다. 가장 흔히 사용하는 코사인 유사도는 두 사용자의 평가등급 벡터 간의 코사인 값으로 정의된다. 사용자의 서로 다른 평가등급 스케일을 반영하기 위해 보정 코사인 유사도(Adjusted Cosine Similarity)의 방식도 개발되었는데, 이는 각 평가등급에서 평균등급을 차감한 후 코사인 값을 구하는 방식이다[2].

위 방법들은 모두 두 사용자 간에 공통평가항목들에 대한 평가등급을 기본으로 하기 때문에, 공통평가항목수가 적으면 산출된 유사도값에 대

한 신뢰가 저하될 수밖에 없다. 이러한 문제를 해결하기 위한 다수의 연구결과가 존재하는데 이들은 대개 공통평가항목개수를 반영하여 기존의 유사도 측정방식을 개선하려 하였다. 구체적으로, 가중화된 피어슨 상관도(Weighted Pearson Correlation)은 다른 사용자에 대한 신뢰를 고려하는데, 이 신뢰는 공통평가항목수가 많을수록 증가하게 하였다[6]. Jamali와 Ester는 사용자들 간에 공통평가항목수가 적을수록 낮은 유사도가 산출되도록 하는 sigmoid 함수 기반의 유사도 측정방식을 제안하였다[5]. 한편 Bobadilla 외 3인은 기존의 평균 제곱 차이(Mean Squared Difference, MSD) 유사도 척도에 자카드 지수(Jaccard Index)를 곱하는 방법을 통해 새로운 사용자들을 위한 추천 결과를 향상시켰다[4]. 자카드 지수는 두 사용자간의 공통평가항목수의 비율로서 정의되는 척도이다[10]. 그러나, 이 방법은 각 척도의 성능 향상에 대한 기여도가 정확하게 얼마인지 모르는 상태에서 두 척도의 비중을 동일하게 하는 단순 곱을 취하였다는 문제점이 있다.

2.2 유전 알고리즘

유전 알고리즘(Genetic Algorithm)은 자연계의 진화과정을 모방하여 최적화된 해법을 찾기 위한 탐색 방법이다[11]. 우선 다수의 초기 해(solution)들을 개발자가 선정하고 이들을 점차적으로 변형하여 새로운 세대로 진화시켜 나가면서 최적의 해를 구한다. 대개 이들 해집합으로 구성된 인구(population)는 다음 세대로 진화하면서 일정하게 그 수가 유지된다. 해의 변형은 주로 선택(selection), 변이(mutation), 교배(crossover) 연산 등을 통하여 이루어진다. 인구를 구성하는 특정 해가 최적인지를 판단하기 위하여 적합도 함수(fitness function)를 선정하여 적용한다.

해를 변형시키기 위한 연산의 정의 및 과정은 다음과 같다. 각각의 해를 특정한 유전자 형식으로 표현하고, 이들 중 적합도가 우수한 해들을 선택한 후, 이들과 이들로부터 다양한 연산을 통해 생성한 새로운 해들을 세대 물림한다. 우수한 적합도의 해를 물려준다면, 보다 나은 유전자가 다음 세대로 넘겨지게 될 확률이 커지게 되고, 결론적으로 세대가 지날수록 최적에 가까운 해가 구해질 확률이 커지게 되는 원리이다.

구체적으로, 교배 연산은 두 해의 서로 다른 위치의 일부분을 합하여 새로운 해를 만들어내는 기법이다. 대개 1-point 또는 2-point 교배를 실시하는데, 1-point 교배를 예를 들어 설명하면 다음과 같다. 해를 일정한 규칙에 의한 bit들로 표현한 후, 특정 위치의 bit를 랜덤 선정한다. 두 개의 해를 각각 H1, H2라고 하고 이들의 bit 표현을 각각 b11b12...b1m, b21b22...b2m이라고 하자. 랜덤 선정된 bit 위치를 n이라고 하면, 1-point 교배 연산은 두 개의 새로운 자손해(offspring)를 만들어내는데, 첫째 자손의 bit 표현은 b11...b1nb2(n+1)...b2m이고 두 번째 자손의 bit 표현은 b21...b2nb1(n+1)...b1m이 된다. 2-point 교배 연산도 같은 원리로 자손들을 생성한다.

