A^3NCF: An Adaptive Aspect Attention Model for Rating Prediction

Zhiyong Cheng[1], Ying Ding[2], Xiangnan He[1], Lei Zhu[3], Xuemeng Song[4], Mohan Kankanhalli[4]

1. National University of Singapore
2. Vipshop US Inc., USA
3. Shandong Normal University, China
4. Shandong University, China

19 July, IJCAI'18, Stockholm

MOTIVATION

- **Review-based recommendation**: review contains rich information about user preference and item features.

Traditional Models	Topic Models & Latent Factors	Joint Models of Aspects and Ratings	Deep Learning Models
Weight on Quality of ratings, RecSys'12	HFT, RecSys'13	JMARS, KDD'14	ConvMF, RecSys'16
User reviews as content, RecSys'13	RMR, RecSys'14	EFM, SIGIR'14	DeepCONN, WSDM'17
Aspect Weighting, WI'15	CMR, CIKM'14	TRCF, IJCAI'13	TransNet, RecSys'17
TriRank, CIKM'15	TopicMF, AAAI'14	SULM, KDD'17	D-Attn, RecSys'17
	JMARS, KDD'14		NARRE, WWW'18
	FLAME, WSDM'15		
	RBLT, IJCAI'16		
	ITLFM, TKDE'16		
	ALFM, WWW'18		

M. Chelliah &S. Sarkar, RecSys'17 tutorial

MOTIVATION

- **Limitation**: ignores the fact that "*a user may place different importance to the various aspects of different items*"

Traditional Models
- Weight on Quality of ratings, RecSys'12
- User reviews as content, RecSys'13
- Aspect Weighting, WI'15
- TriRank, CIKM'15

Topic Models & Latent Factors
- HFT, RecSys'13
- RMR, RecSys'14
- CMR, CIKM'14
- TopicMF, AAAI'14
- JMARS, KDD'14
- FLAME, WSDM'15
- RBLT, IJCAI'16
- ITLFM, TKDE'16
- ALFM, WWW'18

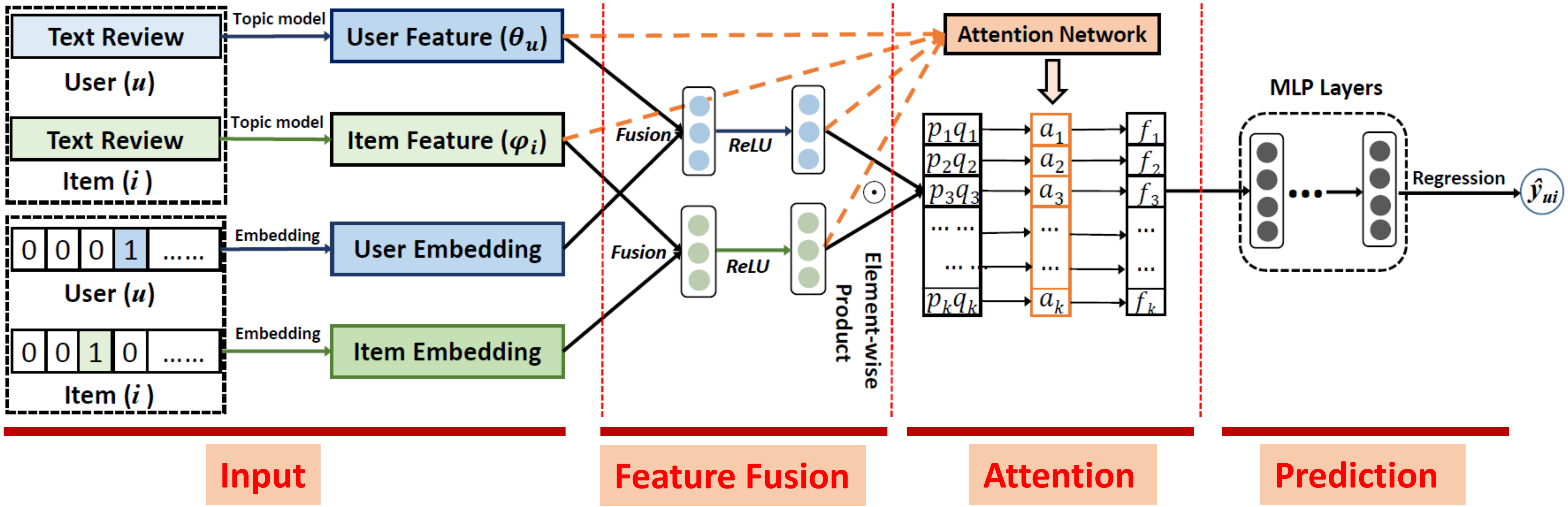
Joint Models of Aspects and Ratings
- JMARS, KDD'14
- EFM, SIGIR'14
- TRCF, IJCAI'13
- SULM, KDD'17

