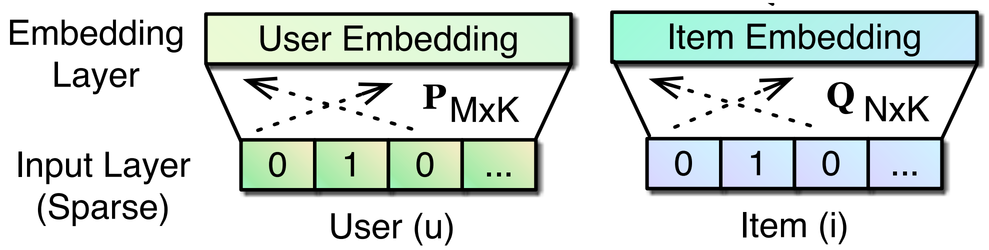
Outer Product-based Neural Collaborative Filtering

Xiangnan He[1], Xiaoyu Du[2], Xiang Wang[1], Feng Tian[3], Jinhui Tang[4] and Tat-Seng Chua[1]

1: National University of Singapore
2: Chengdu University of Information Technology
3: Northeast Petroleum University
4: Nanjing University of Science and Technology

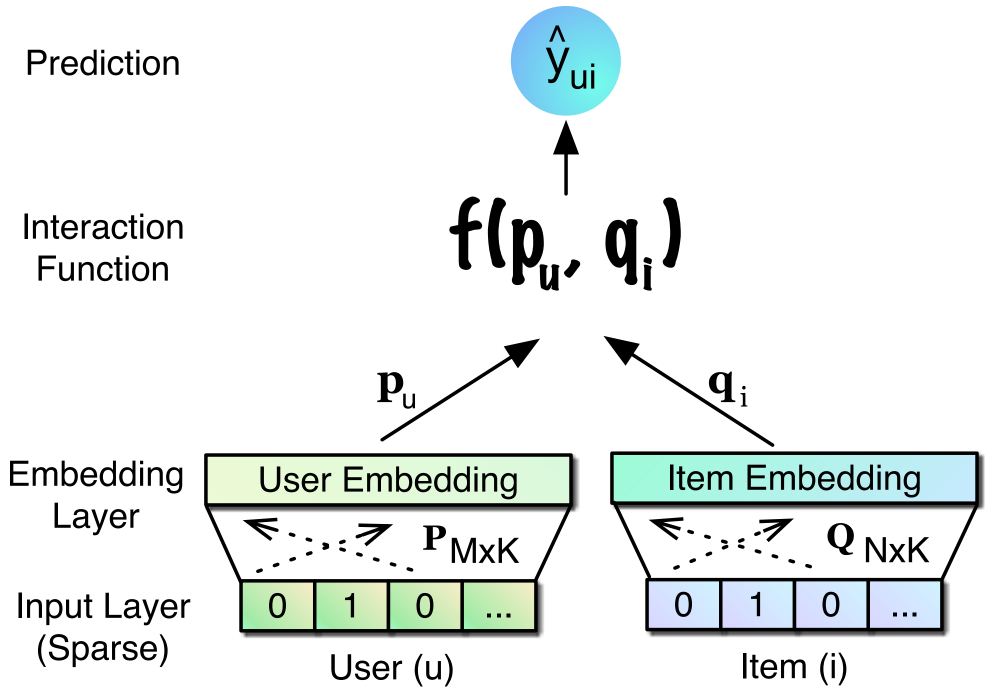
Matrix Factorization (MF)

- A prevalent model for collaborative filtering
  - Represent a user (or an item) as a vector of latent factors (also termed as *embedding*)