가장 흔히 사용되는 또다른 연산방식인 변이 연산은 특정한 위치의 bit를 랜덤 선정한 후 그 값을 바꾸어 주는 방식이다. 즉, 현재 값이 0이면 1로, 1이면 0으로 바꾼다. 따라서, 교배 연산과는 달리 하나의 부모로부터 새로운 자손을 생성하는 방식이며, 대개 교배 연산 후에 적용한다.

유전 알고리즘을 협력 필터링 방식에 활용한 연구 결과 중에서 대표적인 것은 Bobadilla 외 3인이 제안한 방법으로서, 두 사용자가 공통평가항목들에 대해 부여한 평가등급 차이값 각각에 대한 가중치의 최적값을 유전 알고리즘으로 구하였다[7]. 그러나 기존의 다수의 연구결과[4][5][6]에서 공통평가항목수를 참조하여 성능 개선의 효과를 가져온 것과 달리, [7]의 연구에서는 이를 고려하지 않았으므로 성능 개선의 여지가 있다고 판단된다.

3. 개선 방법의 제안

기존의 전통적인 유사도 측정 방법에서는 두 사용자의 공통평가항목수를 고려하지 않았다. 보다 정확한 유사도값 산출을 위해 최근 몇몇 연구에서 이를 반영한 새로운 유사도 산출 방법을 제시하였다. 공통평가항목수를 반영하는 방법으로써 자카드 지수[10]가 개발되어 여러 연구결과에서 이를 반영하였다[4].

본 연구에서는 전통적 방식의 유사도값과 자카드 지수를 동일한 비중의 단순 곱으로 계산하여 새로운 유사도값을 산출한 Bobadilla 외 3인[4]의 방법을 개선하여 최적의 조합을 구하고자 한다. 구체적으로, 두 사용자 u와 v 간의 자카드 지수값을 J(u, v)라고 하고 사용자 u가 평가한 총 평가항목들의 집합을 I(u)라고 하자. 또, 두 사용자의 공통평가항목들의 집합을 I(u, v)라고 할 때,

$$\mathcal{J}(u,v)=\frac{|\mathcal{I}(u,v)|}{|\mathcal{I}(u)|+|\mathcal{I}(v)|-|\mathcal{I}(u,v)|}$$

로 계산된다[10]. 두 사용자 u와 v 간의 전통적 방식에 따른 유사도값을 sim_T(u, v)이라고 할 때, 본 연구에서 제안하는 u와 v 간의 유사도값은

$$sim_N(u,v)=sim_T(u,v)^{\alpha} \bullet \mathcal{J}(u,v)^{\beta}$$

로 정의한다. 따라서, 위 식에서 각 구성요소의 최적의 가중치, α와 β를 구하는 것이 목표이다.

2.2 절에서 기술하였듯이, 다양한 연구 영역에서 유전 알고리즘의 성질을 이용하여 왔으며, 특히 주요 특성들을 선정하여 가중치를 부여하는 문제를 포함한 넓은 스펙트럼의 최적화 문제의 해결에 유전 알고리즘의 유용함을 나타냈으므로 [12], 본 연구에서도 최적의 가중치 산출을 위해 유전 알고리즘을 활용하도록 한다.

적합도 함수로는 평균절대오차(Mean Absolute Error, MAE)을 사용하였다. 이 척도는 협력필터링 시스템 평가를 위해 매우 널리 사용되어왔으며, 각 테스트 항목 i의 평가등급의 실제치 ri와 그 예측치 r'i의 차이값의 평균으로 정의된다[10]. 따라서 이 값을 작게 할수록 좋은 방법으로 평가된다. 예측치는 평가대상자 u와 유사도가 높은 사용자들(인접이웃들, nearest neighbors)의 항목 i에 대한 평가치를 조합하여 산출하며 대개 다음과 같은 식을 이용한다[8].

$$r'_{u,i}=\overline{r_u}+\frac{\sum_{v\in NN_u} sim(u,v)\times(r_{v,i}-\overline{r_v})}{\sum_{v\in NN_u}|sim(u,v)|}$$

이 때, $\overline{r_u}$는 u의 평균 평가치, sim(u, v)는 사용자 u와 v 간의 유사도값, $r_{v,i}$는 사용자 v의 항목 i에 대한 평가치, NN_u는 u의 인접 이웃 집합이다.