Deep Learning Models
- ConvMF, RecSys'16
- DeepCONN, WSDM'17
- TransNet, RecSys'17
- D-Attn, RecSys'17
- NARRE, WWW'18

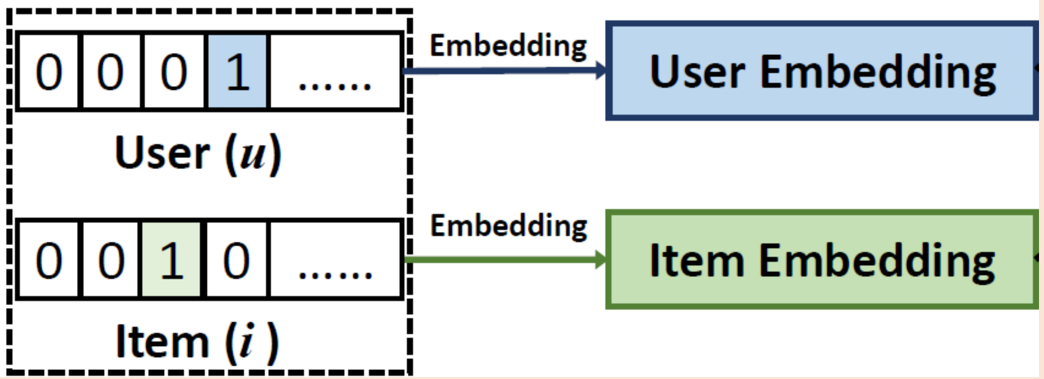
to

ider

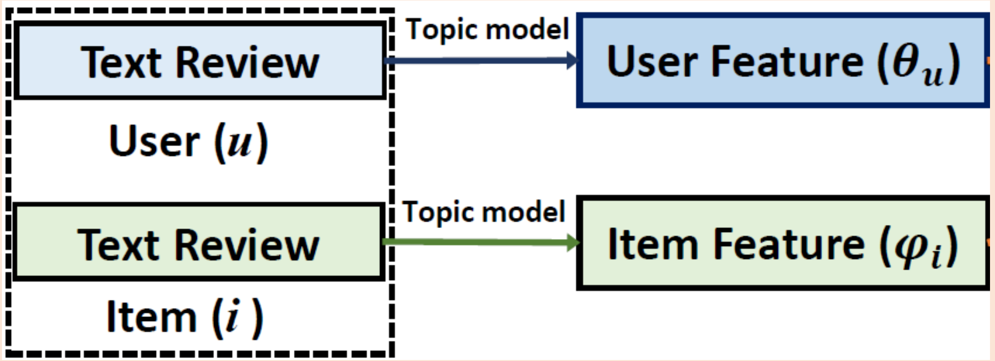
OUR MODEL - OVERVIEW



OUR MODEL – INPUT MODULE

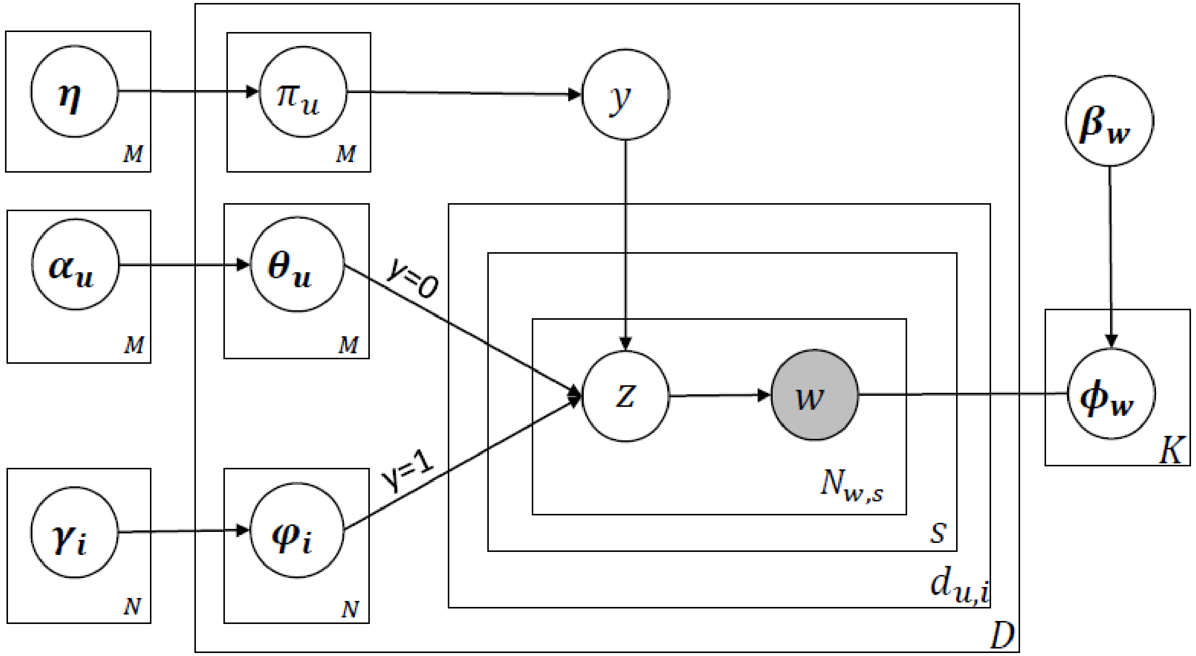


- User/item identity: binary one-hot encoding
- Embedding layer ->identity representation



- User/item features: from the user/item's review
- Topic model -> topic distribution as features

