JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 18, No. 4, pp. 707-712, July. 2017

Factorization Machine을 이용한 추천 시스템 설계

정 승 윤 · 김 형 중
고려대학교 정보보호대학원 빅데이터 응용 및 보안학과

A Recommender System Using Factorization Machine

Seung-Yoon Jeong · Hyoung Joong Kim
Division of Information Security Graduate School of Information Security, Korea University, Seoul 02841, Korea

[요 약]

데이터의 양이 기하급수적으로 증가함에 따라 추천 시스템(recommender system)은 영화, 도서, 음악 등 다양한 산업에서 관심을 받고 있고 연구 대상이 되고 있다. 추천시스템은 사용자들의 과거 선호도 및 클릭스트림(click stream)을 바탕으로 사용자에게 적절한 아이템을 제안하는 것을 목적으로 한다. 대표적인 예로 넷플릭스의 영화 추천 시스템, 아마존의 도서 추천 시스템 등이 있다. 기존의 선행 연구는 협업적 여과, 내용 기반 추천, 혼합 방식의 3가지 방식으로 크게 분류할 수 있다. 하지만 기존의 추천 시스템은 희소성(sparsity), 콜드스타트(cold start), 확장성(scalability) 문제 등의 단점들이 있다. 이러한 단점들을 개선하고 보다 정확도가 높은 추천 시스템을 개발하기 위해 실제 온라인 기업의 상품구매 데이터를 이용해 factorization machine으로 추천시스템을 설계했다.

[Abstract]

As the amount of data increases exponentially, the recommender system is attracting interest in various industries such as movies, books, and music, and is being studied. The recommendation system aims to propose an appropriate item to the user based on the user's past preference and click stream. Typical examples include Netflix's movie recommendation system and Amazon's book recommendation system. Previous studies can be categorized into three types: collaborative filtering, content-based recommendation, and hybrid recommendation. However, existing recommendation systems have disadvantages such as sparsity, cold start, and scalability problems. To improve these shortcomings and to develop a more accurate recommendation system, we have designed a recommendation system as a factorization machine using actual online product purchase data.

색인어 : 추천시스템, 협업필터링, 행렬분해, Factorization Machine, SVD

Key word : Recommendation System, Collaborative Filtering, Matrix Factorization, Factorization Machine, SVD

http://dx.doi.org/10.9728/dcs.2017.18.4.707

Received 11 June 2017; **Revised** 17 July 2017
Accepted 28 July 2017

***Corresponding Author; Hyoung Joong Kim**

Tel: +82-02-3290-4895
E-mail: khj-@korea.ac.kr

Ⅰ. 서 론

추천시스템(Recommendation System)이란 사용자가 선호할 만한 상품을 추천하는 시스템을 일컫는다. 현재 온라인에서 적용되고 있는 추천서비스 사업자는 Netflix, Amazon, MovieLens, Melon 등의 업체가 있으며, 이들 업체에서는 사용자의 과거 선호도, 구매기록, 클릭패턴 등의 정보를 사용하여 사용자에게 적절한 항목을 추천해주고 있다. 전통적인 추천기술은 크게 협업필터링(collaborative filtering), 내용기반 추천(content-based filtering), 그리고 혼합 추천(hybrid recommendation)을 들 수 있다.

협업필터링 기반 추천시스템은 사용자의 행동 정보를 분석하여 해당 사용자와 비슷한 성향의 사용자들이 기존에 좋아했던 항목을 추천하는 기술이다. 예컨대 이를 가장 직관적으로 예측하는 방법은 한 유저의 한 영화에 대한 평점을 예측하기 위해서 그 영화를 본 다른 유저의 이력을 이용한다. 단순히 전체 영화에 대한 평균으로 예측하는 것이 아니라, 그 유저와 가까운 성향을 가진 유저의 그 영화에 대한 평점 평균을 구하고 그 결과를 제공하는 것이다.