유전 알고리즘의 구체적 절차는 <표 1>에 제시하였다. 1단계에서는 전체 인구를 구성하는 해인 α와 β 쌍을 랜덤하게 발생시키는 과정이다. 각 가중치가 어떤 값일 때 성능이 좋을지 알 수 없

<표 1> GA를 이용한 가중치 산출 절차

- P: population of solutions of a pair α and β.
- W_{range} : real number range of a solution
- $N_{generations}$: number of generations
- $F_{threshold}$: fitness threshold
- $fitness_i$, $fitness_{best}$, $fitness_{max}$: fitness of solution i, best and maximum fitness among all solutions', respectively.
- P_S: a set of selected solutions per generation
- N_{BITS}: number of bits composing a weight of a solution
- $Prob_M$: mutation probability

1. Initialize solutions in P: Generate a real random number within W_{range} for each solution.
2. Compute $fitness_i$ for each solution i in P.
3. while number of generations< $N_{generations}$ and $fitness_{best}$ >$F_{threshold}$ do
   3.1 Select solutions probabilistically with the probability for solution i given by
   $$Prob_{S,i}=0.5 \bullet (1.0-\frac{fitness_i-fitness_{best}}{fitness_{max}-fitness_{best}})$$
   3.2 Randomly select two solutions, s1 and s2, in P_S. Choose two random numbers among (0, N_{BITS}). Produce two offsprings from s1 and s2 by applying the two-point crossover operator with the chosen random numbers.
   3.3 If the number of offsprings produced in Step 3.2 is less than P-P_S, then go to Step 3.2.
   3.4 Select solutions in P with $Prob_M$ and change the value of a random bit of α and that of β.
   3.5 Compute $fitness_i$ for each solution i in P.
4. Return the solution with the best fitness.

으므로 특정 휴리스틱을 적용하지 않은 채 미리 정해놓은 전체 범위(W_{range}) 내에서 임의의 숫자를 발생시켰다. 2단계는 각 해를 적용한 시스템의 성능 결과를 평균절대오차 측면에서 산출하는 과정이다. 이들 중 가장 좋은 적합도($fitness_{best}$)가 미리 정한 하한선($F_{threshold}$)에 도달하지 못하였고, 아직 진화된 세대수가 정해진 세대수($N_{generations}$)에 못미쳤을 경우 3단계의 반복절차를 수행한다. 3단계는 유전 알고리즘의 세가지 주요 연산을 실행하는 과정이다. 우선 좋은 적합도의 해가 선택될 확률이 높도록 해들을 선택하고(3.1단계), 이들 중 임의로 두 개의 해를 선택하여 교배연산을 수행한다(3.2단계). 교배연산의 실행 회수는 3.1단계에서 선택된 해들의 수와 합하여 전체 인구가 P로 고정되도록 한다(3.3단계). 마지막 연산으로 3.4단계에서 변이연산을 주어진 확률(ProbM)에 의거하여 실행하는데 각 가중치에 대해서 임의의 한 비트를 변환한다. 이와 같이 모든 연산을 실행한 후 다시 적합도를 산출하고 조건이 부합되는 한(3단계) 반복 실행한다. 최종적으로 가장 좋은 적합도를 산출한 해를 출력하면 원하는 가중치를 얻게 되므로, 목적을 달성하는 것이다.

4. 성능 평가

4.1 실험 배경

앞절에서 설명한 유전 알고리즘을 구현하기 위한 각종 변수값의 설정은 다음과 같이 하였다. 각 해의 비트 표현을 위해 α와 β 모두 각기 10 비트씩을 할당하였다. 인구를 구성하는 모든 해들의 초기값은 랜덤하게 할당하였으며, 값의 범위는 [0, 2]에 속하는 실수값이 되도록 하였다. 인구의 크기에 대해서는 해를 구성하는 비트수의 2배로 하기를 권장한 연구결과[13]에 따르면 40이 되지만, 본 연구에서는 해를 구성하는 가중치 수가 2에 불과하므로 보다 다양한 형태로 진화될 수 있도록 인구 크기를 60으로 늘려 실험하였다. <표 2>에 본 실험에 적용할 파라미터값들을 제시하였다.

