The ACM Conference Series on
Recommender Systems

Convolutional Matrix Factorization for Document Context-Aware Recommendation

Donghyun Kim[1], Chanyoung Park[1], Jinoh Oh[1], Sungyoung Lee[2], Hwanjo Yu*[1]

[1]Datamining Lab @ POSTECH

[2]Ubiquitous Computing Lab @ Kyunghee University

*corresponding author

Matrix Factorization (MF)

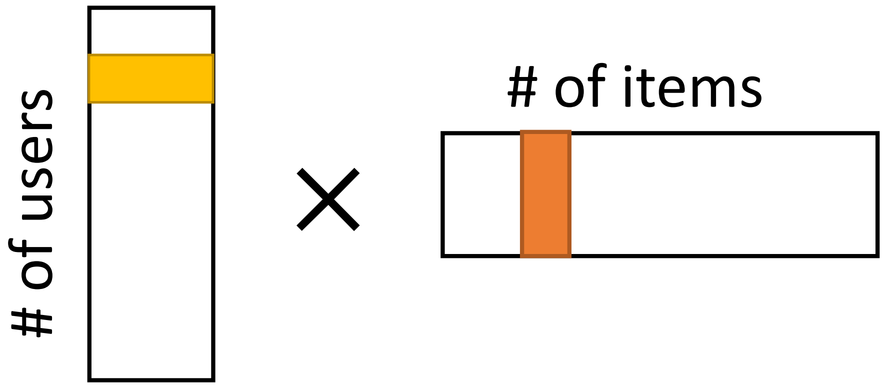
- A popular model-based collaborative filtering for recommendation

Items

Users

5	?	?	3
4	?	?	2
?	1	3	1

$\approx$



user latent models

item latent models

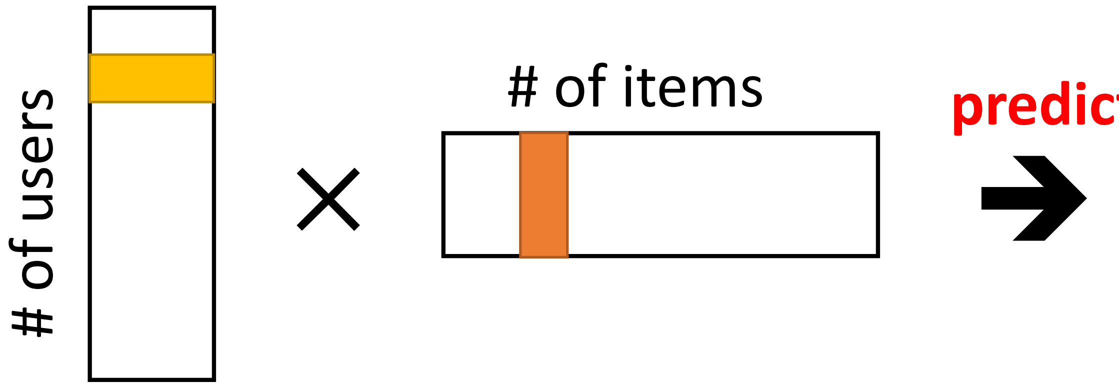
$$r_{ij} \approx u_i^T v_j$$

ratings

Matrix Factorization (MF)

- A popular model-based collaborative filtering for recommendation

matrix completion



Items

Users				
	5	?	?	3
	4	?	?	2
	?	1	3	1

user latent models

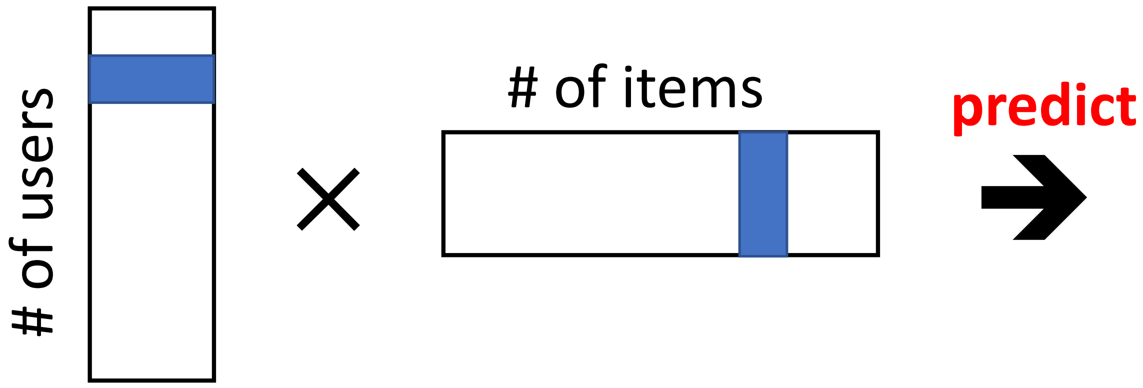
item latent models

$$\boldsymbol{u}_i^T \boldsymbol{v}_j = \hat{\boldsymbol{r}}_{ij}$$

predicted ratings

Matrix Factorization (MF)

- A popular model-based collaborative filtering for recommendation



matrix completion

Items

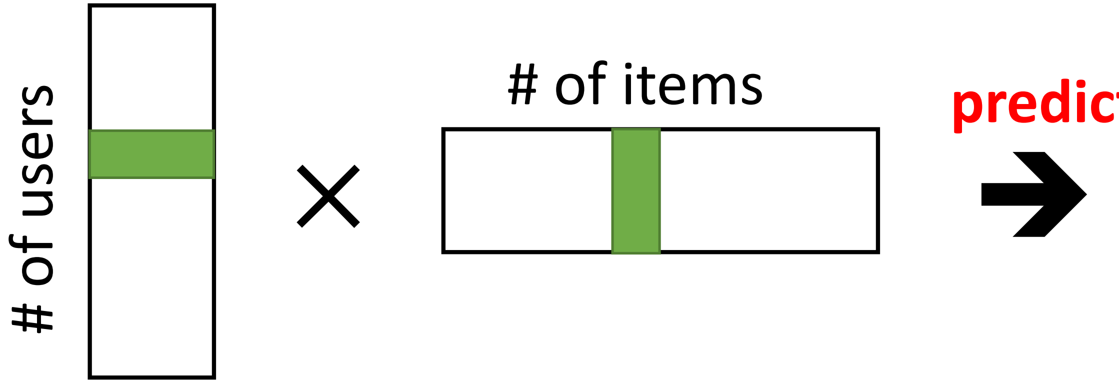
Users

5	?	4	3
4	?	?	2
?	1	3	1

$$u_1^T v_3 = \hat{r}_{1,3}$$

Matrix Factorization (MF)

- A popular model-based collaborative filtering for recommendation



matrix completion

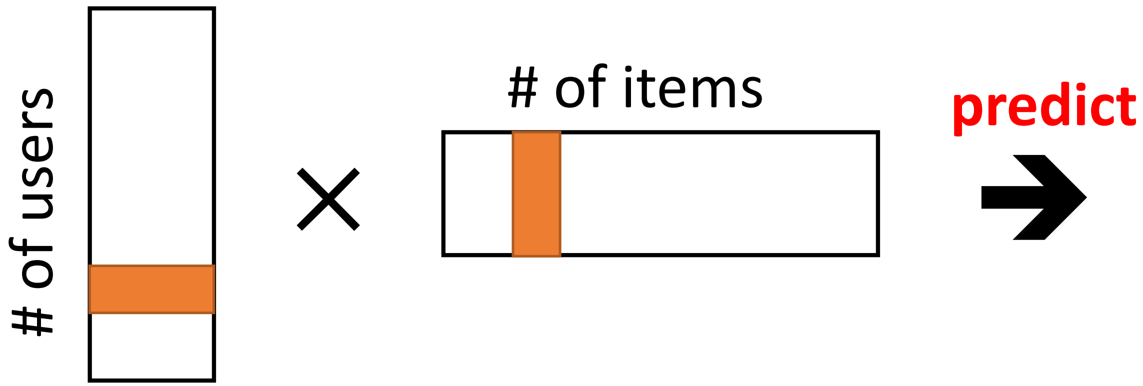
Items

Users				
	5	?	4	3
	4	2	?	2
	?	1	3	1

$$u_2^T v_2 = \hat{r}_{2,2}$$

Matrix Factorization (MF)

