Utilizing Textual Reviews in Latent Factor Models for Recommender Systems

Tatev Karen Aslanyan
Erasmus Univeristy Rotterdam
Data Scientist at Elsevier
tatevkaren@gmail.com

Flavius Frasincar
Erasmus Univeristy Rotterdam
Assistant Professor
frasincar@ese.eur.nl

36th ACM/SIGAPP Symposium on Applied Computing (SAC 2021)

Outline

- Motivation
- Related Work
- Methodology
- Evaluation
- Applied Data Analysis on Amazon Data
- Conclusion and Future Work

Motivation

Due to efficiency and the ease of use, online shopping and services gained large popularity

- During last 5 years e-commerce shares in global retail sales increased 7.4% → 20%

Large amount of online stores and product variations has led to information overload

- Makes online shopping less pleasant and convenient
- Businesses rely on Recommender Systems to solve information overload

Online stores have platforms to collect feedback from about their products and services

- Ratings
- Reviews
- Customer characteristics (age, gender)
- Product characteristics (genre, author, origin, color)

Motivation

Recomender Systems categories

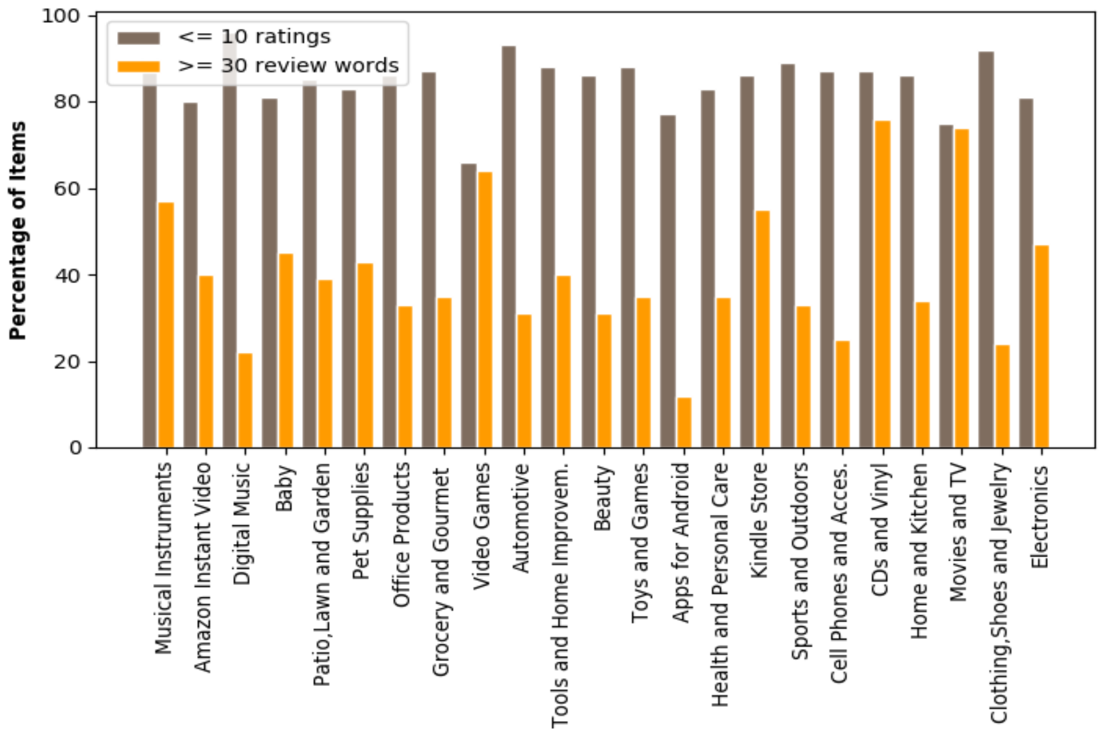
- Collaborative Filtering (rating based)
- Content – Based (review based)
- Hybrid (rating and review based)