<표 2> 유전 알고리즘 구현을 위한 파라미터값

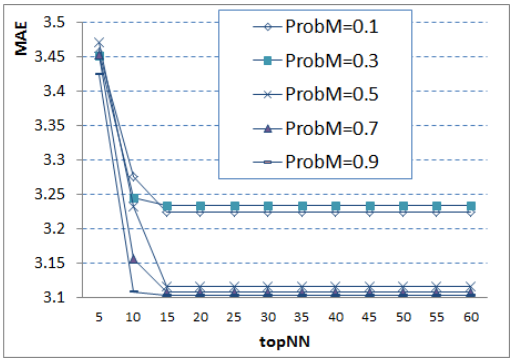
P	W_{range}	$N_{generations}$	$F_{threshold}$	N_{BITS}	$Prob_M$
60	[0, 2]	20	3.0	10	[0.1, 0.9]

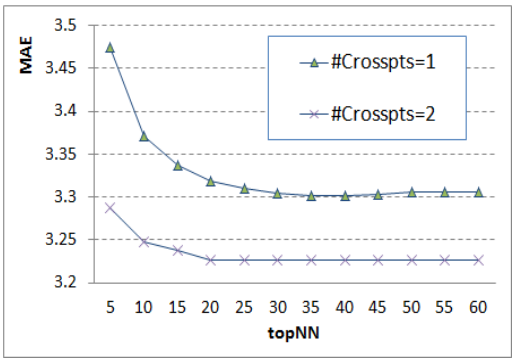
제안 방법의 성능을 평가하기 위하여, CF 관련 연구에서 널리 활용되어왔던 데이터베이스인 MovieLens, Jester, Netflix 등을 고려할 수 있으나, Netflix는 더 이상 자료 공개를 하지 않아 접근이 불가하고, 나머지 두 개의 데이터베이스들 중에서 평가개수가 매우 많아 희소성 수준 측면에서 월등한 Jester 데이터셋[1]을 사용하였다. 이 데이터셋은 Eigentaste 알고리즘[14]과 그 밖의 여러 CF 연구의 실험에 활용되었다[2][15]. Jester 셋은 원래 24,983명의 사용자들이 100개의 농담에 대해 평가한 것이나, 본 연구에서 사용한 PC 환경과 유전 알고리즘의 계산복잡도를 고려하여 998명의 사용자들로 축소하여 실험하였다. <표 3>에 실험한 데이터셋의 특성에 대해 상세 기술하였다. 희소성 수준(sparsity level)이란 행렬 내 데이터가 없는 요소, 즉, 평가가 매겨지지 않은 요소의 비율을 의미하며, (값이 0인 요소 개수)/(행렬의 크기)로 산출한다.

<표 3> 실험 데이터 집합

평가개수	각 사용자 당 36개 초과
행렬크기 (사용자수×항목수)	998 × 100
평가범위	-10～+10의 실수
희소성수준	0.4297

앞절에서 설명하였듯이 기존에 널리 활용되었던 유사도 계산 방식인 피어슨 상관도(COR), 코사인 유사도(COS), 평균 제곱 차이(MSD)를 기준으로 하고, 이들 각각에 자카드 지수를 곱한 방식, 즉, α=β=1.0으로 한 경우를 COR*J, COS*J, MSD*J로 표기하였다. 또한 본 연구의 유전 알고리즘을 적용한 유사도 산출 방법은 COR-GA, COS-GA, MSD-GA로 각각 표기하였다. 성능 평가 척도로는 예측 정확도를 나타내는 평균절대오차(MAE)[10]를 사용하였으며, 사용자 u에 대해 다음과 같이 정의된다.





<그림 1> 변이 확률(ProbM)과 교배연산점(Crosspts) 개수에 따른 MAE 성능 변화

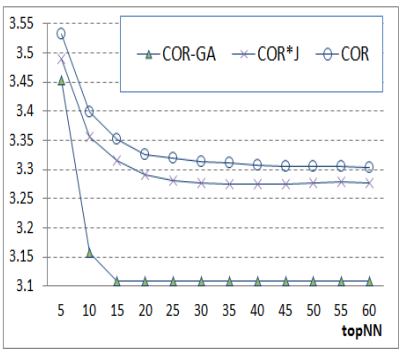
$$MAE_u = \frac{\sum_x |r_{u,x} - r'_{u,x}|}{N}$$

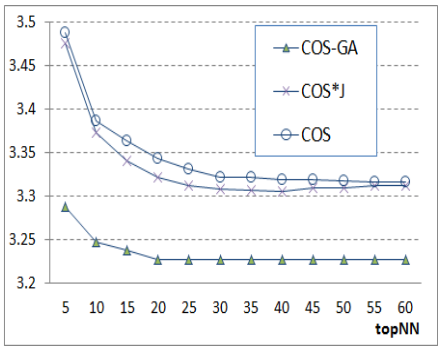
$r_{u,x}$: 항목 x에 대해 사용자 u가 부여한 등급
$r'_{u,x}$: 항목 x에 대한 사용자 u의 등급 예측값
N : 성능 평가 대상 항목의 총개수