- A popular model-based collaborative filtering for recommendation



matrix completion

Items

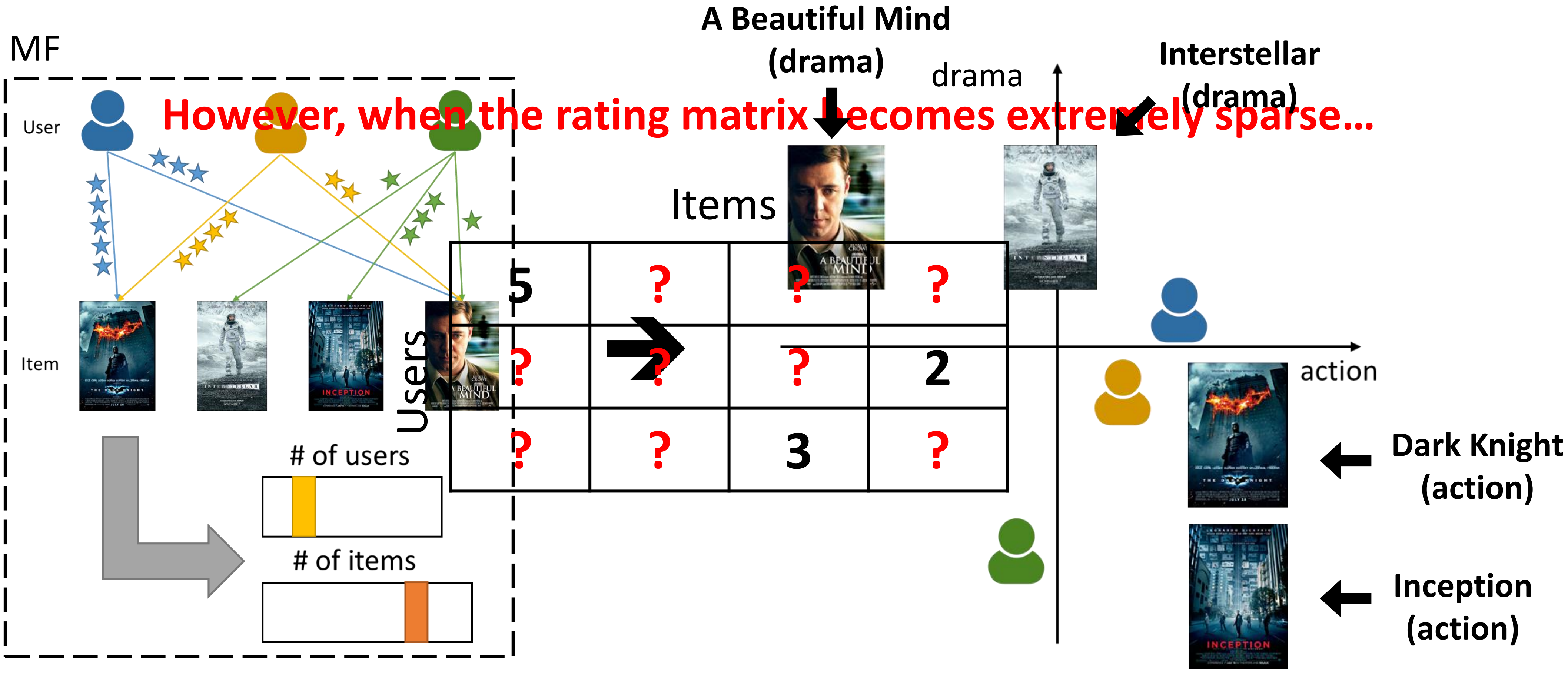
Users

5	?	4	3
4	2	?	2
3	1	3	1

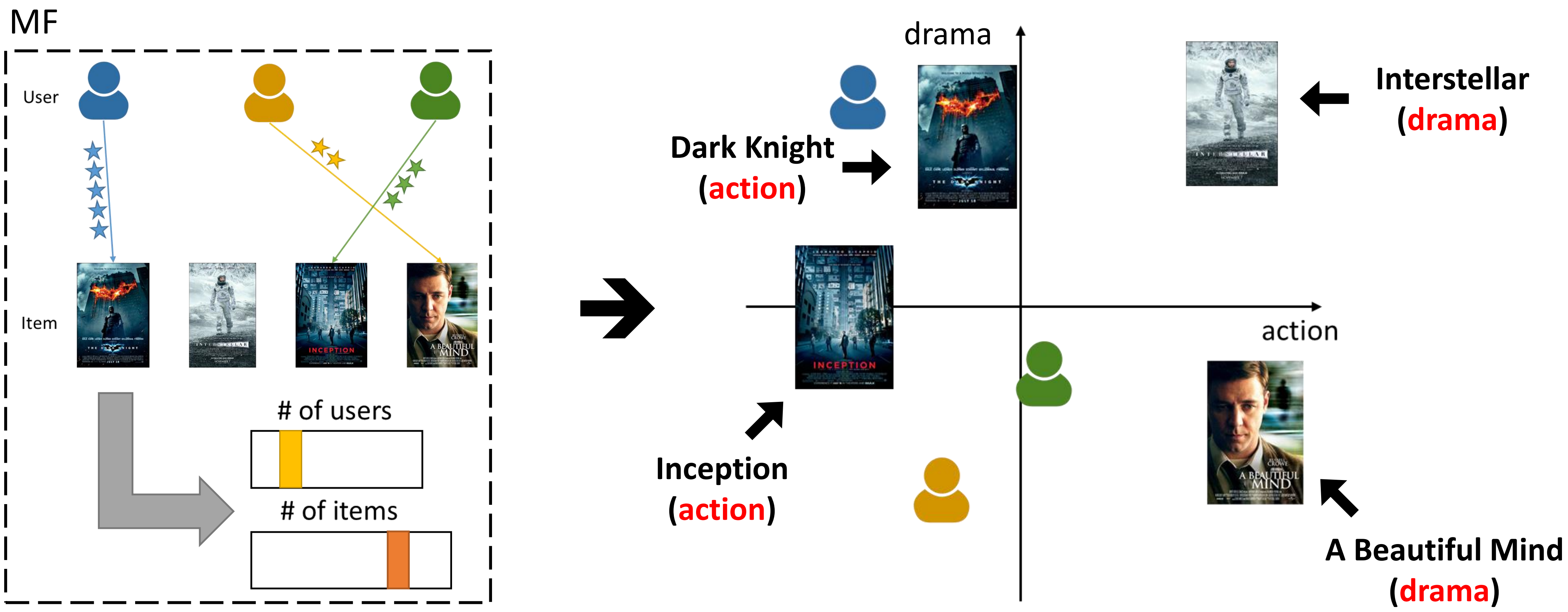
$$u_3^T v_1 = \hat{r}_{3,1}$$

User and item latent models in 2D space!

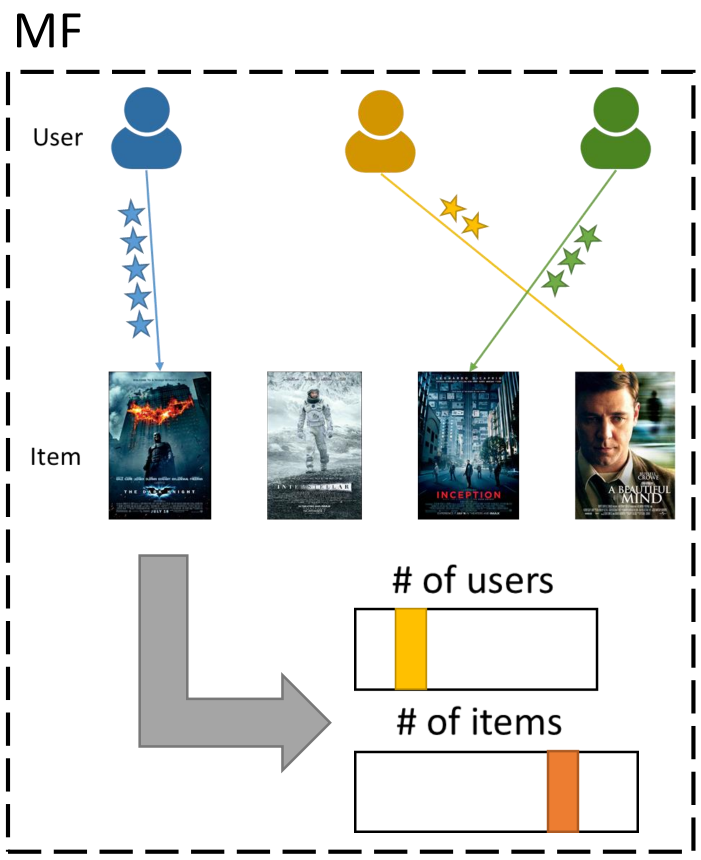
However, when the rating matrix becomes extremely sparse…