Most of Recommender Systems

- Rating based and not scalable

Reviews contain large amount of information

- Can help to better predict customer preferences
- Can complement the absence of product ratings



What We Do

New Recommender System LDA-LFM

- Product ratings
- Product reviews
- Allows adding extra user or item characteristics
- Scalable

- Latent Dirichlet Allocation (LDA)

  topic modelling technique

- Latent Factor Model (LFM)

  rating modelling technique

Generalization: LDA-LFM can also be applied to recommend online services

Related Work

- **Collaborative Filtering**

  (Koren et al., 2009): Recommender algorithm combining LFM and neighborhood based approach to genereate item recommendations

- **Content-Based**

  (Mooney and Roy, 2000): One of the first content-based algorithms to generate book recommendations

- **Hybrid Recommenders**

  (McAuley and Leskovec, 2013): Hidden Factors and Topic (HFT) hybrid recommender combining LFM and LDA to generate article recommendation

  (Ling et al., 2014): Ratings Meet Reviews (RMR) hybrid recommender combining LFM and LDA to generate article recommendation

Methodology

Building Blocks of LDA-LFM

- Latent Factor Models (LFM)

  (for modelling the ratings)

- Latent Dirichlet Allocation (LDA)

  (for modelling the reviews)

- Combining LFM and LDA

- Allowing to add extra user and item features

Latent Factor Model (LFM)

Rating modelling technique

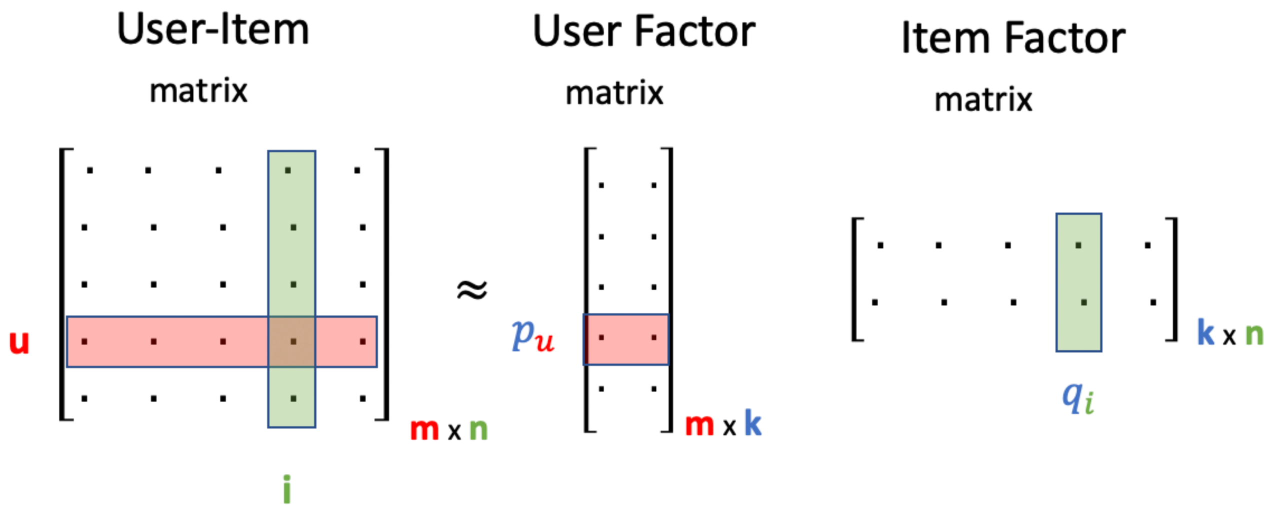
User – Item rating matrix is sparse

- m users and n items

Decomposing User – Item rating matrix into

2 smaller and denser matrices

- User Factor matrix
- Item Factor matrix



$$\hat{r}_{ui} = q_i^T p_u$$

Latent Factor Model (LFM)