콘텐츠기반 필터링 추천시스템은 항목 자체를 분석하여 시스템을 구현한다. 사용자의 선호도를 추출하여 이에 대한 유사성을 계산한다. 콘텐츠 기반 필터링은 내용 자체를 분석하므로 새로운 아이템에 대한 사용자의 선호도를 해당 사용자의 다른 아이템들에 대한 기존의 평가에 기반하여 예측한다.

하지만 이러한 방법의 추천시스템은 사용자의 수가 증가함에 따라 평가 자료의 희소성(sparsity)이 증가하면서 성능이 감소하는 경향이 있다. 따라서 고차원의 데이터를 효과적으로 처리하는 방법이 필요하며, 선형대수의 방법 중 하나인 행렬의 특이값 분해, 즉 SVD(singular value decomposition)를 이용한 추천시스템이 등장하게 되었다. 이후 NMF(non-negative matrix factorization)[1], 텐서분해 모델로 TD(Tucker decomposition)[2]이나 TD를 개선한 PITF (Pairwise Interactive Tensor Factorization)[3] 등이 사용되었다,

한편 기계학습의 방법인 FM(factorization machines)은 기존의 알고리즘과 주목대상이 달라서 사용자와 아이템의 상호작용에 중점을 두며 이는 추천시스템 향상에 많은 기여를 하였다.

본 논문은 다음과 같이 구성된다. 2장에서는 협력필터링 방법, 콘텐츠기반 필터링 방법, SVD, FM에 관련된 선행연구를 소개하며, 3장에서는 본 논문에서 고찰하는 방법에 대한 기본적인 소개와 전체적인 방법을 설명하며, 4장에서는 실제 기업의 평점데이터에 각 알고리즘을 적용하고 성능을 평가하며, 5장에서는 결론을 제시한다.

Ⅱ. 관련연구

2-1 Collaborative Filtering

collaborative filtering은 사용자들의 과거 행동 데이터를 분석한 후, 같은 관심사를 가진 사용자들을 그룹화하고, 그 그룹에 속한 사람들이 많이 소비한 항목을 추천하는 기술이다[4].

collaborative filtering의 기본 가정은 사용자들의 과거 선호도가 미래에서도 그대로 유지될 것이라는 전제에 있다. 예를 들어, 영화에 관한 협력 필터링 혹은 추천 시스템(recommendation system)은 사용자들의 선호도에 대한 과거 데이터를 이용하여 그 사용자의 미래 선호도를 예측하게 된다. 이 시스템은 특정 사용자의 정보에만 국한된 것이 아니라 많은 사용자들로부터 수집한 정보를 사용한다는 것이 특징이다. 이것이 단순히 투표수를 기반으로 각 아이템의 관심사에 대한 평균적인 평가로 처리하는 방법과 차별화 된 것이다. 즉 고객들의 선호도와 관심 표현을 바탕으로 선호도, 관심에서 비슷한 패턴을 가진 고객들을 식별해 내는 기법이다. 사용자의 자세한 프로필이나 아이템에 대한 정보가 필요 없고, 양질의 추천 결과를 제공해 준다는 장점이 있지만, 계산량이 많고 결측 정보에 취약하다는 단점이 있다.

2-2 Content-based Filtering

사용자가 과거에 사용했거나 평가한 아이템을 기반으로 새로운 아이템을 추천하는 방식이다. 예를 들어 영화 추천 서비스의 경우 사용자가 보았던 영화들 중에 높게 평가한 것들에 대해 다양한 측면(주연배우, 감독, 장르, 주제 등)에서 공통점을 찾아낸다. 이렇게 분석된 각 사용자의 영화 선호도에 기반하여 새로운 영화가 사용자가 좋아하는 영화에 얼마나 근접하는지 평가, 높은 점수를 받는 영화를 추천한다. 이와 같이 내용 기반 방식에서는 사용자의 과거 경험에서 선호도, 취향, 욕구 등의 사용자 정보를 어떻게 찾아내는가가 핵심 이슈다[5]. 이를 위해 각 아이템의 내용을 분석하여 그것을 특징지을 수 있는 속성들을 찾아내는 과정이 필요하다. 그리고 나서 계산된 사

용자의 과거 아이템들에 대한 속성들을 통합하여 사용자의 선호도에 대한 정보를 구축한다. 이것이 새로운 아이템에 대한 평가에 대한 기준으로 사용된다.