OUR MODEL – TOPIC MODEL

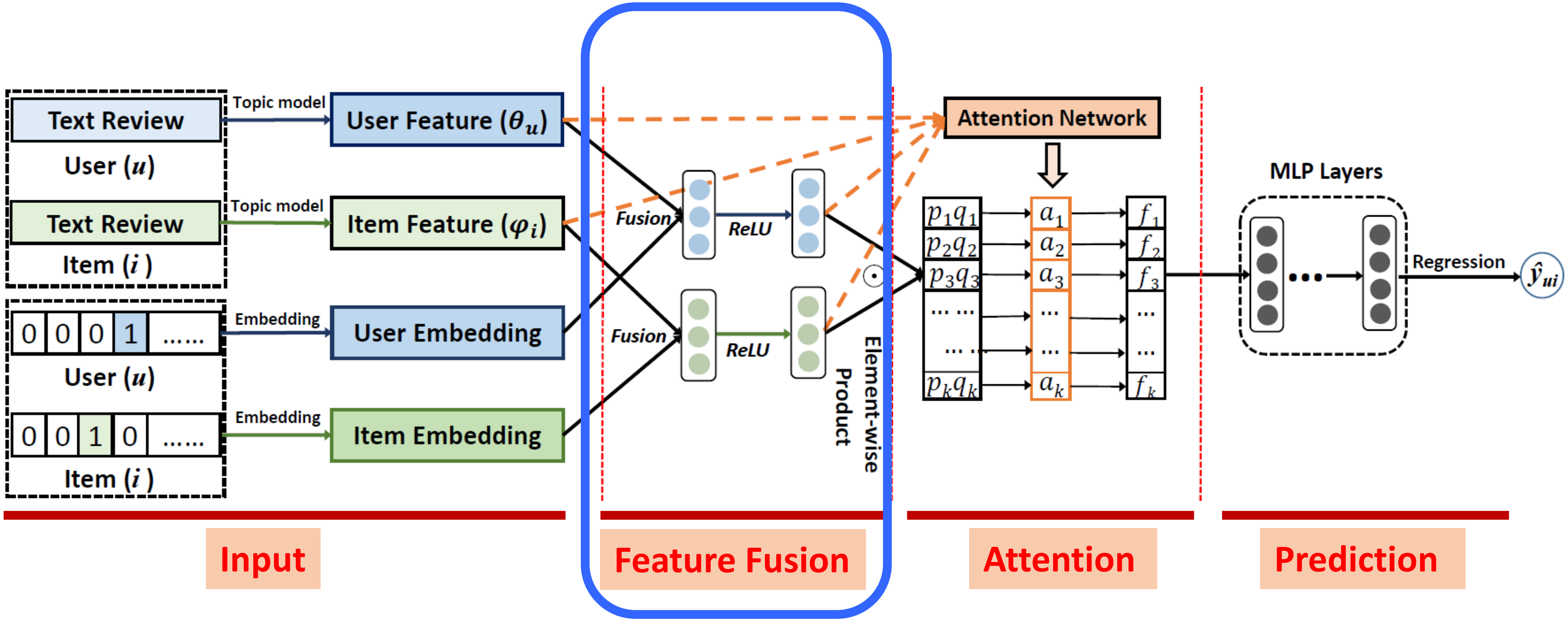


Graphical representation of the topic model

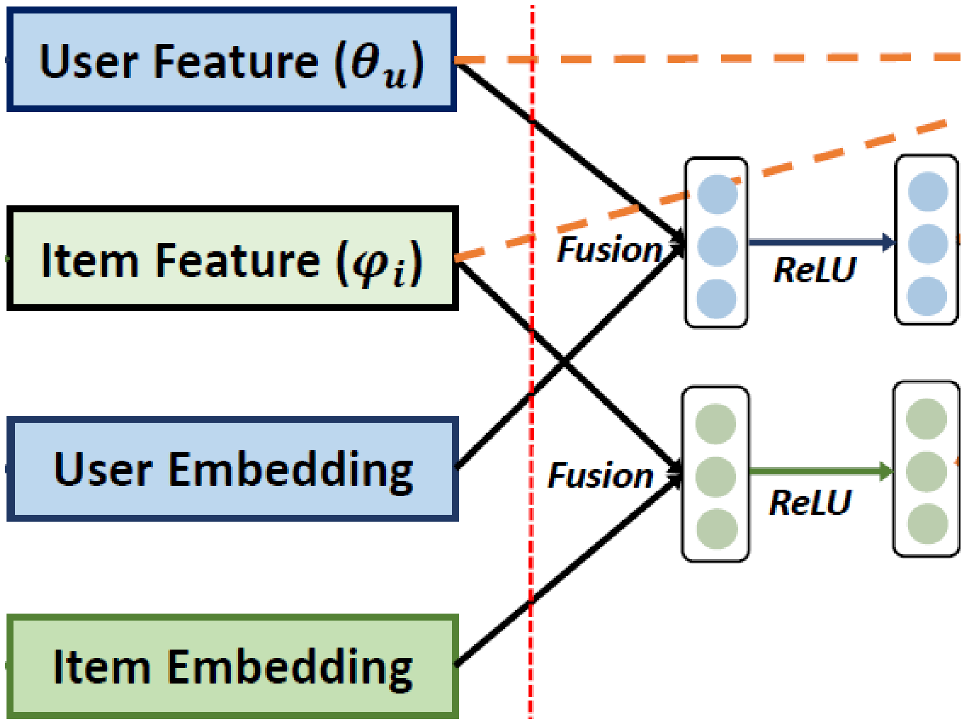
- *K*: number of latent topics
- θ_u: user feature – topic distribution of user *u*
- φ_i: item feature – topic distribution of item *i*
- π_u: decide the current word w is drawn from θ_u or φ_i
- *w:* a word in the review
- *z:* the latent topic of the word *w*

- **Assumption**:
  - ✓ A sentence in a review fucoses on the same topic *z*
  - ✓ When written a sentence, a user could comment from his own preferences θ_u or from item's characteristics φ_u : user-dependent parameter: π_u
- **Our model**: mimics the processing of writing a review sentence
- **Goal**: Estimate θ_u and φ_{i}