Some customers tend to give higher rates

- User bias

Some products tend to be rated higher

- Item bias

$$\hat{r}_{ui} = \alpha + b_u + b_i + q_i^T p_u$$

$$e_{ui} = r_{ui} - \hat{r}_{ui}$$

Latent Factor Model (LFM)

Generalization from one pair of user and item to the entire sample

$$e_{ui} = r_{ui} - \hat{r}_{ui}$$

- Minimize the quadratic loss function

To solve the optimization problem, Adam Optimizer is used

- Closely related to Stochastic Gradient Decent (SGD)
- Faster and less prone to errors

$$\arg\min \frac{1}{|\mathcal{T}|} \sum_{u,i \in \mathcal{T}} (e_{ui})^2 + \lambda \left(\|p_u\|_2^2 + \|q_i\|_2^2 + \|b_u\|_2^2 + \|b_i\|_2^2\right)$$

Latent Dirichlet Allocation

LDA relies on 4 concepts

1. Words carry strong semantic information
2. Documents discussing similar topics are likely to use similar words
3. Documents are probability distributions of words
4. Topics are probability distributions of words

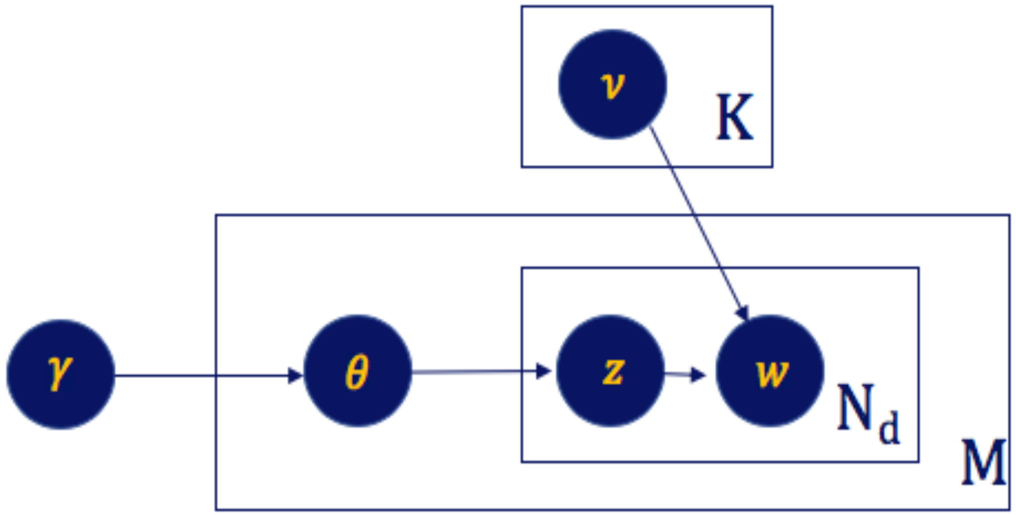
Example of the topic about “animals”

- Words “zoo” and “species” will have high probability

Latent Dirichlet Allocation

Corpus Entity

- Collection of M documents



Document Entity

- Sequence of N words
- All reviews for single item

Word Entity

- Each word in a document has its position

Latent Dirichlet Allocation

Topic distribution of document d / item i

- $\theta_d = \theta_i$

Corpus Likelihood $p(\mathcal{T} \mid \theta, \varphi, z) = \prod_{d \in \mathcal{T}} \prod_{j=1}^{N_d} \theta_{d, z_{d,j}} \varphi_{z_{d,j}, w_{d,j}}$

Log Corpus Likelihood $\ell(\mathcal{T} \mid \theta, \varphi, z) = \sum_{d \in \mathcal{T}} \sum_{j=1}^{N_d} log(\theta_{d, z_{d,j}} \varphi_{z_{d,j}, w_{d,j}})$

Combining LDA and LFM

Key assumption

- Properties of a product correspond to certain topics
- These topics will be discussed in product reviews