기본적인 방법으로 구현이 간단하고 사용자의 명시적인 신호 정보를 직접적으로 반영할 수 있다는 장점이 있지만 아이템의 내용 기반 정보를 구하기 어려운 경우가 많고, 사용자의 명시적 프로필을 얻기 힘들며, 특히 사용자의 선호 취향을 특정 단어로 표현하기가 매우 힘이 든다는 단점 또한 존재한다.

2-3 Singular Value Decomposition (SVD)

협업 필터링 방법은 사용자의 수가 증가함에 따라 평가 자료의 희소성이 증가하고 성능이 감소되는 성향을 갖게 된다. 따라서 고차원 평가 자료를 효과적으로 처리하는 방법이 필요하게 되었다. SVD는 행렬의 전체적인 구조(global structure)를 기반으로 추천을 하는 방식이다[6]. 차원축소 개념의 일종이며 고차원의 행렬을 저차원의 행렬로 축소시켜 분석의 정확성을 높이고 계산 속도를 향상시킬 수 있다.

SVD는 원래의 행렬을 직각행렬 2개와 1개의 대각행렬로 분해한다. 추천 시스템에서는 각각의 분해된 행렬을 이용하여 차원축소를 실시할 수 있다. 특이값 분해가 유용한 이유는 행렬이 정방행렬이든 정방행렬이 아니든 관계없이 모든 $m \times n$ 행렬에 대해 적용 가능하기 때문이다. 행렬 분해 모델은 여러 가지 모델이 존재하지만 SVD는 명확한 평가 또는 관계 정보가 있을 경우, 이를 바탕으로 행렬을 분해하였을 때, 암묵적 요인(latent factor)를 잘 정의한다고 알려져 있다. 특히 사용자의 항목에 대한 선호도 결측 정보가 많을 때 용이하게 사용된다[7].

SVD는 임의의 특성행렬을 구성하여 관찰된 목표 행렬 간의 차이 값을 기울기 강하(gradient descent)[8] 기법을 사용하여 최소화함으로써 특성 행렬을 학습하는 방법이 많이 사용된다. SVD를 이용하여 차원을 줄일 수 있게 되면 빠른 계산이 가능하게 되고 실시간 추천을 할 수 있다.

2-4 Factorization Machines (FM)

factorization machine은 Rendle에 의해 제안된 행렬분해 기법이다[9]. 타겟 변수가 있는 지도학습 기반 기계학습이며 회귀와 분류 모두 할 수 있는 방법이다. SGD (stochastic gradient descent)방법[8], ALS (alternative least square)방법[10], 또는 마코프체인 방법[11]으로 훈련되는 시스템이다. 변수들을 저차원 공간에 매핑함으로써 변수의 상호 작용을 모델링한다. factorization machine은 사용자와 아이템` 상호작용(pair-wise) 특징벡터에 주목한 것이 기존 추천 알고리즘과 비교된다[4].

모델 방정식과 파라미터는 다음과 같다.

Model: Y≈f(x), x is highly sparse

$$\hat{y}(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} < v_i, v_j > x_i x_j \quad (1)$$

: $w_0 \in R,\ w \in R^n,\ V \in R^{n \times k}$ (2)

여기서 각 표기에 대한 설명은 다음과 같다.

$\hat{y_i}$: target variable.

w_0 : global bias.

w_i : strength of the *i*-th variable.

$\hat{w}_{i,j} := < v_i, v_j >$: interaction.