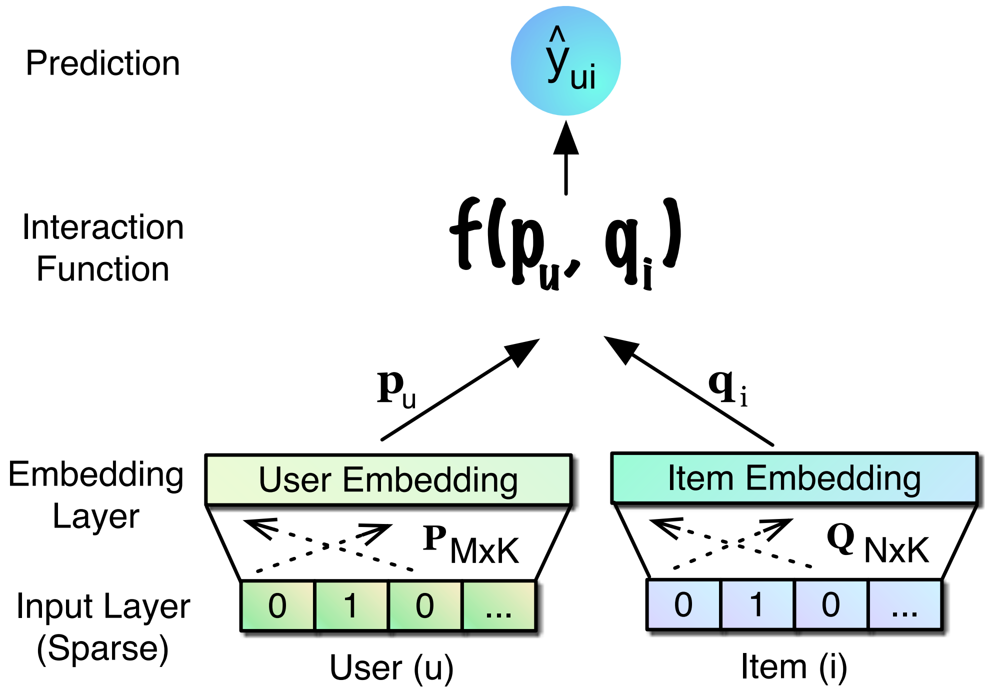
Matrix Factorization (MF)

- A prevalent model for collaborative filtering
  - Represent a user (or an item) as a vector of latent factors (also termed as *embedding*)
  - Estimate an interaction as the inner product between the user embedding and item embedding



Matrix Factorization (MF)

- A prevalent model for collaborative filtering
  - Represent a user (or an item) as a vector of latent factors (also termed as *embedding*)
  - Estimate an interaction as the inner product between the user embedding and item embedding
- Many extensions on MF
  - Model perspective: NeuMF [He et al, WWW'17], Factorization Machine etc.
  - Learning perspective: BPR, Adversarial Personalized Ranking [He et al, SIGIR'18]



Interaction Function in MF

- MF uses **Inner Product** as the interaction function
- The implicit assumption in Inner Product:
  - The embedding dimensions are independent with each other

$$f(\mathbf{p}_u, \mathbf{q}_i) = \mathbf{p}_u^T \mathbf{q}_i = \sum \begin{pmatrix} p_{u1} q_{i1}, \\ p_{u2} q_{i2}, \\ \cdots \\ p_{uk} q_{ik} \end{pmatrix}$$

However, the implicit assumption is impractical.

- ☐ The embedding dimensions could be interpreted as certain properties of items [Zhang *et al.*, SIGIR'14], which are not necessarily to be independent

Recent DNN-based models either use element-wise product or concatenation.

- ☐ E.g., NeuMF [He et al, WWW'17], NNCF [Bai et al, CIKM'17], JRL [Zhang et al, CIKM'17], Autoencoder-based CF Models [Wu et al, WSDM'16]
- ☐ Still, the relations among embedding dimensions are not explicitly modeled.

Research Questions

- How to model the relations between embedding dimensions?

- Next:our proposed method:
  1. Outer product on user&item embedding for pairwise interaction modeling
  2. CNN on the outer product matrix to extract and reweight prediction signals.

Outer Product-based Neural Collaborative Filtering (ONCF)