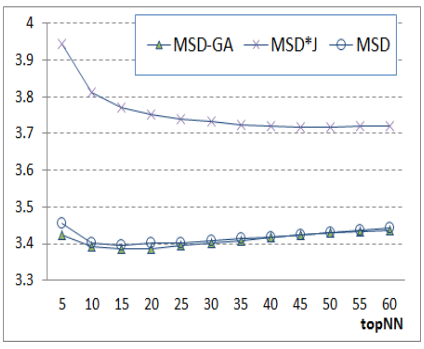
4.2 성능 결과

<그림 1>은 변이확률(ProbM)과 교배연산점의 개수를 변화시켰을 때 인접이웃수의 변화에 따른 평균절대오차 성능에 대한 영향을 살펴본 결과이다. COR에 대해 실험한 결과, 대체로 ProbM이 작을 때보다 큰 경우에 성능이 더 좋았다. 또한 ProbM이 0.5 이상인 경우의 성능 결과가 거의 유사하고, 그보다 작은 경우의 성능도 서로 유사한 것을 알 수 있다. 변이확률이 클수록 해들의 변화가 매우 다양해져서 좋은 적합도의 해가 생성될 가능성이 커지므로 이와 같은 결과를 가져왔다고 판단된다. 우측 그림은 COS에 대해 실험한 결과로서 교배연산점 개수가 2일 때 더욱 성능이 좋았다. 그 이유 또한 교배연산점이 2인 경우, 진화의 폭이 더욱 다양해지므로 좋은 적합도의 해를 산출한 것으로 보인다. COR나 MSD의 경우엔 교배연산점의 영향을 거의 받지 않았다. 결론적으로, 본 실험에서는 각 유사도 척도별로 최적의 성능결과를 보인 파라미터값을 설정하여 실험을 진행하였다.

<그림 2>는 각 유사도 척도별로 세가지 변종에 대해 인접이웃수(topNN)의 변화에 따른 평균절대오차를 측정한 결과이다. COR와 COS의 경우에 그 양상이 유사한데, 원래의 방식이 가장 나쁜 결과를 보이고, COR*J와 COS*J가 그보다 매우 약간의 차이로 좋은 성능을 보였으며, GA 방식은 월등히 우수한 결과를 나타냈다. 이로서 유전 알고리즘을 통하여 기존 유사도 측정값과 자카드 지수값의 최적 조합을 구하여 적용한 방식이 성능에 큰 영향을 미침을 알 수 있다. 구체적으로 인접이웃수가 10 이상일 때, GA 방식은 COR보다 약 6~7.3%, COR*J보다 약 5~6.3% 정도 향상된 결과를 가져왔고, COS보다 약 2.7~4.2%, COS*J보다 약 2.4~3.7%의 개선효과가 있었다.







<그림 2> 각 유사도 측정방식에 의한 MAE 성능 결과 비교

MSD의 경우, 다소 다른 양상을 보이는데, MSD*J가 나머지 두 방식보다 월등히 낮은 성능을 보였고, MSD와 MSD-GA는 거의 같은 결과를 나타냈다. 이는 MSD 산출값이 신뢰도가 높아 공통평가항목수를 반영하지 않아도 무관하며, 오히려 자카드 지수를 반영했을 때 악영향을 미치는 것으로 확인된 것이다. 실제로 MSD-GA 실험 결과 α=1.9, β=0.051일 때 최적의 성능이 추출되었으며, 따라서 자카드 지수의 반영은 성능 향상에 바람직하지 않음을 알 수 있다. 또한, MSD의 경우 변이확률의 변화나 교배연산점 개수의 변화가 성능 변화에 거의 영향을 주지 않았으므로 MSD-GA와 MSD의 성능이 유사한 결과를 낳은 것으로 보인다. 구체적으로 향상 정도를 살펴보면, MSD*J보다 다른 두 방법의 성능이 대략 7.4 ~11% 정도 우수하였다.

5. 결 론

본 연구에서는 협력 필터링 기반의 추천시스템을 위한 새로운 유사도 측정방식을 제안하였다. 기존의 유사도 측정값과 공통평가항목수를 유전알고리즘을 통해서 최적 결합할 수 있는 방법을 제안하였고 이를 다양한 조건하에서의 실험을 통해 성능 평가하였다. 실험 결과 예측 정확도 면에서 제안 방법이 월등히 좋은 성능을 나타냈다.