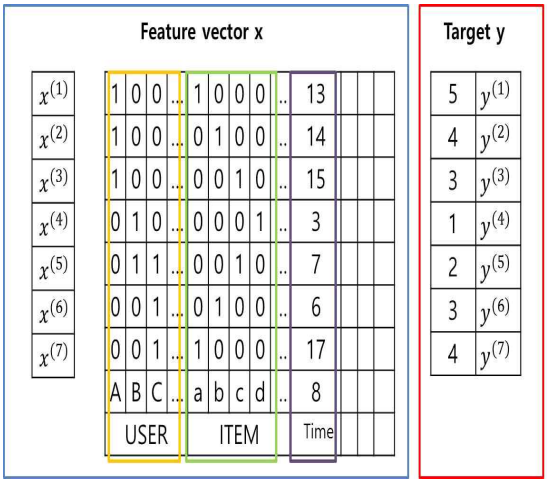

그림 1. FM 도식표

Figure. 1. Figure of Factorization Machines Parameters

Ⅲ. 연구방법 및 모델

3-1 문제정의

추천시스템은 사용자가 항목에 대해 평가한 과거 선호도를 기반으로 아직 사용하지 않는 항목에 대한 사용자의 평가를 예측하는 문제라고 할 수 있다[12].

추천 문제를 다시 정의한다면, 그림 2에서처럼 사용자와 항목이라는 두 가지 요소가 존재한다. 사용자 u가 아이템 i를 얼마나 좋아할 것인지 나타내는 값을 등급 r_{ui}라 하자. 이 때 등급 값은 평점 데이터처럼 1에서 5사이의 실

수형 등급 값일 수도 있고, 클릭 여부를 나타내는 이진 등급 값일 수도 있다. 이런 경우 우리가 가지고 있는 데이터는 r_{ui}들의 값이 될 것이고, 그림 2와 같은 행렬로 나타낼 수 있다.

이 때 R은 비어있는 곳이 없는 원래 데이터를 의미한다. $\hat{R}$는 원래 데이터로부터 비어있는 곳을 복구한 데이터를 의미한다. 여기에서, 원래 데이터 행렬 R에서 값이 없었던 부분을 제외하고 에러를 계산하고 최소화 시키는 것, 즉 RMSE값을 줄이는 것이 추천문제라고 할 수 있다.

이를 식으로 식 (3)과 식 (4)로 표현할 수 있다.

$$\min \| \hat{R} - R \|^2 \tag{3}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (\hat{r_i} - r)^2}{n}} \tag{4}$$

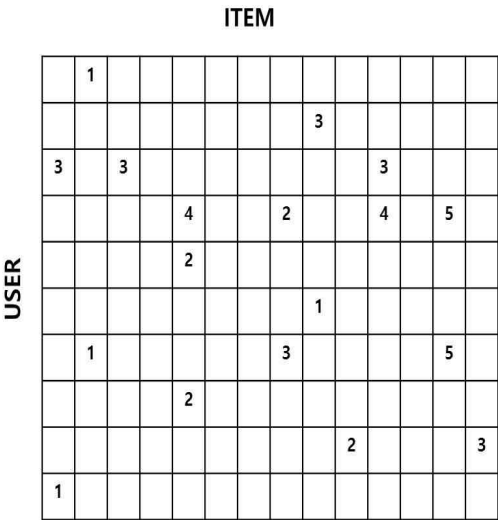

그림 2. 희소행렬
Figure. 2. Sparse Matrix

3-2 연구모형 및 연구범위

본 연구는 기업의 실제 데이터를 대상으로 추천시스템의 최적화에 대한 실증 연구를 진행하려고 한다. 추천시스템 평가를 위한 알고리즘은 사용자와 아이템의 상호작용(pair-wise interaction) 특징벡터[13]를 잘 나타낸다고 알려진 FM을 이용하기로 하였다.

고객의 상품에 대한 평점 및 시간에 대한 로그 데이터를 sparse matrix 로 변환한 후 데이터를 학습시키기 위해 트레이닝 데이터와 테스트 데이터를 8:2의 비율로 나누었고, 추천시스템 최적화를 위해 SGD(stochastic gradient descent) 방법을 통해 최적의 파라미터를 구해보고자 한다. 그림 3은 본 연구를 도식화 한 그림이다.