Positive correlation between item property and review topic

$$\theta_{i,k} = \frac{\exp(kq_{i,k})}{\sum_{l=1}^{K} \exp(kq_{i,l})}$$

$$\sum_{k} \theta_{i,k} = 1$$

$$q_i \in \mathrm{R}^{\mathrm{K}}$$

LDA-LFM

Objective function of LDA - LFM

$$f(\mathcal{T}\,|\alpha, b_u, b_i, p_u, q_i, \mathrm{k}, \theta, \varphi, z) = \sum_{u,i\in\mathcal{T}}(e_{ui})^2 + \lambda\,(\|p_u\|_2^2 + \|b_u\|_2^2 + \|b_i\|_2^2) - \mu\ell\,(\mathcal{T}\mid\theta, \varphi, z)$$

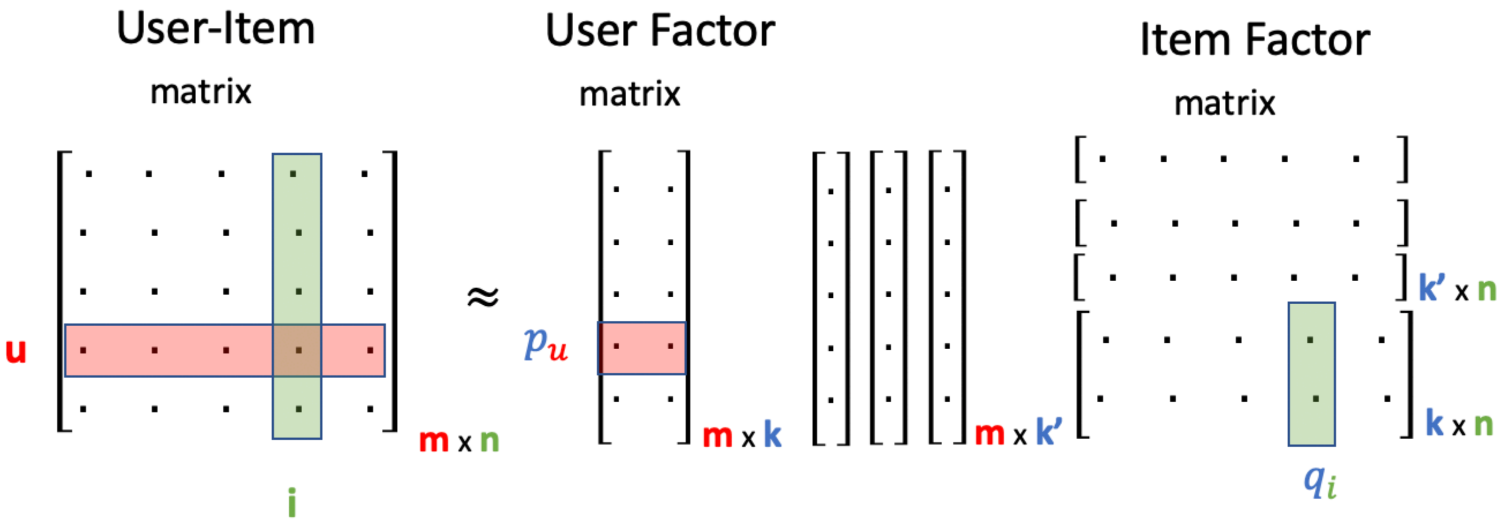
LFM
Latent Factor of Ratings

LDA
Latent Factor of Reviews

Adding extra user- and item-features

Extra features added to the LFM part of the model

- Extra user features added as extra columns to User Factor Matrix
- Extra item features added as extra rows to Item Factor Matrix
- Number of extra user features shoud be equal to extra item features



Evaluation

Offset Model $\hat{r}_{ui} = \alpha$

Baseline Rating Model (BRM) $\hat{r}_{ui} = \alpha + \overline{r_u} + \overline{r_i}$