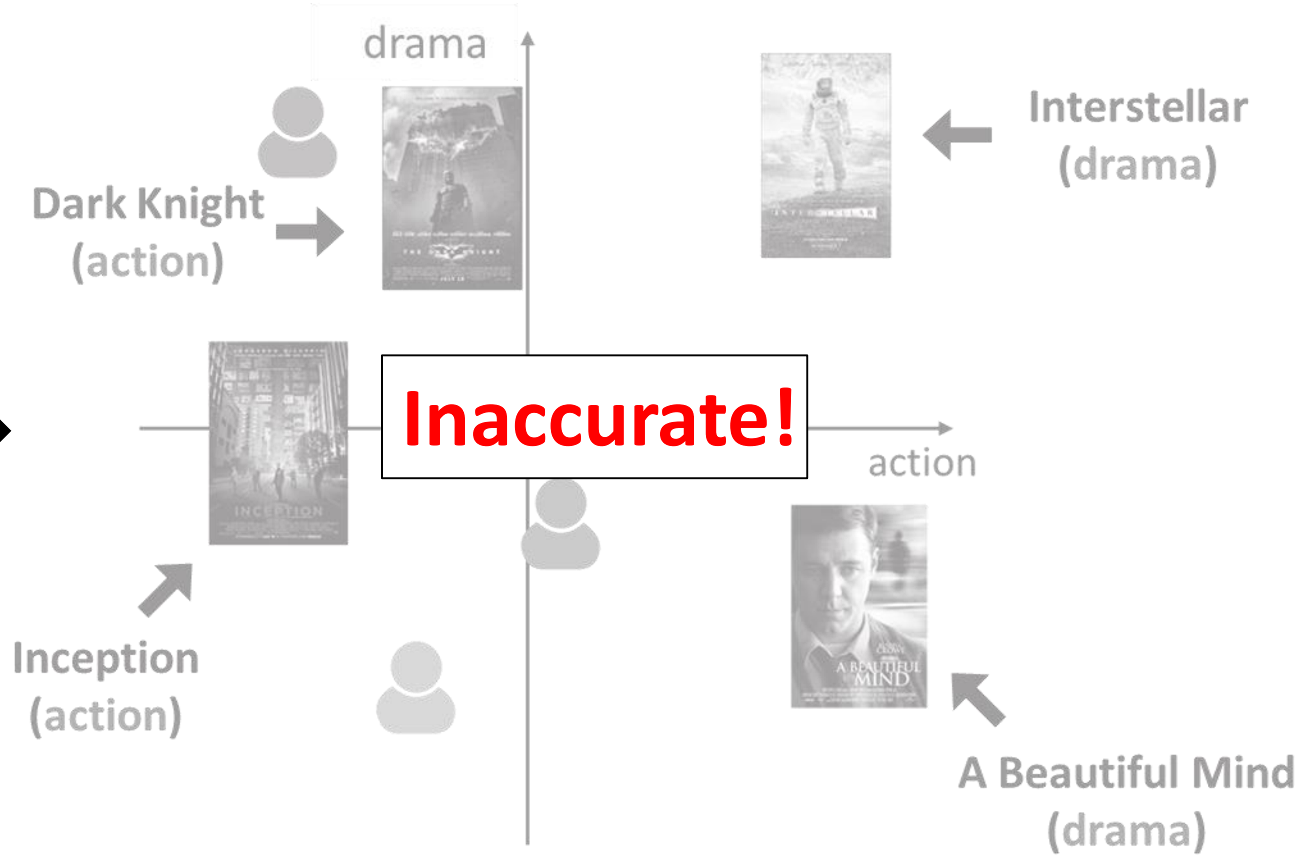


User and item latent models in 2D space!



User and item latent models in 2D space!





Common approaches

- To handle sparseness of a rating matrix, text information (review, synopsis, abstract, etc.) has been widely used in recent researches. [KDD`15, RecSys`14, RecSys`13, KDD`11]

a description document

Common approaches

- Trial to understand description documents for recommendation

Common approaches

- Trial to understand description documents for recommendation
  - *Collaborative topic modeling for scientific articles* (CTR) [KDD`11]
    - **Latent Dirichlet Allocation (LDA)**

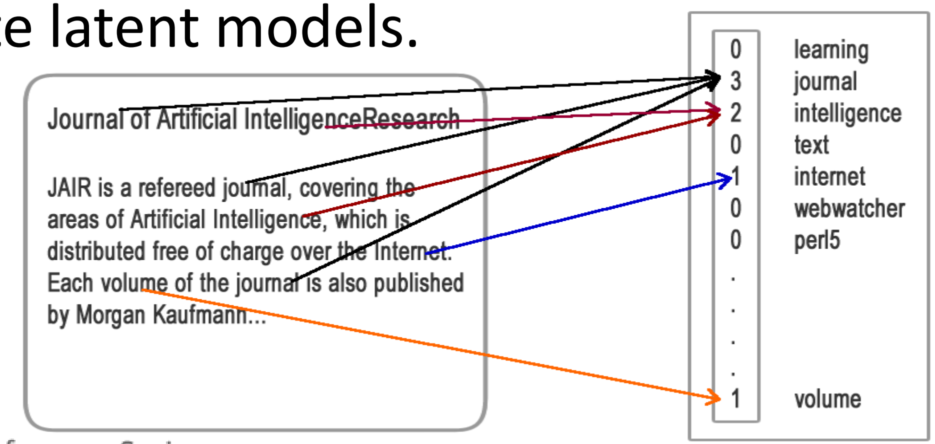
Common approaches

- Trial to understand description documents for recommendation
  - *Collaborative topic modeling for scientific articles* (CTR) [KDD`11]
    - **Latent Dirichlet Allocation (LDA)**
  - *Collaborative deep learning for recommender system* (CDL) [KDD`15]
    - **Stack Denoising AutoEncoder (SDAE)**

Drawback of common approaches

- Trial to understand description documents for recommendation
  - *Collaborative topic modeling for scientific articles* (CTR) [KDD`11]
    - **Latent Dirichlet Allocation (LDA)**
  - *Collaborative deep learning for recommender system* (CDL) [KDD`15]
    - **Stack Denoising AutoEncoder (SDAE)**

- However, LDA and SDAE analyze "**bag of words models**" of item descriptions to generate latent models.



Ignore

✓ surrounding words of a word
✓ word order

“Contextual information” in documents

- Considering surrounding words and word order as “contextual information” **improves the accuracy of word vectors** in the word embedding.
  - *Word2Vec* [NIPS`13]
- What if recommender systems are able to capture *contextual information* in documents?
  - **Generate more accurate item latent models** through a deeper understanding of item descriptions.
- Thus, **contextual information should be considered for better recommendation!**

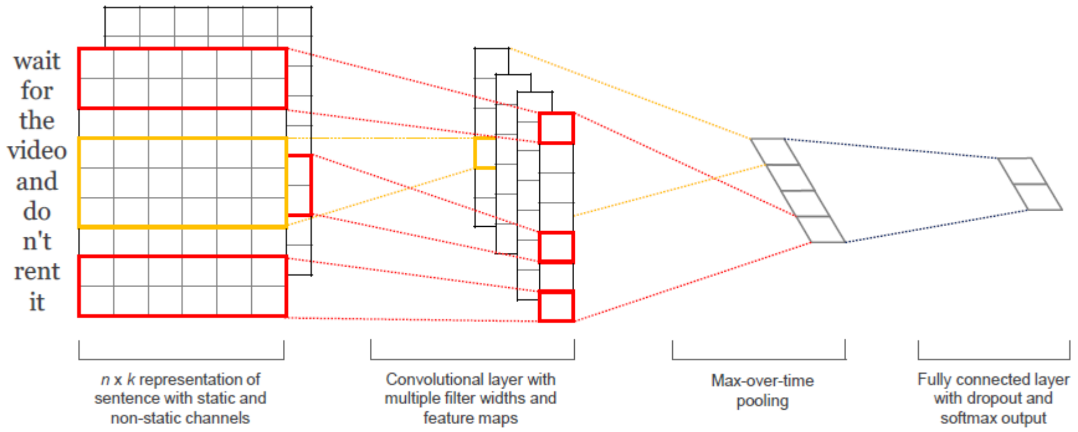
Our proposed model

- We develop a novel document context-aware recommendation model, **Convolutional Matrix Factorization (ConvMF)**.
  - To consider contextual information
  - To effectively exploit both ratings and description documents
  - To jointly optimize the recommendation model in order to properly predict ratings to items of users

Inspired by Convolutional Neural Network (CNN)

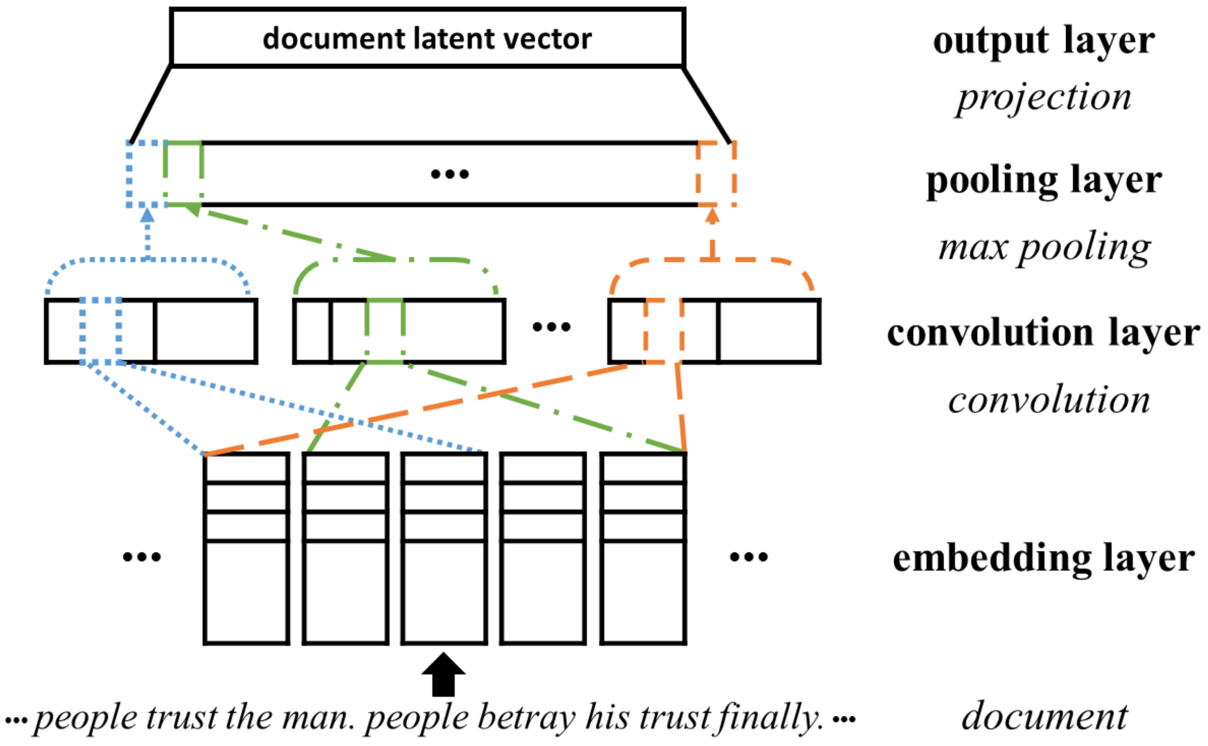
- For the NLP and IR tasks, convolutional neural networks (CNNs) have been mainly developed to consider local contextual information in a document.
  - NLP: [JMLR`11, ACL`14, EMNLP`14], IR: [EMNLP`14, CIKM`14]