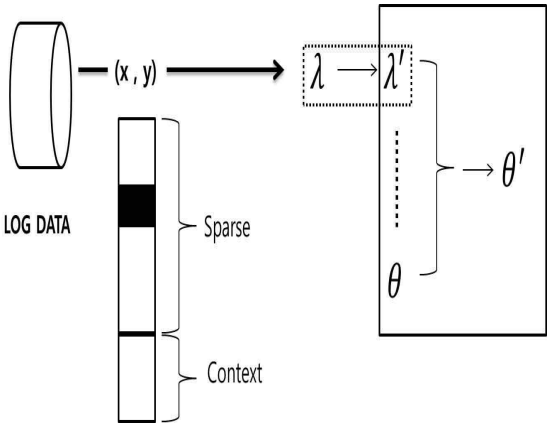

그림 3. 연구방법론
Figure. 3. Overview of research

IV. 실험 및 결과

4-1 데이터 수집 및 데이터 구조

실험에 사용한 데이터는 참좋은여행사의 약 1년간의 로그데이터 14,553건의 거래 자료이다. (이하 A회사)

표 1. A회사 데이터 구조
Table. 1. Database structure of A company

user	item	ratings	time stamp
u_1	i_1	r_1	t_1
u_2	i_2	r_2	t_1
u_3	i_3	r_3	t_3
⋮	⋮	⋮	⋮

데이터 중 item은 여행상품을 나타낸다. 변수 ranking은 1부터 5까지 사이의 정수로 표현된다. 전체적으로 영행상품 데이터의 변수는 user 데이터, item 데이터, rating 데이터, time 데이터 등 총 4가지 변수가 사용되었다. 여행을 다녀온 회원들의 여행상품에 대한 평점 데이터와 평점을 기록한 시간에 대한 로그 데이터를 사용해 추천 시스템을 만들었다.

실험은 통계패키지 R을 이용하였고, R로 작성된 FM

패키지인 FactorizationMachines를 이용해 실험이 수행되었다. 패키지의 최적화를 위해 하이퍼파라미터 iteration 값과 regularization 값 각각을 조정하며 실험을 수행하였다. 사용한 변수는 user, item, rating, time 등으로 네 가지라서 factorizing parameter는 4라고 표현했다.

4-2 실험결과 및 분석

1) 반복회수에 따른 실험결과 (factorizing parameter=4)

식 (1)의 계수를 반복적인 최적화 기법을 적용하므로 반복회수를 나태내는 iter의 수에 따라 성능이 달라진다. 실험에서 반복회수 값은 10부터 1,000까지 10단위로 변화시키며 변화를 살펴보았다. 실험의 반복회수에 따른 실험결과의 경우 iter = 126 까지는 RMSE 가 점차 감소하지만 그 이상 반복되면 RMSE가 증가하는 over-fitting 문제가 발생하였다.

2) regular 값에 따른 실험결과 (factorizing parameter=4)

FM은 기본적으로 선형회귀에 기반을 둔 모델로 식 (1)의 계수를 구하는 것이 목적이다. 그런데 계수에 제한을 두지 않을 경우 과적합이 생길 수 있다. 이에 따라 선형회귀에서도 정규화(regularization)[14]를 위해 L_1 또는 L_2 노옴을 적용하고 있다. FM은 변수 사이의 상호작용(interaction) 항을 포함해 궁극적으로는 선형회귀와 유사하다. 그래서 정규화를 적용하고 있는데 이때 적용하는 regular라는 계수를 사용한다. 이 실험에서 regular값은 0.1부터 1까지 0.01단위로 변화시키며 변화를 살펴보았다. 이 regular에 따른 실험 결과의 경우 역시 regular=0.31까지는 RMSE가 점차 감소하지만 그 이후로는 RMSE가 증가하는 오버피팅 문제가 발생하였다.