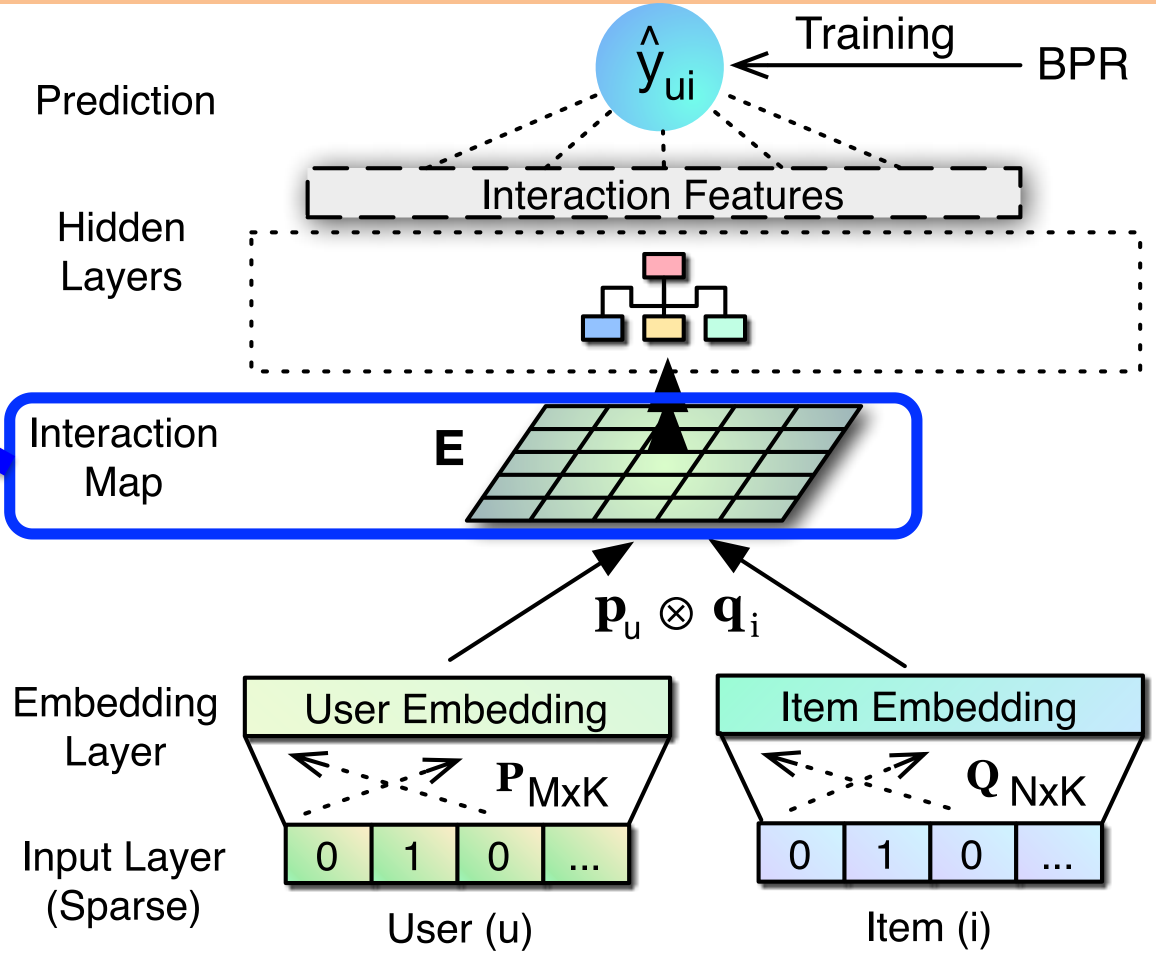
Outer-product **explicitly** models the **pairwise relations** between embedding dimensions:

- Get a 2D matrix, named as **interaction map:**

Signals in Inner product

$$\mathbf{E} = \mathbf{p}_u \otimes \mathbf{q}_i = \mathbf{p}_u \mathbf{q}_i^T = \begin{bmatrix} p_{u1}q_{i1} & p_{u1}q_{i2} & \cdots & p_{u1}q_{ik} \\ p_{u2}q_{i1} & p_{u2}q_{i2} & \cdots & p_{u2}q_{ik} \\ \vdots & \vdots & \ddots & \vdots \\ p_{uk}q_{i1} & p_{uk}q_{i2} & \cdots & p_{uk}q_{ik} \end{bmatrix}$$

Indicating the interaction between the k-th dimension of $\mathbf{p}_u$ and the 2-nd dimension of $\mathbf{q}_i$.