- An example of CNN architecture for sentiment classification. [EMNLP 2014]



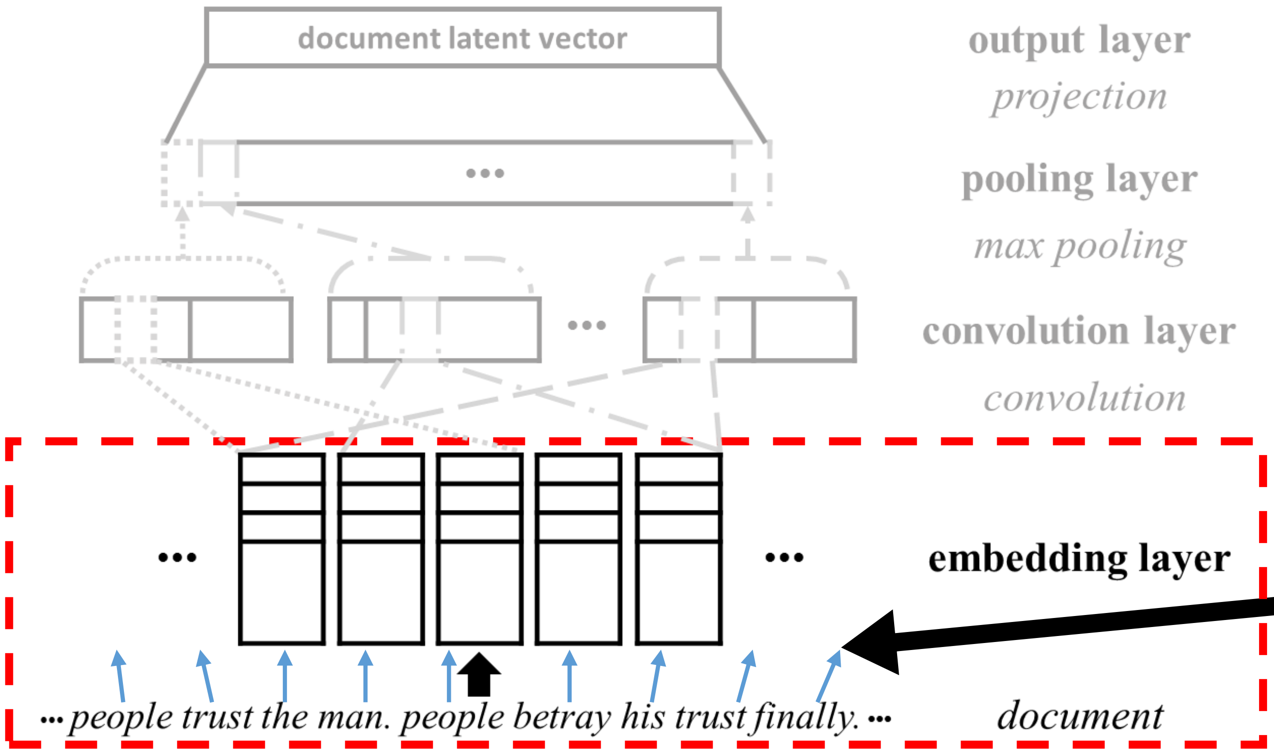
Overview of our CNN architecture

- Trial to generate more accurate item latent models



Embedding layer – *word embedding*

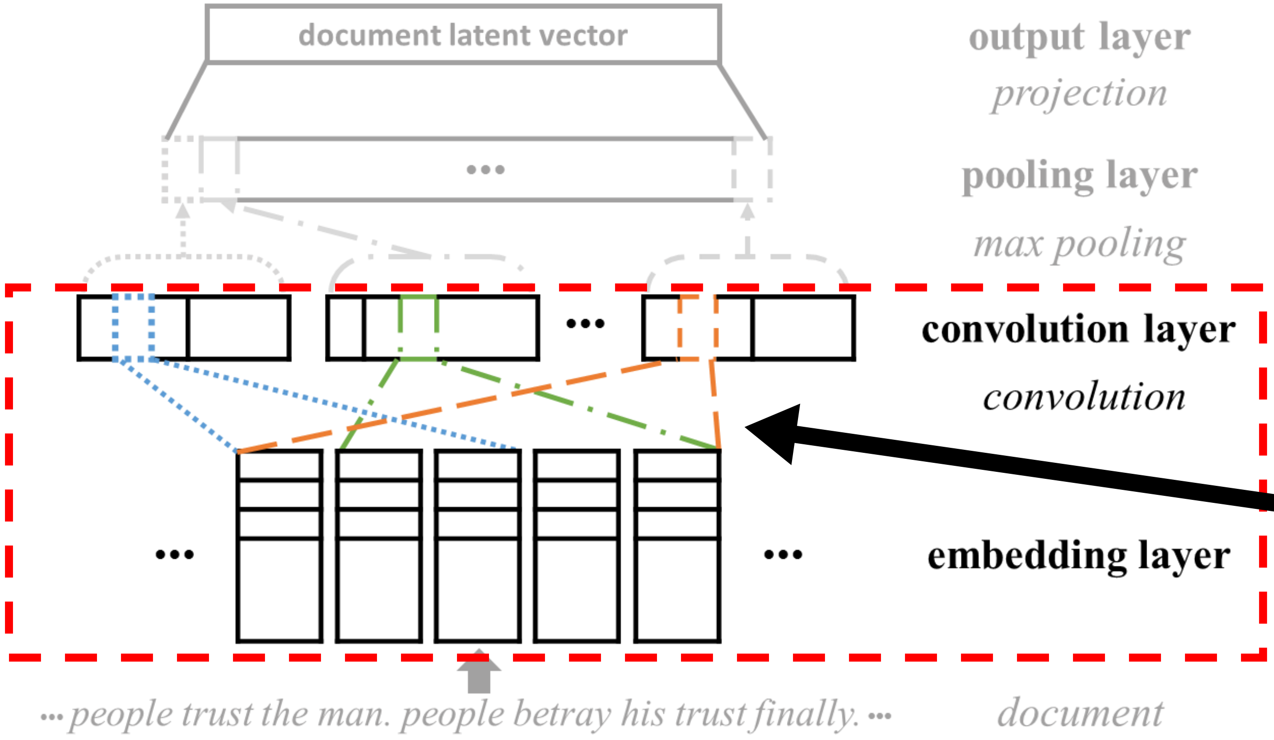
- Transform a raw description document into **a numeric document matrix.**



(*pre-trained*) word embedding models.

Convolution layer – *contextual information*

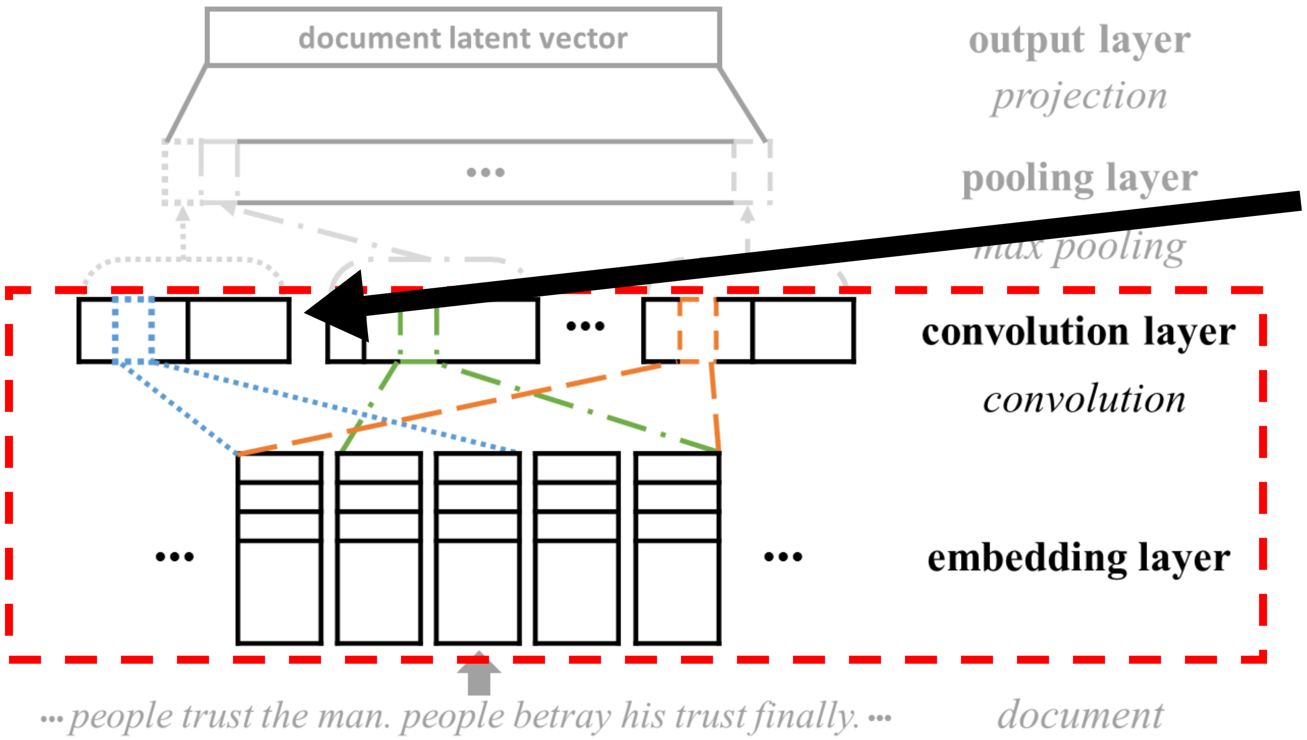
- Extract *contextual features* from a document matrix.