Latent Factor Model (LFM) $\hat{r}_{ui} = \alpha + b_u + b_i + q_i^T p_u$

LDAFirst

- Topic probabilities are sampled once and stay constant

Evaluation metrics

- Mean Squared Error (MSE)

Applied Data Analysis on Amazon Data

Amazon Web Shop Data

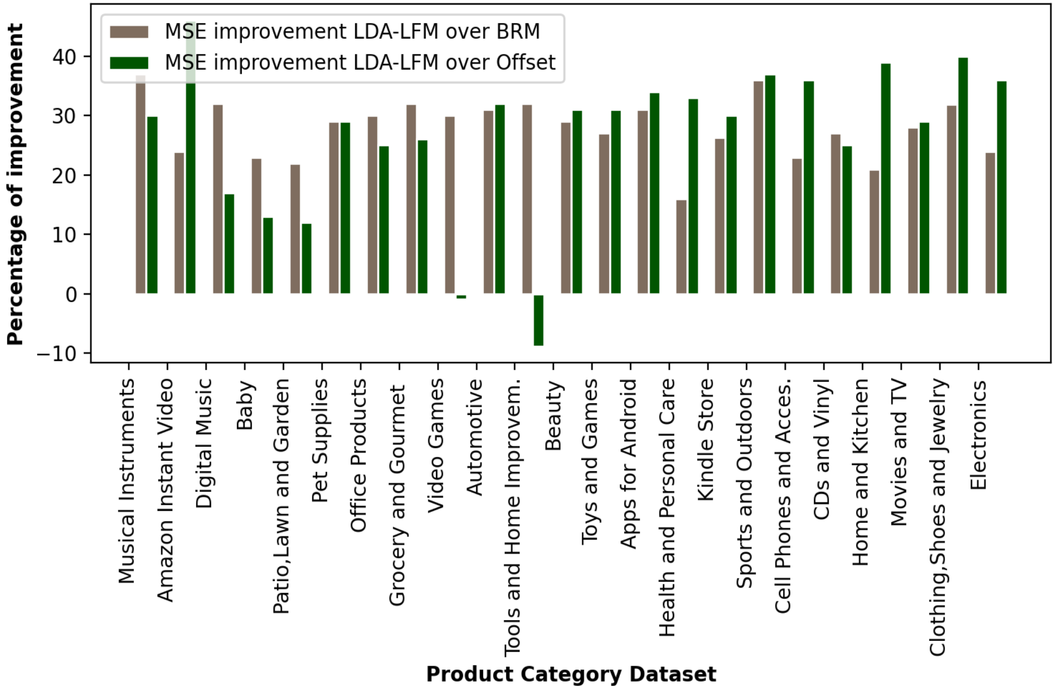
- 23 product categories
- Collected in the period of 1996 – 2014
- Feedback data of 143M (e.g., ratings, reviews, helpness score)
- Metadata of 9.4M products (e.g., price, brand)

Dataset	Nusers	Nitems	Nreviews	Avg Words	Avg Rating	Sparsity
Electronics	4.2M	0.5M	7.8M	43	4.0	0.00039
Clothing, Shoes and Jewelry	3.1M	1.1M	5.8M	26	4.2	0.00016
Instant Videos	0.4M	0.02M	0.6M	28	4.3	0.00571
Musical Instruments	0.3M	0.08M	0.5M	45	4.2	0.00178

Performance of LDA – LFM

Comparing LDA-LFM to Offset

- At least 10% improvement for all datasets except for *Beauty*
- For some cases more than 30% improvement



Comparing LDA-LFM to BRM

- At least 15% imrpovement for all datasets
- For some cases more than 30% improvement