OUR MODEL – FUSION MODULE

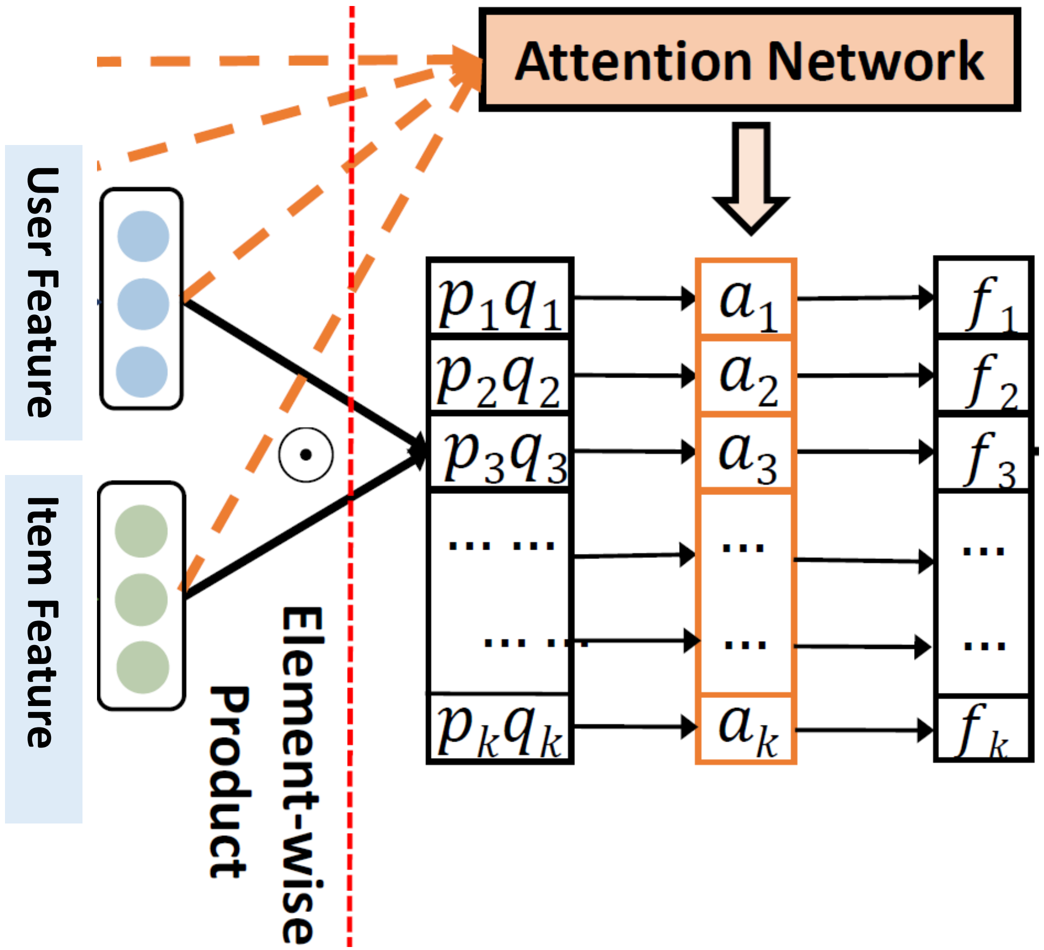


OUR MODEL – FUSION MODULE



- **Fusion**: embedded feature + review-based feature
  - ✓ *Concatenation, **addition**, element-wise product*
- **ReLu fully-connected layer:** further increasing the interaction between the two types of features