다양한 마이닝 기법을 활용한 추천 시스템 개발은 관련 연구 분야에서 주요 관심사가 되어 왔다. 예를 들어, [3]에서는 연관 규칙을 활용하여 상품을 추천하였고, [12]는 클러스터링 기법을 이용하여 시스템의 확장성에 따른 부하를 감소하였다. 또한 보다 정확한 유사도값을 산출하기 위한 노력이 꾸준히 진행되어 왔다[7][16]. 그러나 본 연구에서처럼 데이터 희소성 문제를 고려하여 최적화된 유사도 척도를 개발하려는 노력은 상대적으로 드물었으므로, 그 가치가 주목될 것이라 판단된다.

실험에 사용한 데이터셋은 희소성 수준면에서 우수하며 평가등급의 범위가 큰 경우이다. 따라서 다른 특성을 가진 데이터셋들에 대한 추가 실험이 바람직할 것이며, 예측 정확도뿐만 아니라 추천 정확도 등의 기타 성능 척도에 대한 평가도 이루어질 계획이다.

참 고 문 헌

[1] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge & Data Engineering, 17*(6), 734-749.

[2] Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. *Advances in Artificial Intelligence, 2009,* 4.

[3] 김지혜 · 박두순 (2006). 연관규칙과 협업적 필터링을 이용한 상품 추천 시스템 개발. **컴퓨터교육학회논문지, 9**(1), 1-10.

[4] Bobadilla, J., Ortega, F., Hernando, A., & Bernal, J. (2011). A collaborative filtering approach to mitigate the new user cold start problem. *Knowledge-Based Systems, 26*, 225-238.

[5] Jamali, M., & Ester, M. (2009, June). TrustWalker: a random walk model for combining trust-based and item-based recommendation. In *Prococeedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 397-406). ACM.

[6] Liu, H., Hu, Z., Mian, A., Tian, H., & Zhu, X. (2014). A new user similarity model to improve the accuracy of collaborative filtering. *Knowledge-Based Systems*, *56*, 156-166.

[7] Bobadilla, J., Ortega, F., Hernando, A., & Alcal, J. (2011). Improving collaborative filtering recommender system results and performance using genetic algorithms. *Knowledge-Based Systems, 24*(8), 1310-1316.

[8] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994, October). GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM Cconference*

on Computer Supported Cooperative Work (pp. 175-186). ACM.

[9] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, *22*(1), 5-53.

[10] Koutrica, G., Bercovitz, B., & Garcia-Molina, H. (2009, June). FlexRecs: expresing and combining flexible recommendations. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data* (pp. 745-758). ACM.

[11] Mitchell, T. M. (2010). *Machine Learning*. IL: McGraw Hill.

[12] Hwang, C. S., Su, Y. C., & Tseng, K. C. (2010). Using genetic algorithms for personalized recommendation. In *Computational Collective Intelligence, Technologies and Applications* (pp. 104-112). Springer Berlin Heidelberg.

[13] Alander, J. T. (1992, May). On optimal population size of genetic algorithms. In *CompEuro'92. 'Computer Systems and Software Engineering', Proceedings.* (pp. 65-70). IEEE.

[14] Goldberg, K., Roeder, T., Gupta, D., & Perkins, C. (2001). Eigentaste: a constant time collaborative filtering algorithm. *Information Retrieval, 4*(2), 133-151.

[15] Anand, D., & Bharadwaj, K. K. (2010, July). Adaptive user similarity measures for recommender systems: a genetic programming approach. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on* (Vol. 8, pp. 121-125). IEEE.

[16] 이수정 (2011). 협력 필터링 시스템을 위한 순위 기반의 유사도 척도. **컴퓨터교육학회논문지, 14**(5), 97-104.

이 수 정

1985 이화여자대학교 수학교육과(이학사)
1990 Texas A&M 대학교 컴퓨터과학과(석사)
1994 Texas A&M 대학교 컴퓨터과학과(박사)
1994~1998 삼성전자 통신개발실 선임연구원
1998~현재 경인교육대학교 컴퓨터교육과 교수
관심분야: 컴퓨터교육, 추천시스템, 정보필터링
E-Mail: sjlee@gin.ac.kr