multiple shared weights (kernels)

Convolution layer – *contextual information*

- For example (window size: 3)



$$c = [c_1, c_2, \dots, c_i, \dots, c_{l-ws+1}]$$

c_2

... people betray his trust finally ...

c_3

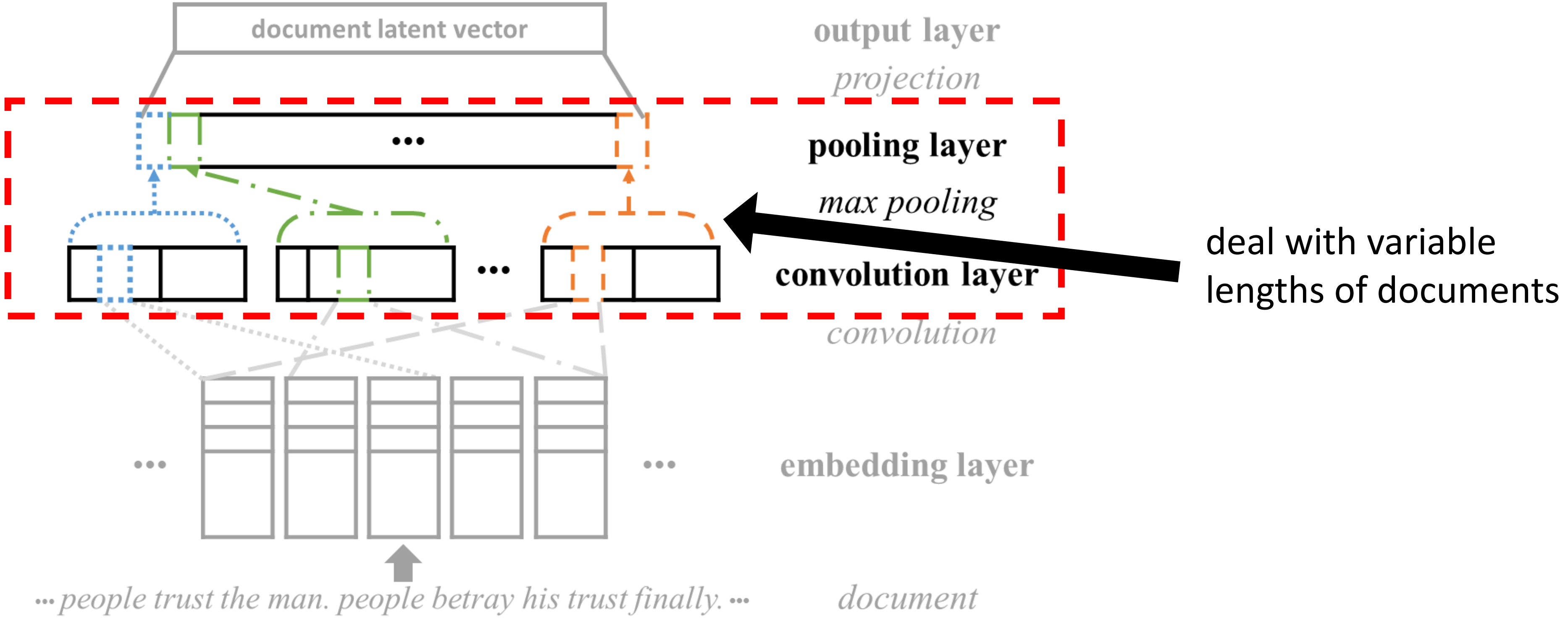
... people betray his trust finally ...

c_4

... people betray his trust finally ...

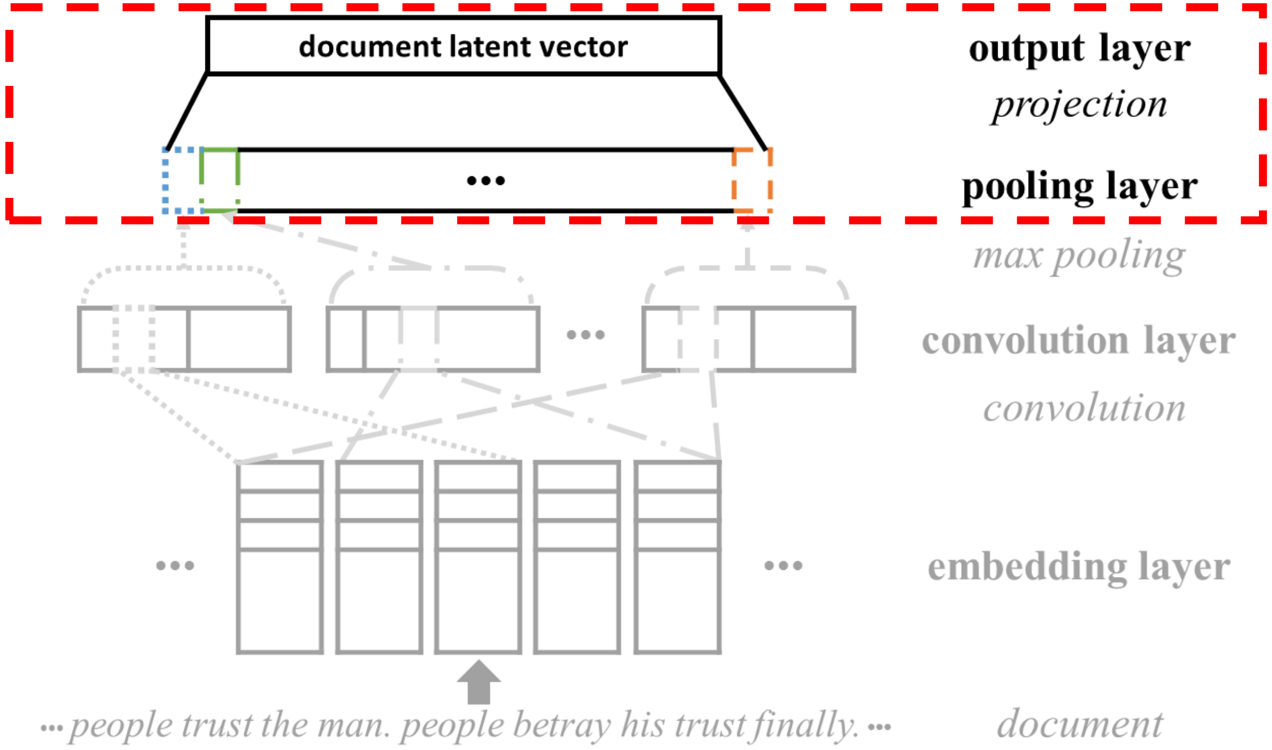
Pooling layer – representative information

- Extract representative features from the convolutional layer



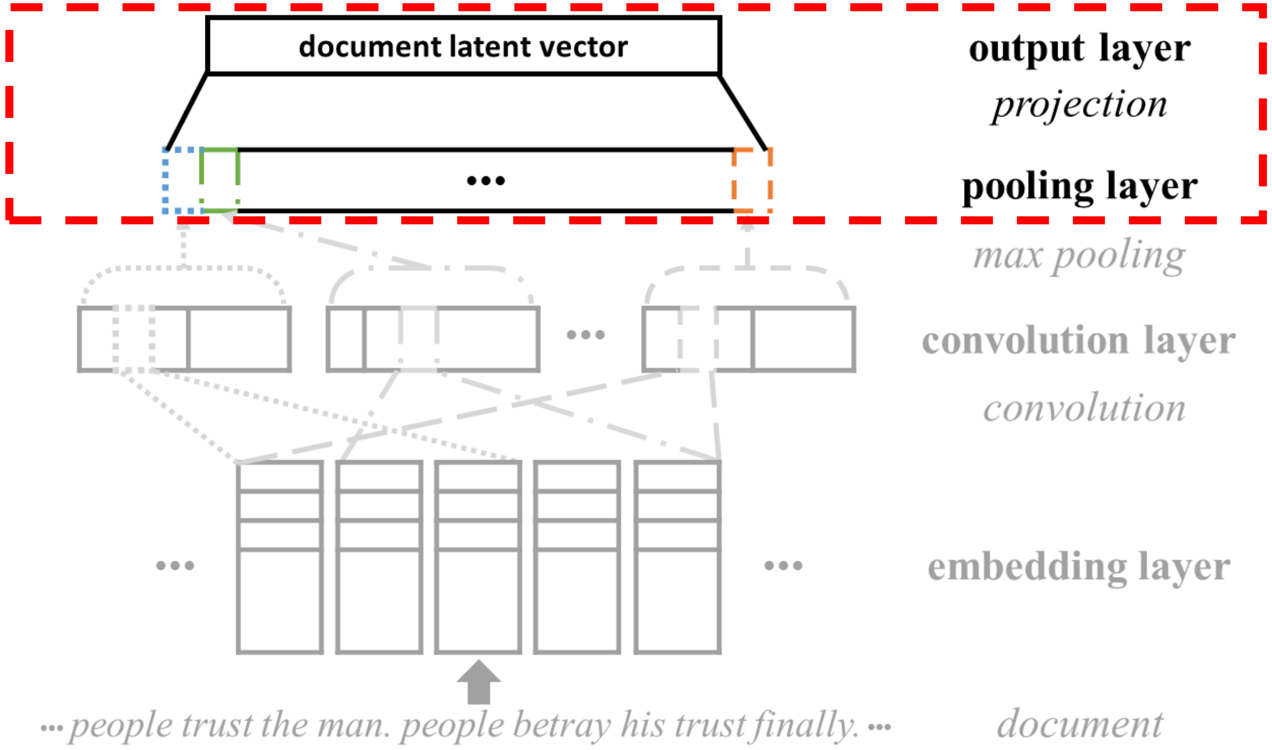
Output layer – high level features of documents

