Neural Representations in Hybrid Recommender Systems: Prediction versus Regularization

Ramin Raziperchikolaei, Tianyu Li, and Young-joo Chung

Rakuten Institute of Technology

Rakuten, Inc.

Rakuten

Recommender systems

Recommender systems take a user and an item as the inputs and returns the estimated rating.

	Item 1	Item 2	Item 3	. . .	Item n
User 1	1	?	?	?	?
User 2	?	5	?	?	?
User 3	?	?	?	2	?
…	?	?	?	?	?
User m	?	2	?	?	3

There are m users and n items.

The sparse rating matrix is m by n, denoted by $\mathbf{R} \in \mathbb{R}^{m \times n}$

$R_{jk} > 0$ is the rating of the user j on the item k, and $R_{jk} = 0$ means the rating is unknown.

The explicit ratings are integers between 1 (not recommended) and 5 (highly recommended).

Goal: predict the unknown ratings.

Hybrid recommender systems

Hybrid recommender systems use user and item ratings and content information (age, occupation, title, reviews, etc.) to predict the ratings.

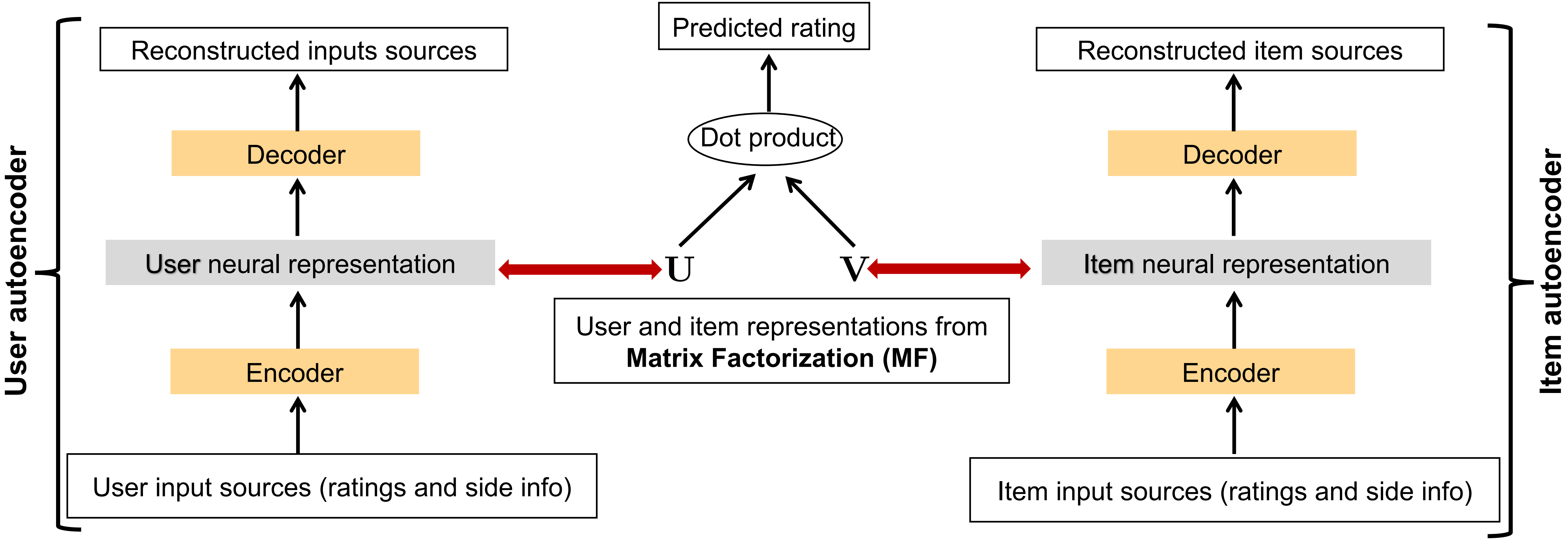
Neural network-based methods have been used to learn better representations.

The most widely-used neural network structure in recommender systems has been (denoising) autoencoders:

- Sedhain et al., AutoRec: Autoencoders meet collaborative filtering, World Wide Web 2015.
- Li et al., Deep collaborative filtering via marginalized denoising autoencoder, CIKM 2015.
- Strub et al., Hybrid recommender system based on autoencoders, DLRS 2016.
- Dong et al., A hybrid collaborative filtering model with deep structure for recommender systems, AAAI 2017
- Zhang et al, AutoSVD++:an efficient hybrid collaborative filtering model via contractive autoencoders, SIGIR 2017
- Li et al., Deep heterogeneous autoencoders for collaborative filtering, ICDM 2018

Existing systems still use representations learned by matrix factorization (MF) to predict the rating, while using representations learned by neural networks as the regularizer.