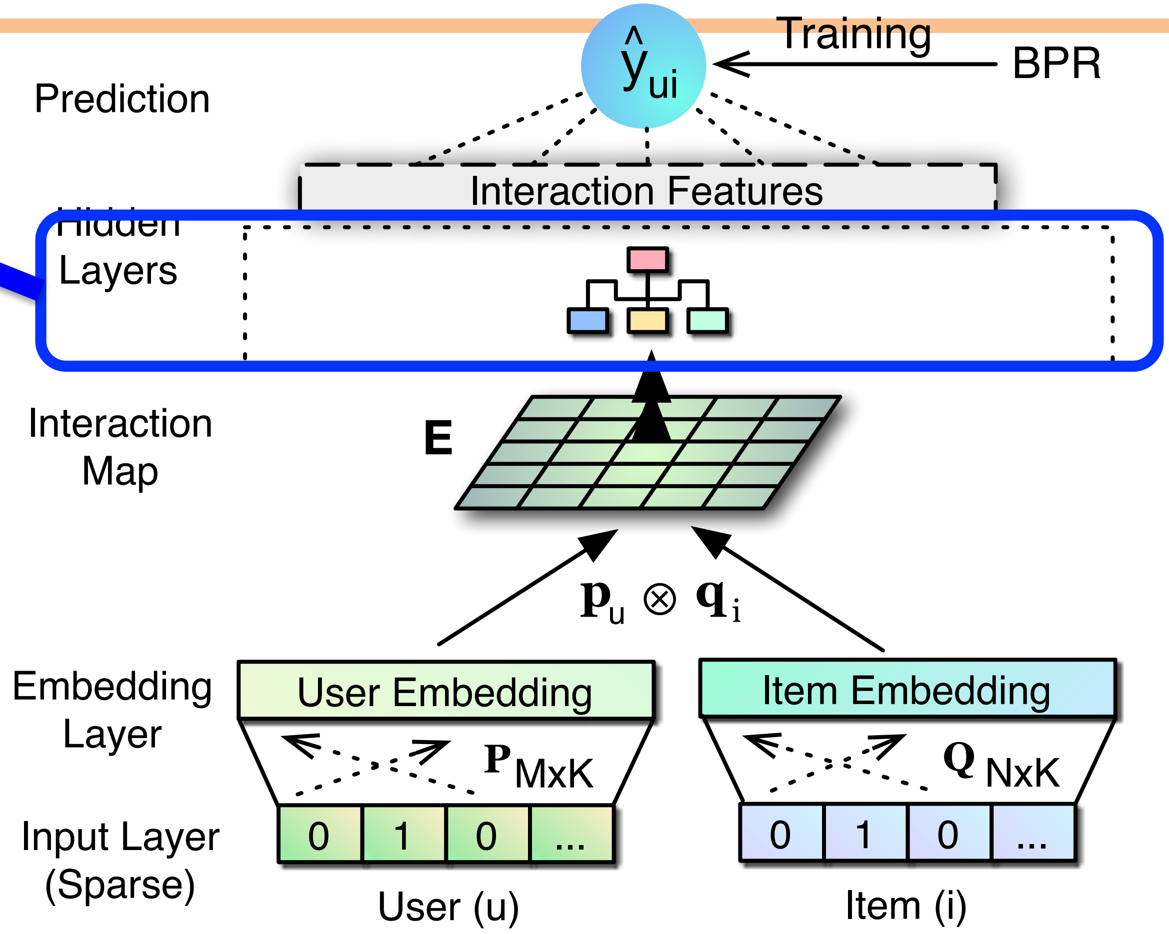


ONCF-MLP

Above the interaction map are hidden layers, which aim to extract useful signal from the 2D interaction map.

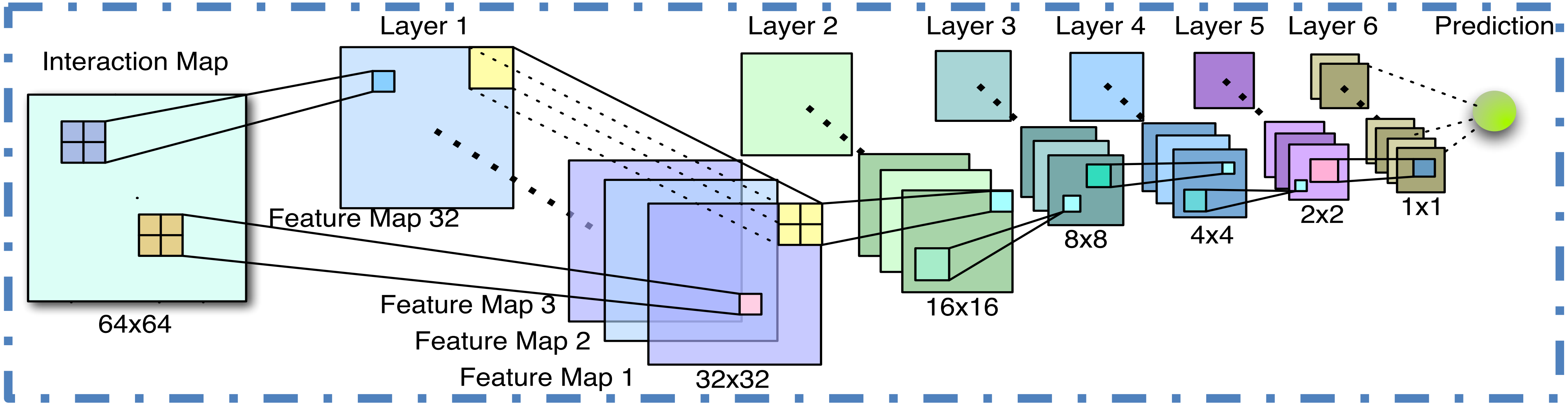
A straightforward solution is to use MLP, however it results in too many parameters:

- Interaction map **E** has K × K neurons (K is embeddings size usually hundreds)
- Require **large memories** to store the model
- Require **large training data** to learn the model well



Convolutional NCF (ConvNCF)

- ConvNCF uses locally connected CNN as hidden layers in ONCF:
  - CNN has much fewer parameters than MLP
  - Hierarchical tower structure: higher layer integrates more information from larger area.
  - Final prediction summarizes all information from interaction map.



- 2 Fully Connected Layers: **> 10M parameters**
- **6 Convolutional Layers: 20K parameters, but achieve better performance!**

Experimental Settings

➢ Datasets

– Yelp: 25,815 users, 25,677 items, and 730,791 interactions.

– Gowalla: 54,156 users, 52,400 items, and 1,249,703 interactions.

➢ Protocols

– Leave-one-out: holdout the latest interaction of each user as the test

– Pair 1 testing instance with 999 negative instances

– Top-K evaluation: ranking 1 positive vs. 999 negatives.

– Ranking lists are evaluated by Hit Ratio and NDCG (@10).

➢ Loss Function

– Bayesian Personalized Ranking

Baselines

- MF-BPR [Rendle *et al.*, UAI'09]
  - Learning MF with a pair-wise classification loss.
- MLP [He *et al.*, WWW'17]
  - 3-layer multi-layer perceptron above user and item embeddings.
- JRL [Zhang *et al.*, CIKM'17]
  - Multi-layer perceptron above the element-wise product of embeddings.
- NeuMF [He *et al.*, WWW'17]
  - A neural network combining hidden layer of MF and MLP.

Performance Comparison

	Gowalla		Yelp		Average Improvement of ConvNCF over Baselines
	HR@10	NDCG@10	HR@10	NDCG@10	
MF-BPR	0.7480	0.5214	0.2817	0.1447	+9.5%
MLP	0.7590	0.5202	0.2831	0.1446	+9.1%
JRL	0.7747	0.5615	0.2922	0.1519	+4.3%
NeuMF	0.7793	0.5660	0.2958	0.1536	+3.3%
ConvNCF	**0.7936***	**0.5826***	**0.3086***	**0.1600***	-

∗ indicates that the improvements over all other methods are statistically significant for $p < 0.05$.

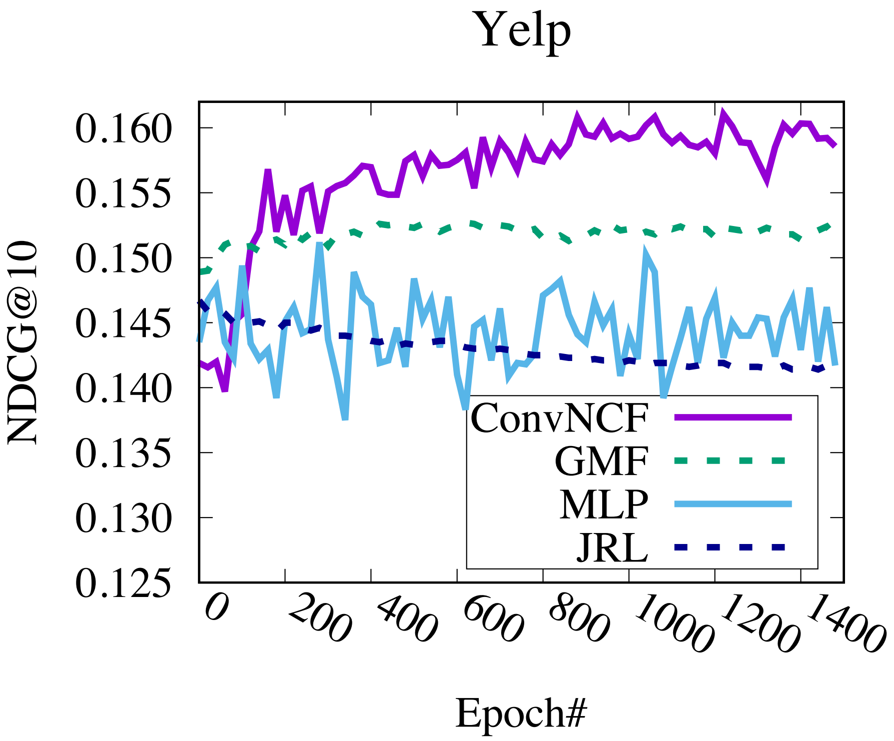
Overall Performance: ConvNCF > NeuMF[He et al., 2017] **> JRL**[Zhang et al., 2017]