- Project representative features to a ***k*-dimensional space**



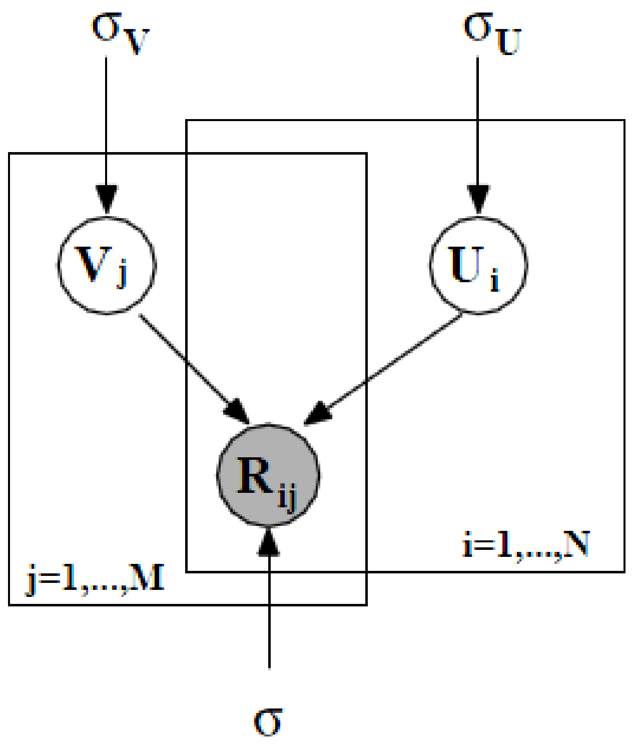
Then, how to predict ratings?

- However, the direct usage of CNNs is not suitable for a recommendation task.



Probabilistic Matrix Factorization (PMF) [NIPS`08]

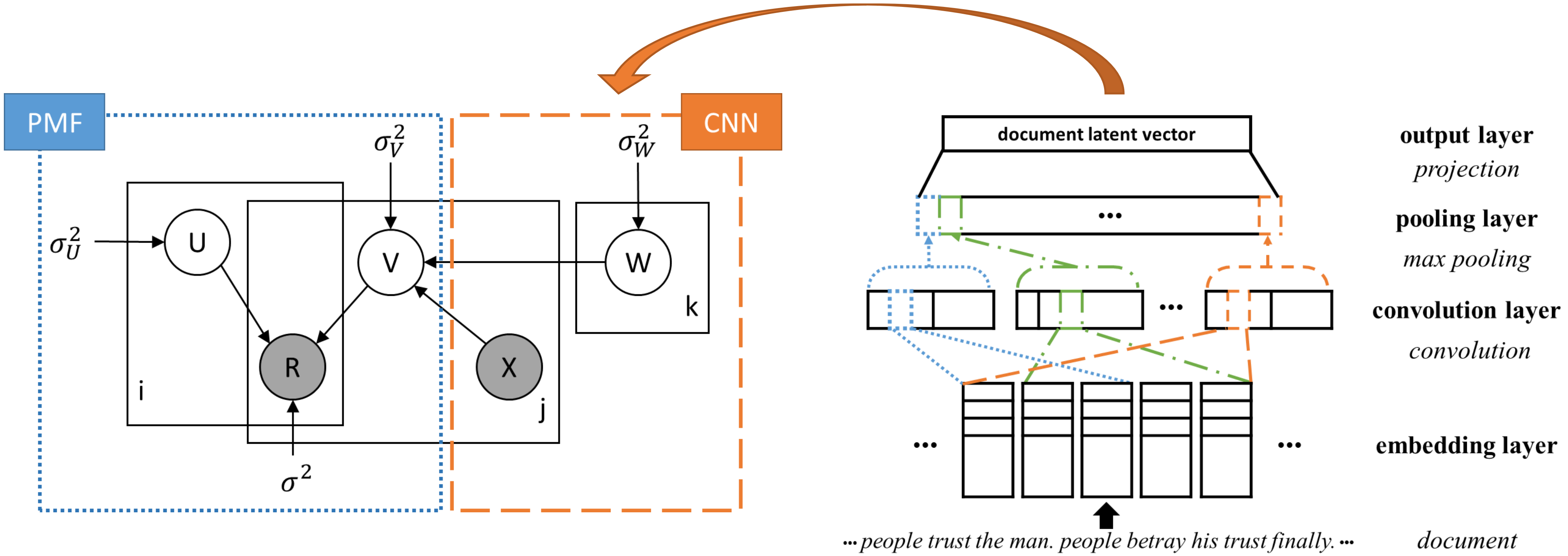
- Ratings can be approximated by probabilistic methods.



<The graphical model of PMF>

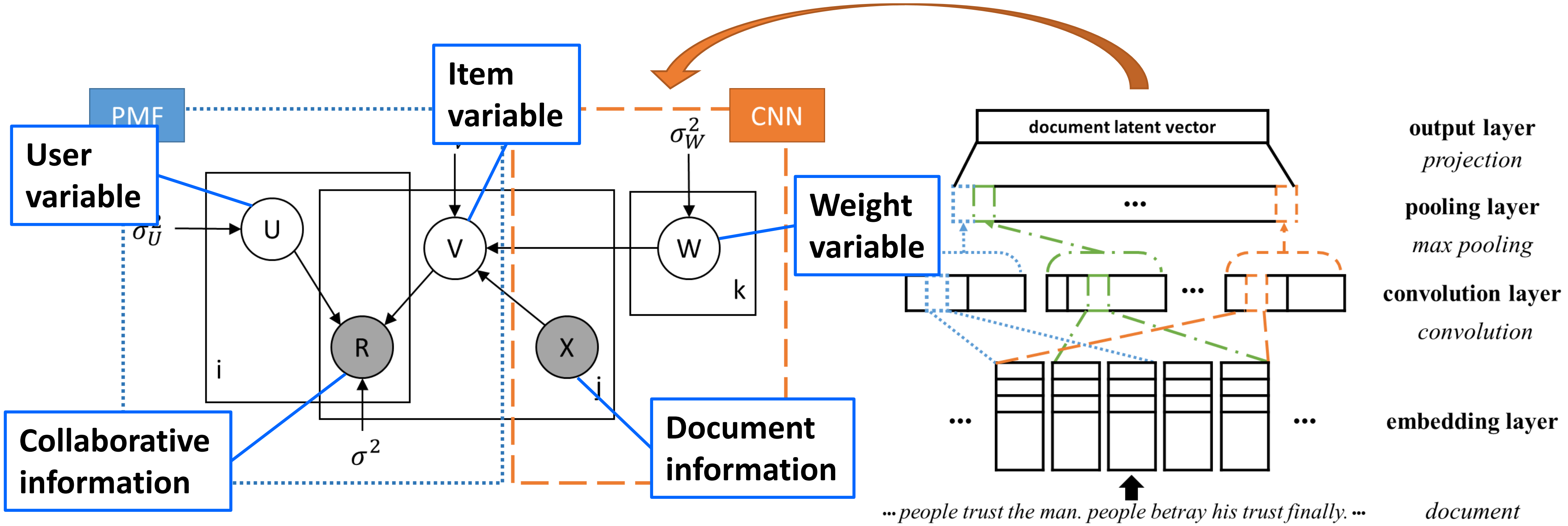
How about PMF + CNN?

- Overview of ConvMF
  - We integrate CNN into PMF for the recommendation task.