Previous autoencoder-based methods



Autoencoder-based hybrid recommender systems

The idea is to predict the ratings and reconstruct the user/item sources of information simultaneously.

$\mathbf{X}$ contains users' side information, such as age, occupation, location, etc.

$\mathbf{Y}$ contains users' side information, such as price, title, reviews, etc.

Reconstruction terms

MF term

$$\min_{\mathbf{U},\mathbf{V},\boldsymbol{\theta}} L(\mathbf{f}^u(\mathbf{g}^u(\mathbf{R},\mathbf{X}))) + L(\mathbf{f}^i(\mathbf{g}^i(\mathbf{R},\mathbf{Y}))) + \lambda_1 \sum_{j,k} \mathbb{1}(R_{jk} > 0)||R_{jk} - \mathbf{U}_{j,:}\mathbf{V}_{k,:}^T||^2 +$$
$$\lambda_2||\mathbf{U} - \mathbf{g}^u(\mathbf{R},\mathbf{X})||^2 + \lambda_3||\mathbf{V} - \mathbf{g}^i(\mathbf{R},\mathbf{Y})||^2 + \text{reg. terms},$$

Regularization terms

The MF term is used for prediction and the neural representations are used for the regularization purpose.

The hyper-parameters λ_2 and λ_3 determine how close the two representations should be from each other.

Issues: lack of motivation behind using neural representations as the regularizer and slow optimization.

Our approach: neural representations as the predictors

In our approach, the encoders' outputs are the only user/item representations in our model:

$$\min_{\boldsymbol{\theta}} L(\mathbf{f}^u(\mathbf{g}^u(\mathbf{R}, \mathbf{X}))) + L(\mathbf{f}^i(\mathbf{g}^i(\mathbf{R}, \mathbf{Y})))+$$
$$\lambda_1 \sum_{j,k} \mathbb{1}(R_{jk} > 0)||R_{jk} - \mathbf{g}^u(\mathbf{R}_{j,:}, \mathbf{X}_{j,:})^T \mathbf{g}^i(\mathbf{R}_{:,k}, \mathbf{Y}_{k,:})||^2$$

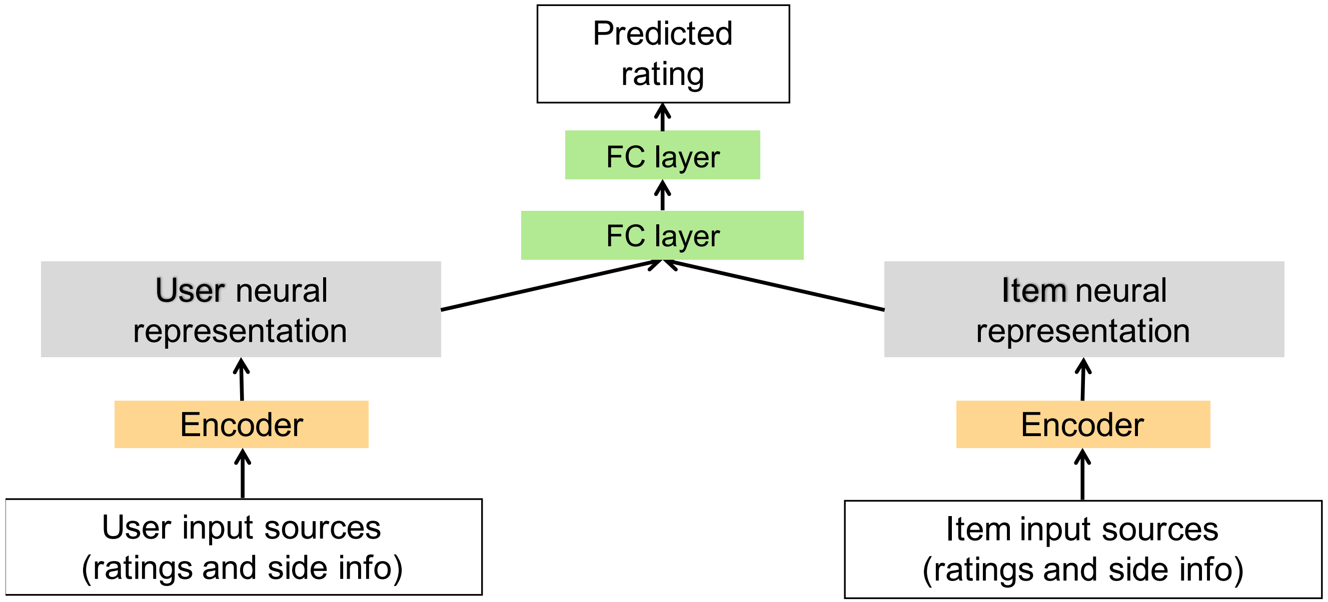
Advantages:

1) The hyper-parameters λ_2 and λ_3 are removed, which leads to less time in hyper-parameter tuning.
2) The number of parameters decreased as we removed $\mathbf{U}$ and $\mathbf{V}$.
3) The network can be trained end-to-end.