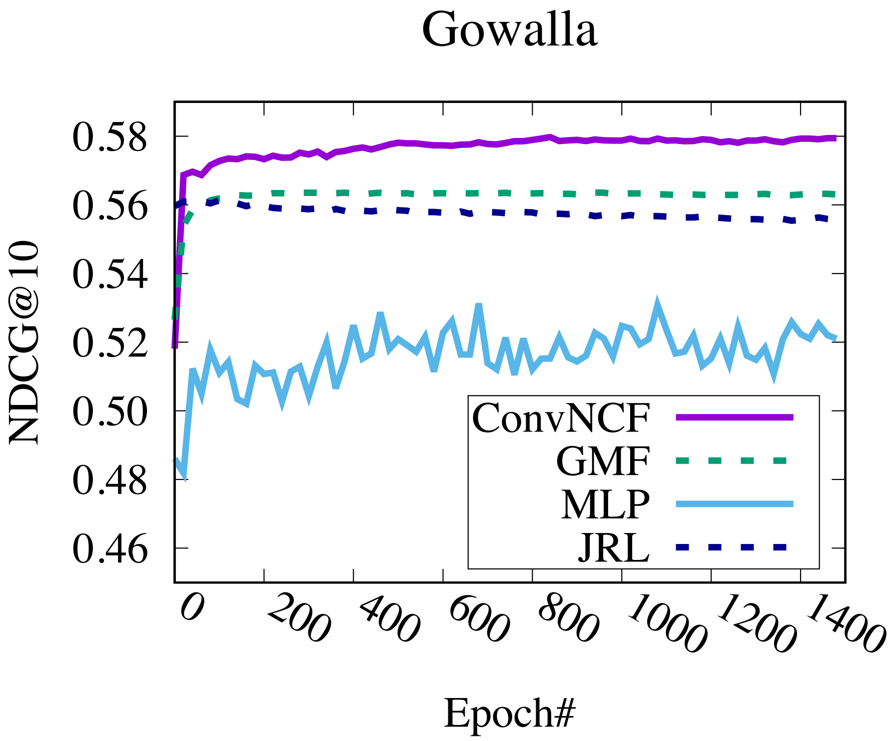
- Usefulness of modeling the relations of embedding dimensions
- Training MLP well is practically difficult.

Efficacy of Outer Product

Training process of neural models that apply different operations above the embedding layer:

- ConvNCF: outer product; GMF: element-wise product; MLP: concatenation; JRL: element-wise product