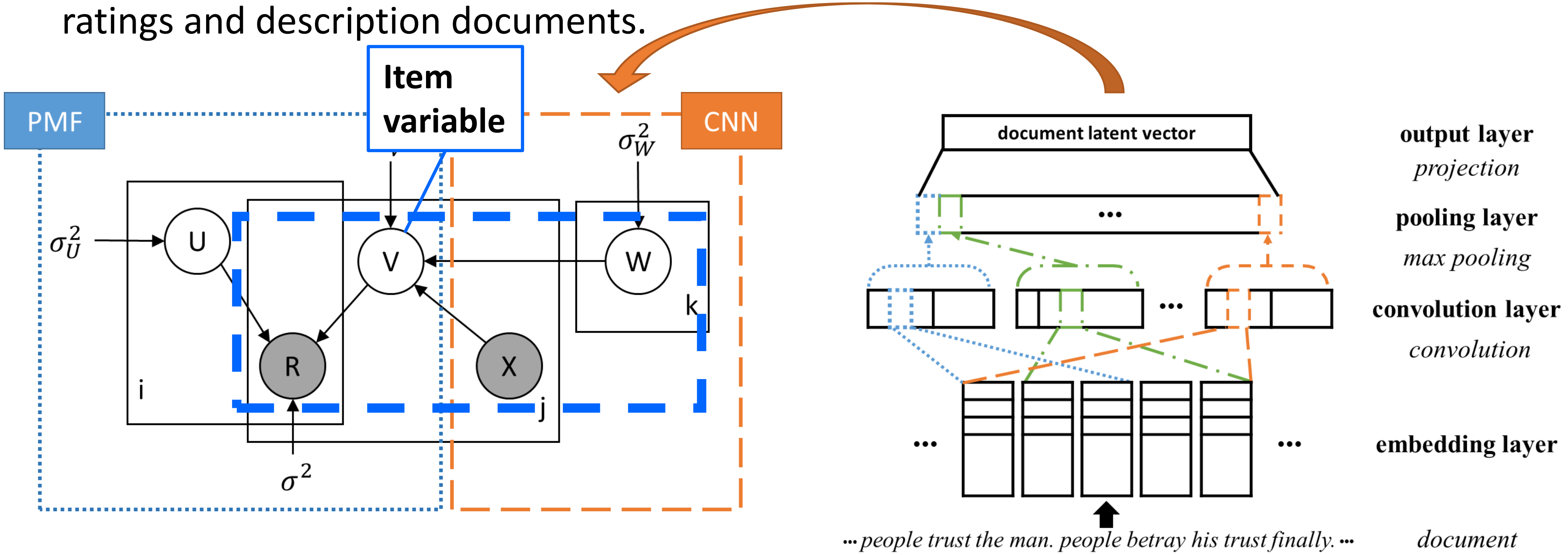
Graphical model of ConvMF

- Overview of ConvMF
  - We integrate CNN into PMF for the recommendation task.



Key of connection – Item variable

- Overview of ConvMF
  - **Item variable** plays a role of **the connection between PMF and CNN** in order to exploit ratings and description documents.



Optimization Methodology

- Use **maximum a posteriori** to solve U, V and W
  - $\max_{U,V,W} p(U,V,W|R,X,\sigma^2,\ \sigma_U^2,\sigma_V^2,\sigma_W^2) =$
    $\max_{U,V,W} p(R|U,V,\sigma^2)p(U|\sigma_U^2)p(V|W,X,\sigma_V^2)p(W|\sigma_W^2)$
  - By taking negative logarithm,

$$\mathcal{L}(U,V,W) = \sum_i^N \sum_j^M \frac{I_{ij}}{2}(r_{ij} - u_i^T v_j)_2 + \frac{\lambda_U}{2}\sum_i^N \|u_i\|_2$$

$$+ \frac{\lambda_V}{2}\sum_j^M \|v_j - cnn(W,X_j)\|_2 + \frac{\lambda_W}{2}\sum_k^{|w_k|}\|w_k\|_2,$$

  - Use coordinate descent to update latent models per iteration

$$u_i \leftarrow (VI_iV^T + \lambda_U I_K)^{-1} V R_i$$
$$v_j \leftarrow (UI_jU^T + \lambda_V I_K)^{-1} \underline{(UR_j + \lambda_V cnn(W,X_j))}$$

λ_v balances between ratings and documents

Optimization Methodology

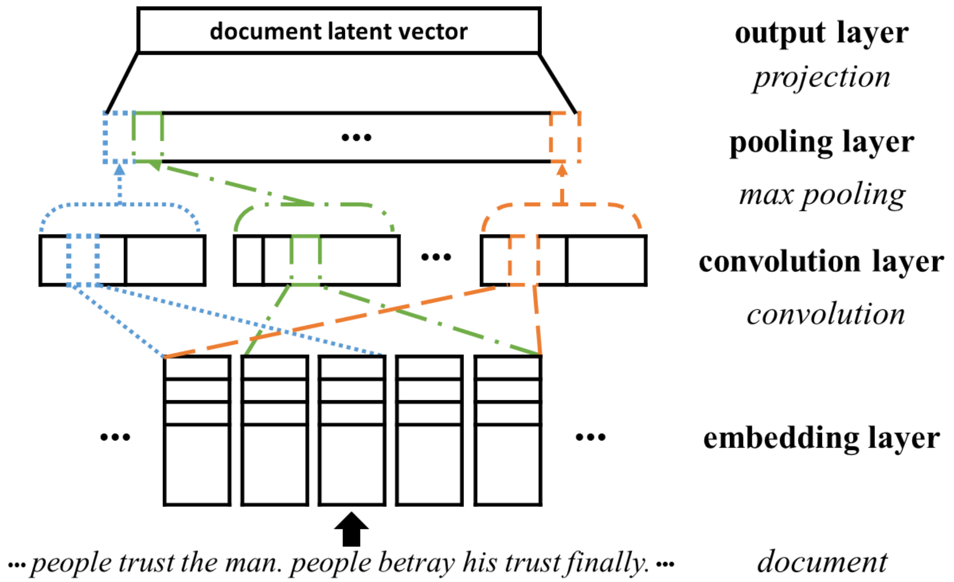
- However, W cannot be solved analytically as we can do for U and V.

Optimization Methodology

- However, W cannot be solved analytically as we can do for U and V.
- Fortunately, when U, V are temporarily fixed,
  loss function $\mathcal{L}$ becomes an error function with regularized terms of neural net.

$$\mathcal{E}(W) = \frac{\lambda_V}{2} \sum_j^M \|(v_j - cnn(W, X_j))\|^2 + \frac{\lambda_W}{2} \sum_k^{|w_k|} \|w_k\|^2 + \text{constant}$$

- To optimize W, we use **backpropagation algorithm with given target value v_j.**

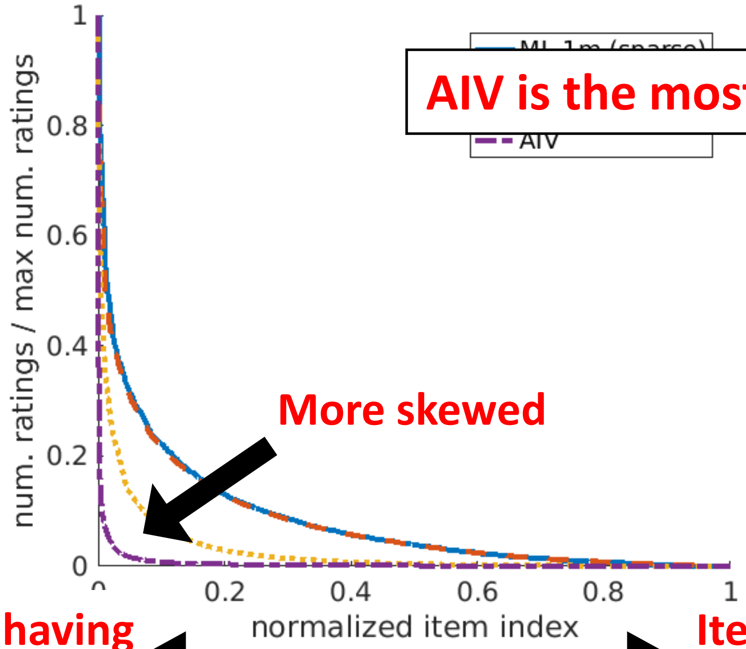


Explicit feedback datasets (range from 1 to 5)

Dataset	# users	# items	# ratings	density	documents
MovieLens-1m (ML-1m)	6,040	3,544	993,482	**4.641%**	IMDB
MovieLens-10m (ML-10m)	69,878	10,073	9,945,875	**1.413%**	IMDB
Amazon Instant Video (AIV)	29,757	15,149	135,188	**0.030%**	Amazon Review

AIV is the most skewed and sparse dataset!

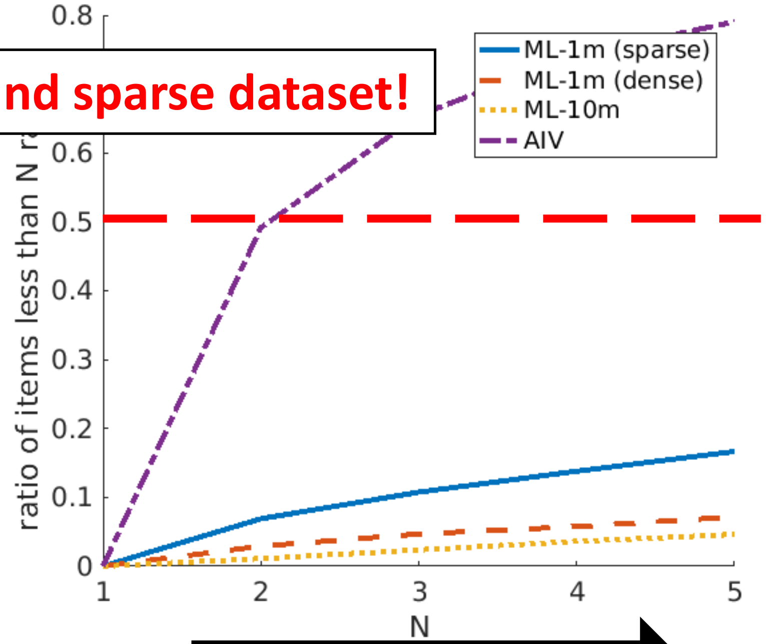
∝ num. ratings



More skewed

Item having more ratings

Item having less ratings



In AIV, 50% of items have only one rating!

num. ratings on item

Experiment Setting

- Competitor
  - *PMF* [NIPS`08] – conventional MF
  - *CTR* [KDD`11] – the state-of-the-art LDA-integrated recommendation
  - *CDL* [KDD`15] – the state-of-the-art SDAE-integrated recommendation
  - *ConvMF* – **our proposed model**
  - *ConvMF+* – **our proposed model with the pre-trained word embedding model (Glove)**