OUR MODEL – ATTENTION MODULE

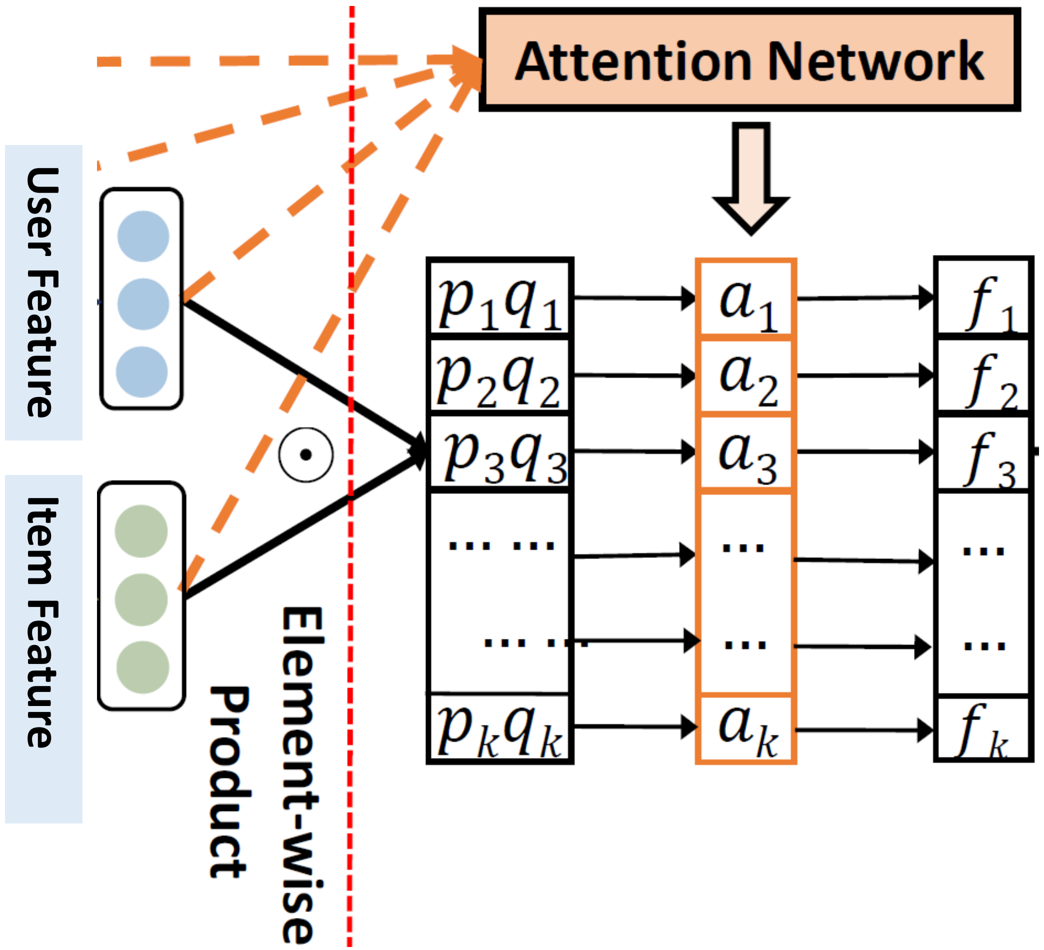


- $\boldsymbol{p_u}$: k-dimensional user feature
- $\boldsymbol{q_i}$: k-dimensional item feature
- **Rating prediction:** inner product of *user-feature* and *item-feature*
- **Attention weight vector $\boldsymbol{a_{u,i}}$**: introduce an attention weight $a_{u,i,k}$ to a factor k to indicate the importance of this factor of item i with respect to user u
  - For a user u, the importance weight of the factors are different with respect to each item i