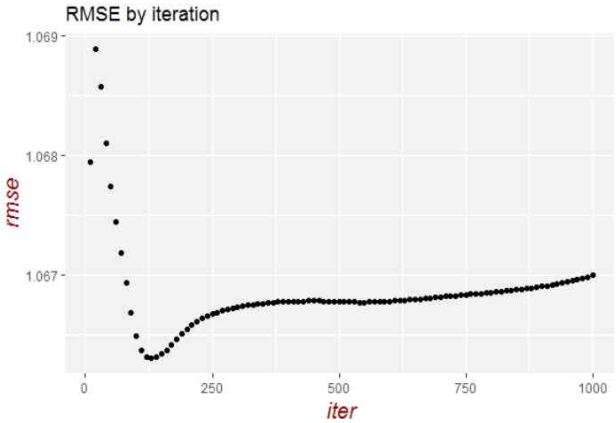

그림 4. 반복회수 최적화 (x축은 반복회수, y축은 RMSE 값임)
Figure. 4. Optimization of iteration

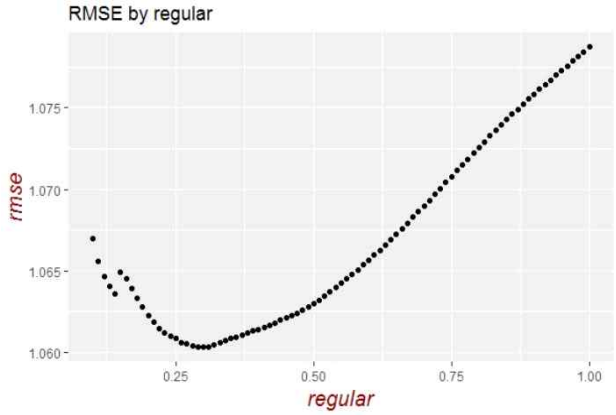

그림 5. 정규화 최적화 (x축은 정규화 계수, y축은 RMSE 값임)
Figure. 5. Optimization of regular

3) 시간 변수를 고려한 FM 성능비교

데이터에서 시간에 대한 변수는 사용자가 한 상품에 대한 평가를 내리기 전에 다른 상품에는 어떤 평가를 했는지 관찰할 수 있게 한다. 이러한 시간 변수를 고려했을 때와 고려하지 않았을 때 추천시스템의 성능에 차이가 있는지 살펴보기 위해 실험을 실시하였다. 실험결과 시간을 고려했을 때, 추천의 성능이 향상되는 것을 알 수 있었다. 표 2를 보면 시간을 고려했을 때 RMSE 값이 1.060093으로 시간을 고려하지 않았을 때의 1.0696보다 작음을 알 수 있다.

표 2. A회사 데이터의 RMSE값 비교
Table. 2. Comparison of RMSE-value in A company

	considering time variable	not considering time variable
RMSE-value	1.060093	1.06096
Performance is 0.08% better, when considering time variable		

4) 다른 알고리즘과의 RMSE 값 비교

다른 알고리즘과의 비교를 위해 반복회수 값과 정규화 계수 값을 최적화 한 FM의 RMSE값은 SVD에 비해 7.7%의 성능향상, SVM에 비해 9.1%의 성능향상을 보였다.

표 3. 각 알고리즘의 RMSE 비교
Table. 3. RMSE of each algorithm

Algorithm	RMSE-value
SVD	1.142661
SVM	1.157557
FM	1.060093

Ⅴ. 결론

온라인과 스마트 기기를 이용한 상품 구매가 빈번하게

일어나면서 사용자가 원하는 상품을 실시간으로 정확하게 추천해 주는 것은 궁극적으로 기업의 이윤을 극대화 시킬 수 있고, 이를 통해 추가적인 데이터를 지속적으로 확보할 수 있다. 이러한 이유로 상품 추천에 대한 중요성이 점점 부각되고 있으며, 추천 시스템에 대한 연구는 계속 진행 중에 있다.