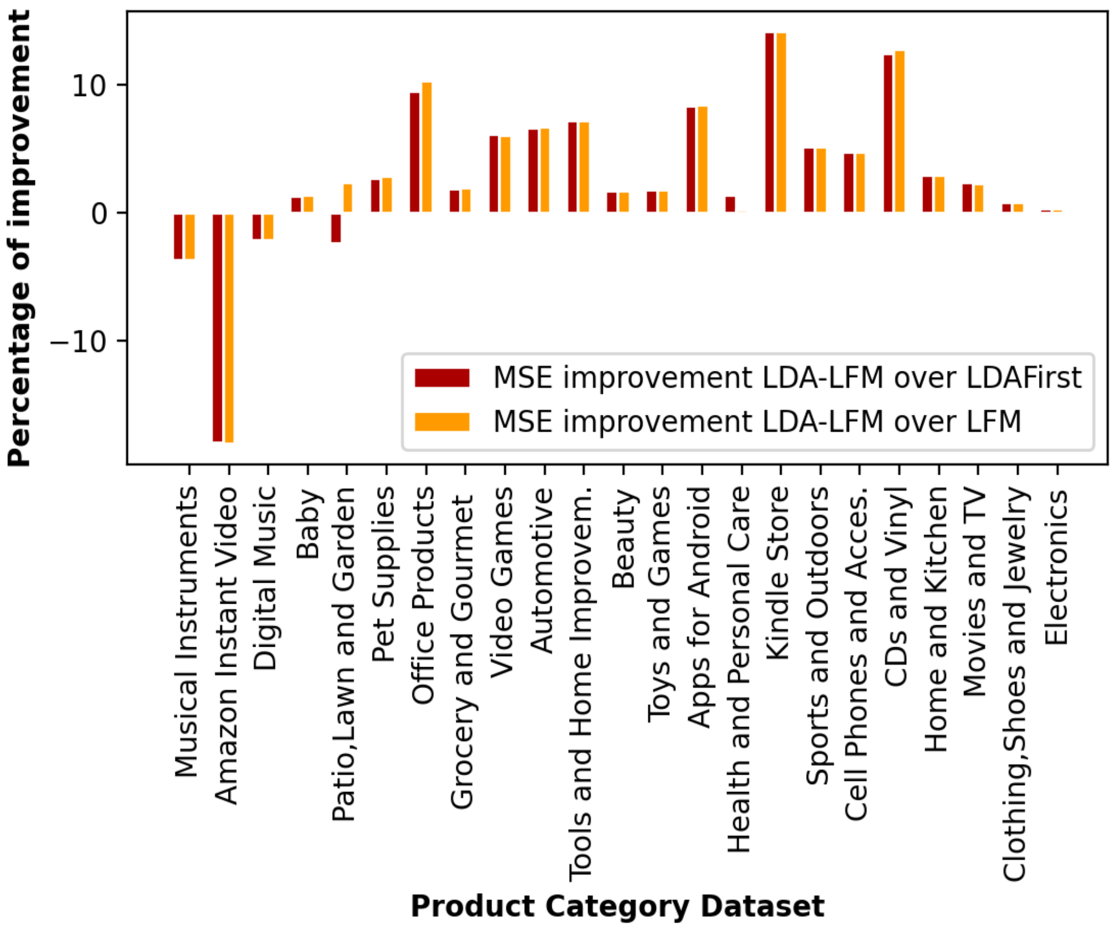
Performance of LDA – LFM

Comparing LDA-LFM to LFM

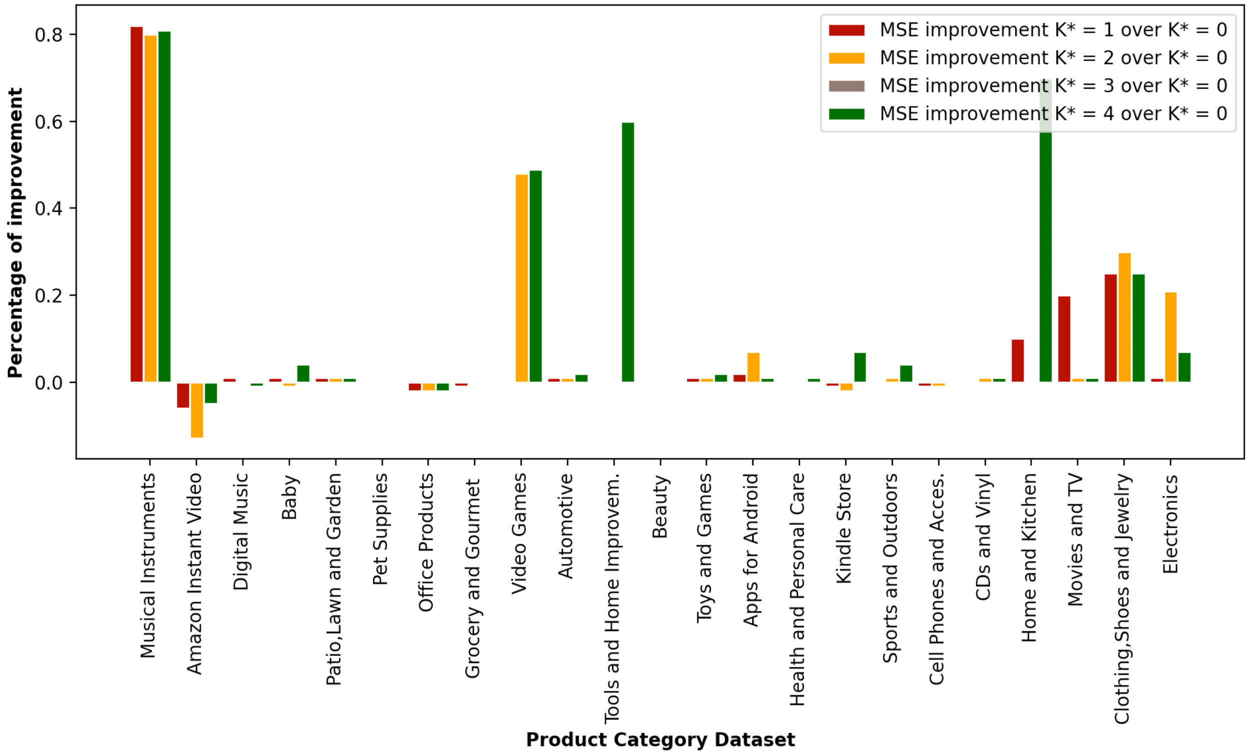
- Improvement for all datasets except for smallest 3
- Significant decrease in MSE for medium or large datasets (e.g. Kindle Store of 14%)

Comparing LDA-LFM to LDAFirst

- Improvement for majority
- Significant decrease in MSE for medium or large datasets (e.g. Kindle Store of 14%)



Performance of LDA – LFM



Adding extra features to LDA-LFM

- Positive improvement for most of the datasets
- More extra features have bigger impact for some datasets

Conclusion and Furture Work

Main Take-aways

Using textual reviews improves the quality of the recommendations

Adding extra user- and item-features often improve recommendations

LDA-LFM is scalable (able to handle millions of observations)

Future Work

Use sentiment analysis for textual review

- (e.g., classifying topics-sentiments as positive or negative)

Combine implicit user and item features from reviews

- (e.g., the gender or age of the reviewer)

References

Koren, Y., Bell, R., and Volinsky, C. (2009). Matrix factorization techniques for recommender systems. Computer, pages 30-37

Mooney, R., and Roy, L. (2000). Content-based book recommending using learning for text categorization. In the 5th ACM Conference on Digital Libraries (DL 2000), pages 195-204. ACM

McAuley, J. and Leskovec, J. (2013). Hidden factors and hidden topics: Understanding rating dimensions with review text. In 7th ACM Conference on Recommender Systems (RecSys 2013), pages 165-172. ACM

Ling, G., Lyu, M., and King, I. (2014). Ratings meet reviews, a combined approach to recommend. In 8th ACM Conference on Recomender Systems (RecSys 2014), pages 105-112. ACM