$$F = a_{u,i} \odot (p_u \odot q_i)$$

***F*: *k*-dimensional feature → rating prediction**

OUR MODEL – ATTENTION MECHANISM

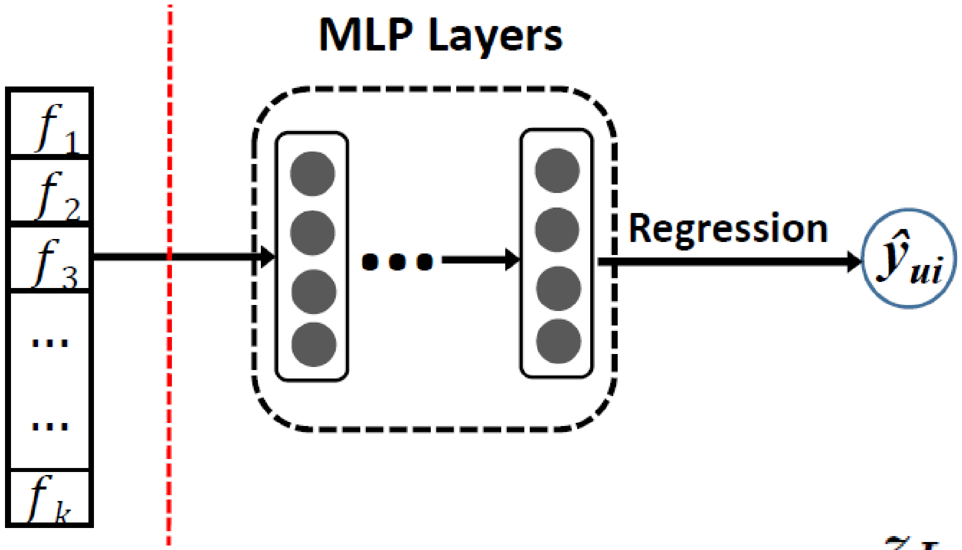


- **How to estimate the attention weight**
  - User preferences and item characteristics can be observed in reviews -> $\boldsymbol{\theta_u}$ and $\boldsymbol{\varphi_i}$
  - $\boldsymbol{p_u}$ and $\boldsymbol{q_i}$ are the fusion feature for the final prediction
  - Concatenation of the four feature: $\boldsymbol{\theta_u}, \boldsymbol{\varphi_i}, \boldsymbol{p_u}, \boldsymbol{q_i}$
- **Attention mechanism**:

$$\hat{\boldsymbol{a}}_{\boldsymbol{u},\boldsymbol{i}} = \boldsymbol{v}^T ReLU(\boldsymbol{W}_a[\boldsymbol{\theta_u}; \boldsymbol{\varphi}_i; \boldsymbol{p_u}; \boldsymbol{q_i}] + \boldsymbol{b}_a)$$

$$a_{u,i,k} = \frac{exp(\hat{a}_{u,i,k})}{\sum_{j=1}^{K} exp(\hat{a}_{u,i,j})}$$

OUR MODEL – RATING PREDICTION



- The obtained feature is fed into fully connected layers (one layer in our experiments)
- Rating prediction: regression

$$\boldsymbol{z_L} = \sigma_L(\boldsymbol{W}_L(\sigma_{L-1}(\boldsymbol{W}_{L-1}\cdots\sigma_1(\boldsymbol{W}_1\boldsymbol{F}+\boldsymbol{b}_1))+\boldsymbol{b}_{L-1})+\boldsymbol{b}_L)$$

$$\hat{r}_{u,i} = \boldsymbol{W}\boldsymbol{z}_L + \boldsymbol{b}$$

EXPERIMENTAL SETUP

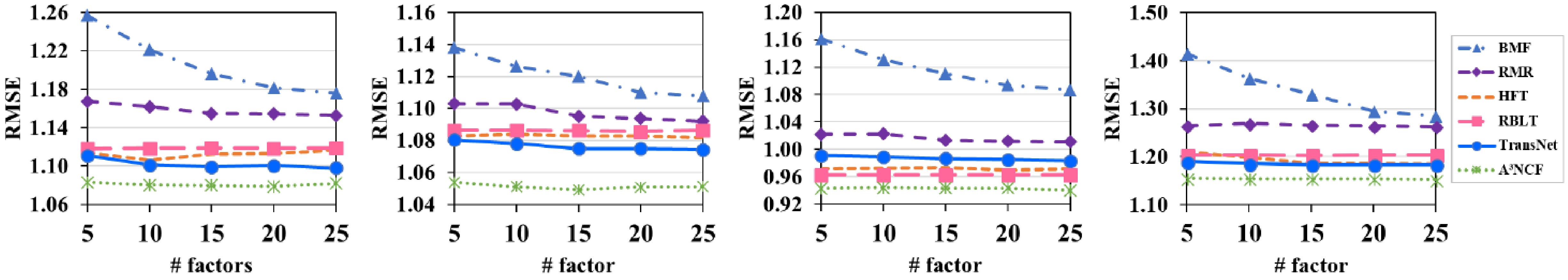
- **Dataset:** Five sub-datasets in the Amazon product Review dataset and The Yelp Dataset 2017
- **Setting**: training:validation:testing = 8:1:1
- **Task**: Rating prediction
- **Metrics**: RMSE (the smaller the better)