본 논문에서는 실제 기업데이터의 평점 추천 모형에 FM 알고리즘을 적용하였다. 그 중 알고리즘의 최적화에 영향을 미치는 hyperparameter인 iter 및 regular를 변화시키며 RMSE 값을 비교하여 RMSE 값이 최소가 되는 값을 구함으로서 최적의 FM을 설정하였다.

또한 상품 추천의 정확도를 비교하기 위하여 시간 변수에 따라 변화하는 추천시스템의 정확도를 측정하였다.

한편 기존의 추천 알고리즘 중 SVD 값과 SVM과의 비교를 통해 추천 성능을 비교한 결과 각각 7.7%, 9.1%의 성능향상을 달성하였다.

이를 바탕으로 특정 응용에 사용될 수 있는 평점 추천 모형 응용에 적합한 알고리즘에 대한 가이드라인을 제시하였다.

참고문헌

[1] P. Paatero and U. Tapper, “Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values,” *Environmetrics*, vol. 5, no. 2, pp. 111–126, 1994.

[2] S. Rendle, L. B. Marinho, A. Nanopoulos, and L. Schmidt-Thieme, “Learning optimal ranking with tensor factorization for tag recommendation,” in *Proceeding of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 727-736, 2009.

[3] S. Rendle and L. Schmidt-Thieme, “Pairwise interaction tensor factorization for personalized tag recommendation,” in *Proceedings of the ACM International Conference on Web Search and Data Mining*, pp. 81-90, 2010,

[4] J. S. Breese, D. Heckerman, and C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering,” in *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, pp. 43-52, 1998.

[5] R. J. Mooney and L. Roy, “Content-based book recommendation using learning for text categorization,” in *Proceedings of the ACM Conference on Digital Libraries*, pp. 195-204, 2000.

[6] S. Banerjee and A. Roy, *Linear Algebra and Matrix Analysis for Statistics*, Chapman and Hall/CRC, 2014.

[7] Y. Koren, R. Bell, and C. Volinssky, Matrix factorization technique for recommender filtering, IEEE Computer, vol. 42, no. 8, pp. 30-37, 2009.

[8] R. H. Keshavan and S. Oh, “A gradient descent algorithm on the grassman manifold for matrix completion." arXiv preprint arXiv:0910.5260, 2009.

[9] S. Rendle, "Factorization machines," in *Proceedings of the IEEE International Conference on Data Mining*, pp. 995-1000, 2010.

[10] K. Madsen, H. B. Nielsen, and O. Tingleft, Methods for Non-Linear Least Squares Problems, Lecture Note, Informatics and Mathematical Modelling, Technical University of Denmark, 2004.

[11] S. K. Trivedi, *Probability and Statistics with Reliability, Queueing, and Computer Science Applications*, John Wiley & Sons, 2002.

[12] K. Mohan and J. Pearl, “On the testability of models with missing data,” in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pp. 643-650, 2014.

[13] I.-H. Jung, “The phase space analysis of 3D vector fields,” *Journal of Digital Contents Society*, vol. 16, no. 6, pp. 909-916, 2015.

[14] T. Hastie, S. Rosset, R.Tibshirani, and J. Zhu. “The entire regularization path for the support vector machine,” *Journal of Machine Learning Research*. vol. 5, pp. 1391–1415, 2004.

정승윤(Seung-Yoon Jeong)

2015년 : 숭실대학교 정보통계보험수리학과 (학사)
2017년 : 고려대학교 정보보호대학원 (석사과정)

2015년～현 재: 매일경제신문사
※관심분야 : 추천시스템, 머신러닝, 빅데이터분석, 데이터시각화, AI 등

김형중(Hyung-Joong Kim)

1978년 : 서울대학교 전기공학과 학사
1986년 : 서울대학교 제어계측공학과(공학석사)
1989년 : 서울대학교 제어계측공학과(공학박사)

1989년～2006년: 강원대학교 교수
2006년～현 재: 고려대학교 정보보호대학원 교수
관심분야 : 컴퓨터보안, 패턴인식, 가역정보은닉, 머신러닝, 빅데이터분석 등