Our approach: neural representations with a direct structure

The direct structure is achieved by making two modifications to our autoencoder structure:

1) We remove the decoders from the structure, which leads to saving around 50% of memory and faster optimization.
2) We use a set of fully connected layers to predict the final rating, instead of the dot product.



Experiments

We use four datasets in our experiments.

Table 1: Summary of the four datasets.

Dataset	# of users	# of items	sparsity
ml100k	1 000	1 600	94%
ml1m	6 000	4 000	96%
Amazon R.	86 400	108 500	99.994%
Ichiba	324 000	294 000	99.84%

We use user/item contents, in addition to the ratings, as the sources of information.

We report RMSE and precision:

$$\mathrm{RMSE} = \sqrt{\frac{1}{|T|} \sum_{R_{jk} \in T} (R_{jk} - \hat{R}_{jk})^2}. \qquad \mathrm{precision} = \frac{|\text{relevant items}| \cap |\text{retrieved items}|}{|\text{retrieved items}|}$$

Neural representation: prediction vs regularization

We compare our proposed NRP framework, trained with the autoencoder and direct structures, versus MF and autoencoder-based methods on ml100k and ml1m datasets.

	ml1m dataset			
method	RMSE	precision	# params.	time
MF	0.892 ± 0.004	68.2% ± 0.3	(0, 1M)	45s
DHA	0.865 ± 0.001	69.3% ± 0.2	(44M, 1M)	1 097s
$\mathbf{NRP_{DHA}}$	0.855 ± 0.002	69.6% ± 0.2	(44M, 0)	1 027s
aSDAE	0.879 ± 0.005	69.0% ± 0.1	(66M, 1M)	1 155s
$\mathbf{NRP_{aSDAE}}$	0.877 ± 0.008	68.5% ± 0.4	(66M, 0)	1 055s
$\mathbf{NRP_{direct}}$	**0.851 ± 0.001**	**70.0% ± 0.1**	(22M, 0)	640s

Our framework combined with the direct structure achieves the best prediction results, fastest training, and minimum memory usage compared to the autoencoder-based methods.

Neural representations are better for prediction than regularization.

Comparison with the hybrid and collaborative filtering methods

We compare RMSE and precision of our method with several SOTA methods.

method	ml100k	ml1m	Amazon	Ichiba
MF [Koren et al. 2009]	0.940	0.892	1.153	1.00
Autorec [Sedhain et al. 2015]	0.921	0.889	2.19	2.47
NeuMF [He et al. 2017]	0.948	0.886	1.140	0.900
DSSM [Huang et al. 2013	0.934	0.941	NA	0.913
DHA [Li et al. 2018]	0.939	0.865	OM	OM
$\mathrm{NRP}_{\mathrm{DHA}}$	0.926	0.855	1.135	OM
aSDAE [Dong et al. 2017]	0.946	0.879	OM	OM
$\mathrm{NRP}_{\mathrm{aSDAE}}$	0.910	0.877	1.24	OM
HIRE [Liu et al. 2019]	0.930	0.861	OM	OM
$\mathrm{NRP}_{\mathrm{direct}}$	**0.897**	**0.851**	**1.135**	**0.889**

	ml1m		Amazon review	
method	top 10%	top 25%	top 10%	top 25%
MF	55.6%	68.05%	64.9%	71.5%
Autorec	57.6%	69.5%	62.6%	69.8%
NeuMF	56.8%	68.9%	66.8%	72.6%
DSSM	54.7%	67.2%	NA	NA
DHA	57.4%	69.3%	OM	OM
$\mathrm{NRP}_{\mathrm{DHA}}$	57.3%	69.5%	66.6%	72.9%
aSDAE	56.4%	68.0%	OM	OM
$\mathrm{NRP}_{\mathrm{aSDAE}}$	57.1%	69.0%	64.5%	71.2%
HIRE	57.4%	69.4%	OM	OM
$\mathrm{NRP}_{\mathrm{direct}}$	**58.1%**	**69.9%**	**67.3%**	**73.1%**

Our method achieves the best results on different datasets

Conclusion

The current autoencoder-based hybrid recommender systems learn two types of representations:

- One comes from the matrix factorization and used for prediction.
- The other one comes from neural networks and used for regularization.

We proposed a framework that uses the neural networks' representation directly for the prediction task.

We have shown that by applying our approach to the same autoencoder structure as previous works, we achieve faster training and better performance.

We also proposed a simpler network structure by removing the decoders and replacing dot product with MLP in autoencoders.

Our approach combined with the new proposed structure outperforms the previous works.

It also has a fast training and small memory usage compared to the autoencoder-based methods.

Rakuten