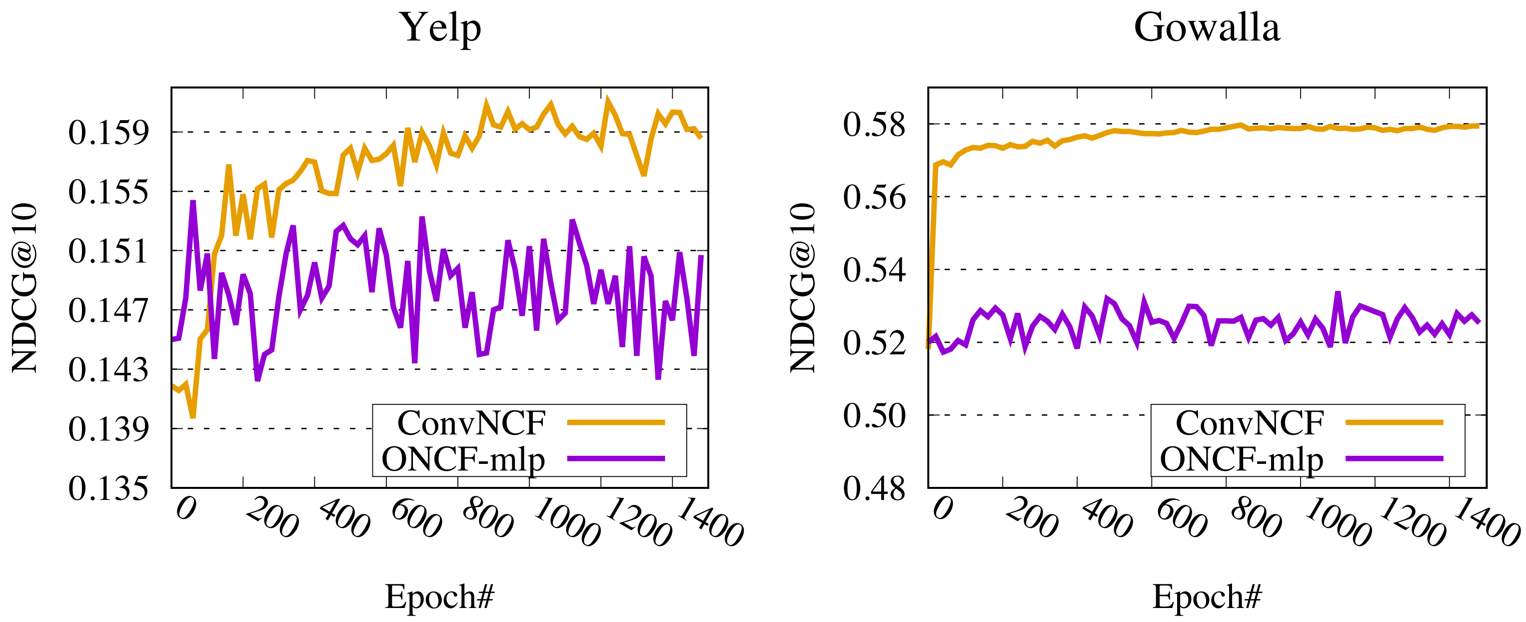


Outer product is a simple but effective merge of user&item embeddings.

Efficacy of CNN

NDCG@10 of using different hidden layers for ONCF:

- ConvNCF uses a 6-layer CNN.
- ONCF-mlp uses a 3-layer MLP above the interaction map.



1. ConvNCF outperforms ONCF-mlp.
2. ConvNCF is more stable than ONCF-mlp.

Conclusion & Future Work

Summary of contributions:

- A new neural framework for CF --- ONCF, which explicitly captures pairwise correlations between embedding dimensions with outer product
- A new model of ONCF framework --- ConvNCF, which uses CNN as hidden layers .
- Extensive experiments show effectives of ONCF framework and ConvNCF method.

Future work:

- We will explore more advanced CNN models to further explore the potentials of our ONCF framework.
- We will extend ONCF to content-based recommendation scenarios, e.g., items have image and textual content.

References

[Xue *et al.*, 2017] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. Deep matrix factorization models for recommender systems. In *IJCAI*, pages 3203–3209, 2017.

[Rendle *et al.*, 2009] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, pages 452–461, 2009.

[He *et al.*, 2018] Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. Adversarial personalized ranking for item recommendation. In *SIGIR*, 2018.

[Zhang *et al.*, 2014] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *SIGIR*, pages 83–92, 2014.

[Tay *et al.*, 2018] Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. Latent relational metric learning via memory-based attention for collaborative ranking. In *WWW*, pages 729– 739, 2018.

[Bai *et al.*, 2017] Ting Bai, Ji-Rong Wen, Jun Zhang, and Wayne Xin Zhao. A neural collaborative filtering model with interaction-based neighborhood. In *CIKM*, pages 1979–1982, 2017.

[He *et al.*, 2017] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Neural collaborative filtering. In *WWW*, pages 173–182, 2017.

[Zhang *et al.*, 2017] Yongfeng Zhang, Qingyao Ai, Xu Chen, and W Bruce Croft. Joint representation learning for top-n recommendation with heterogeneous information sources. In *CIKM*, pages 1449–1458, 2017.

THANK YOU!

Codes: https://github.com/duxy-me/ConvNCF

For questions: email to Dr. He Xiangnan
xiangnanhe@gmail.com