- Measure
  - Follow the convention in recommender system.

$$\text{RMSE} = \sqrt{\frac{\sum_{i,j}^{N,M} (r_{ij} - \hat{r}_{ij})^2}{\#\text{ of ratings}}}$$

Overall performance comparison

- RMSE – training / valid / test dataset (80% / 10% / 10%)

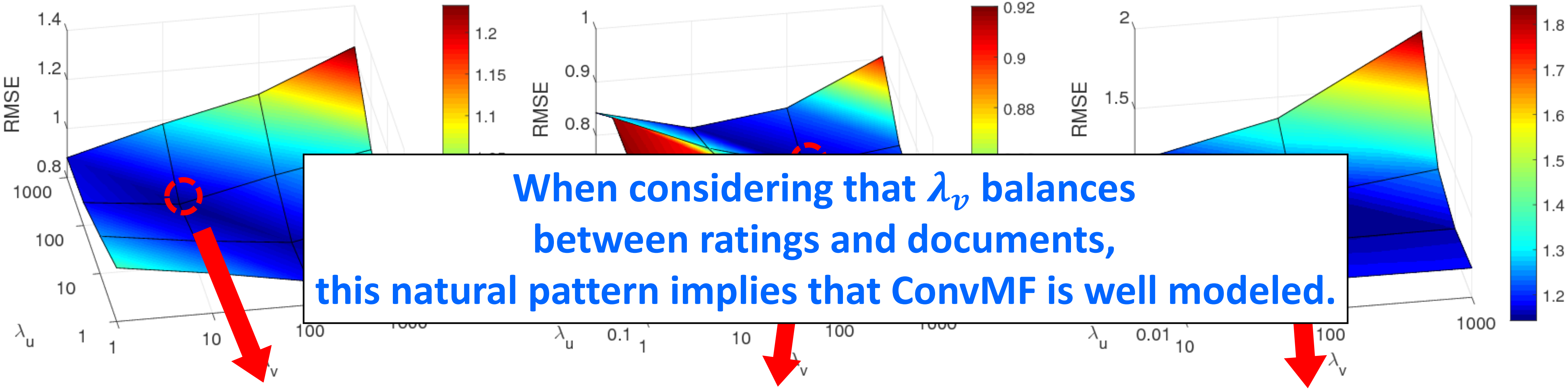
ConvMF and ConvMF+ achieve significant improvements on all the datasets.

Model			V
PMF	0.8971 (0.0020)	0.8311 (0.0010)	1.4118 (0.0105)
CTR	0.8969 (0.0027)	0.8275 (0.0004)	1.5496 (0.0104)
CDL	0.8879 (0.0015)	0.8186 (0.0005)	1.3594 (0.0139)
ConvMF	**0.8531** (0.0018)	**0.7958** (0.0006)	**1.1337** (0.0043)
ConvMF+	**0.8549** (0.0018)	**0.7930** (0.0006)	**1.1279** (0.0073)
Improve	**3.92%**	**2.79%**	**16.60%**

Improvement by pre-trained word embedding

extremely sparse dataset!

Best performing parameter analysis – λ_u and λ_v



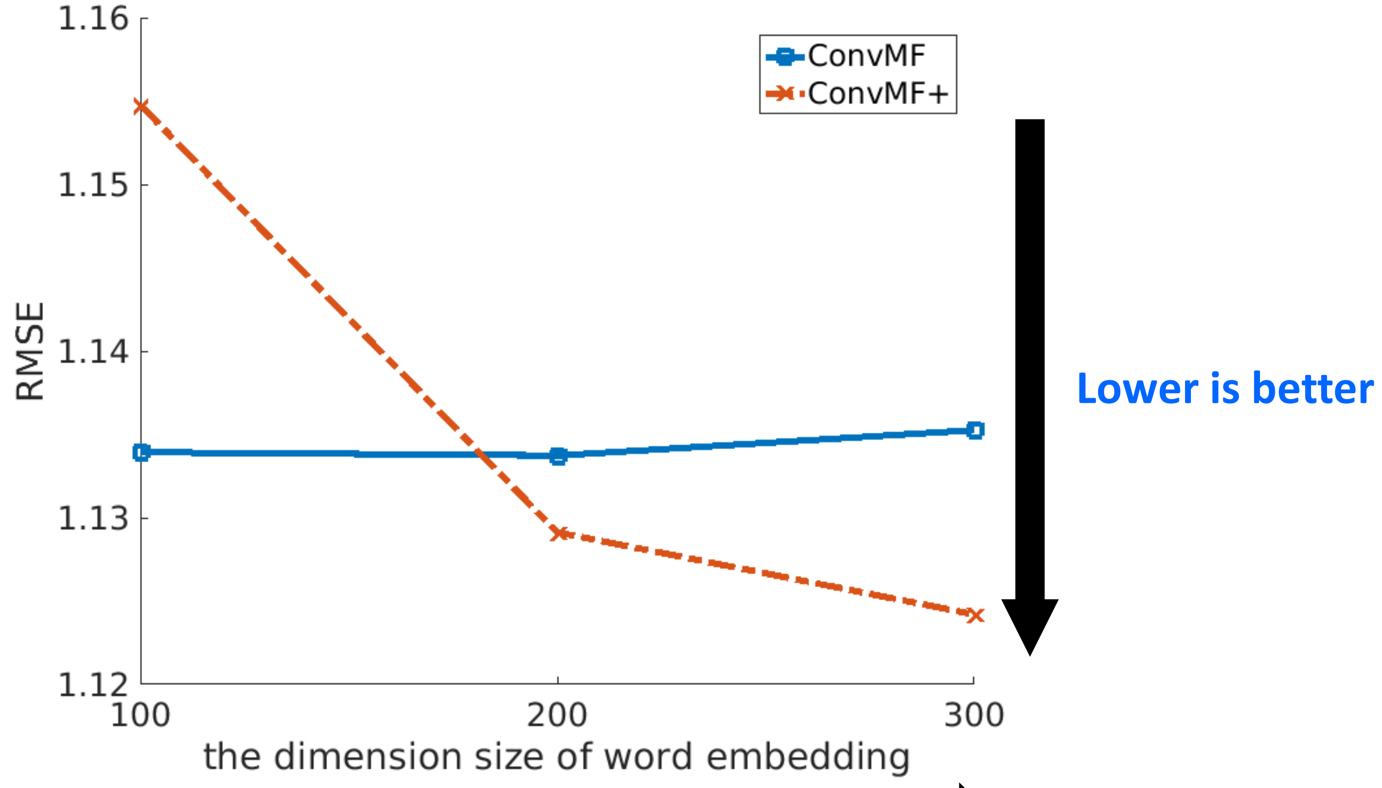
When considering that λ_v balances between ratings and documents, this natural pattern implies that ConvMF is well modeled.

	MovieLens-1m	MovieLens-10m	Amazon Instant Video
λ_u	100	10	1
λ_v	10	100	100

More skewed and sparse dataset

Impact of pre-trained word embedding model

- On AIV dataset



Information contained in the model gets richer

Case study of subtle contextual differences

The only max feature value affects the performance of ConvMF.
➔ A higher value has more chance to affect the performance!

	Phrase captured by W_c^{11}	$\max(c^{11})$	Phrase captured by W_c^{86}	$\max(c^{86})$	
as a verb ➡	people **trust** the man	**0.0704**	betray his **trust** finally	**0.1009**	⬅ as a noun
	Test phrases for W_c^{11}	$\max(c_{test}^{11})$	Test phrases for W_c^{86}	$\max(c_{test}^{86})$	
as a verb ➡	people **believe** the man	**0.0391**	betray his **believe** finally	0.0682	⬅ as a verb
as a noun ➡	people **faith** the man	0.0374	betray his **faith** finally	**0.0693**	⬅ as a noun
irrelevant ➡	people **tomas** the man	0.0054	betray his **tomas** finally	0.0480	⬅ irrelevant

W_c^{11} is more likely to capture "**trust**" as a **verb** W_c^{86} is more likely to capture "**trust**" as a **noun**

ConvMF distinguishes a subtle contextual difference of the term "trust"

Conclusion

- We demonstrate that considering **contextual information** provides **a deeper understanding** of description documents

- We develop a novel document context-aware recommendation model, ConvMF, that **seamlessly integrates CNN into PMF** in order to capture contextual information for the rating prediction

- Since ConvMF is based on PMF, ConvMF is able to be **extended to combining other MF-based recommendation models** such as SVD++

Thank you

- ConvMF webpage
  - http://dm.postech.ac.kr/ConvMF
- Any question?

Reference

- [KDD`15] Collaborative deep learning for recommender systems
- [RecSys`14] Ratings meet reviews, a combined approach to recommend
- [RecSys`13] Hidden factors and hidden topics: Understanding rating dimensions with review text
- [IJCAI`13] Hierarchical Bayesian matrix factorization with side information
- [NIPS`13] Deep content-based music recommendation
- [ICML`12] Collaborative topic regression with social matrix factorization for recommendation systems
- [KDD`11] Collaborative topic modeling for recommending scientific articles
- [JMLR`11] Natural language processing (almost) from scratch
- [ACL`14] A convolutional neural network for modelling sentences
- [EMNLP`14] Convolutional neural networks for sentence classification
- [EMNLP`14] Modeling interestingness with deep neural networks
- [CIKM`14] A latent semantic model with convolutional-pooling structure for information retrieval