Datasets	# users	# items	# ratings	Sparsity
Baby	17,177	7,047	158,311	0.9987
Grocery	13,979	8,711	149,434	0.9988
Home & Kitchen	58,901	28,231	544,239	0.9997
Garden	1,672	962	13,077	0.9919
Sports	31,176	18,355	293,306	0.9995
Yelp2017	169,257	63,300	1,659,678	0.9998

EXPERIMENTAL SETUP - COMPETITORS

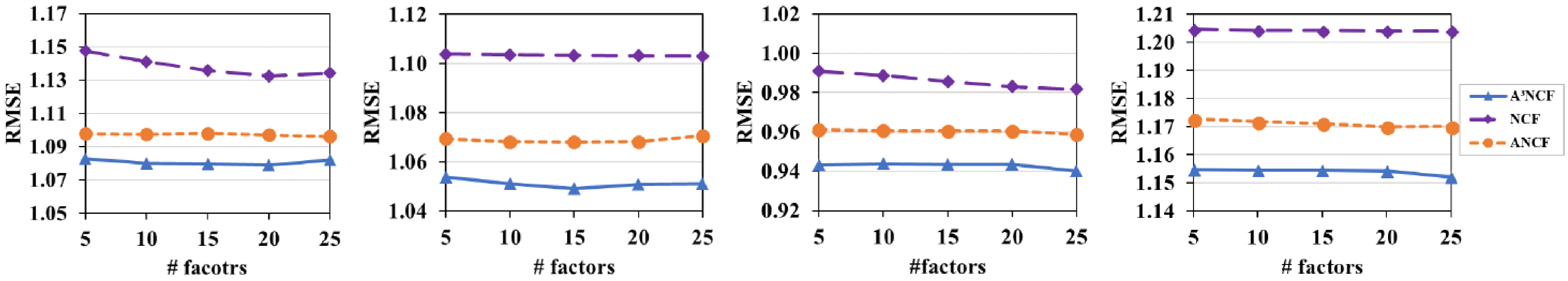
- **BMF:** Matrix factorization (MF) with biased terms
- **HFT:** Use *a linking function* to connect the latent factors in MF (ratings) and LDA (reviews)
- **RMR:** Mixture of Gaussian (ratings) +LDA (reviews)
- **RBLT:** Use *a linear combination* of the latent factors in MF (ratings) and LDA (reviews)
- **TransNet:** Neural networks on user and item reviews for rating prediction

PERFORMANCE COMPRASIONS



- All better than **BMF**: indicating the importance of reviews in preference modeling
- Review-based methods
  - are relative more stable than BMF with the increase of *#factor*;
  - can achieve relatively good performance with a small *#factor*
- A^{3}NCF is the best; > RBLT (2.9% ↑) and > TransNet (2.2%↑), because it
  - applies more complicate interactions to integrate reviews and ratings via non-linear neural networks,
  - uses an attention mechanism to capture users' attention weights on different aspects of an item.

EFFECTS OF ASPCT ATTENTION



- **Comparisons**
  - NCF: without *review-based feature* and *attention mechanism*
  - ANCF: with review-based feature but without *attention mechanism*
- **Results**
  - **ANCF > NCF**: (1) the effectiveness of using reviews in recommendation; and (2) our model on integrating review and rating information
  - **A^{3}NCF > ANCF**: (1) user's attentions are varied for different items; and (2) the effectiveness of our attention model

CONCLUSIONS

- Advocate the point that "a user may place different attentions to different items"
- Propose an attentive neural network to capture a user's attention weight for different items
- Conduct experiments on benchmarking dataset to demonstrate our viewpoints and the effectiveness of the proposed model

Thanks！

Homepage: https://sites.google.com/view/zycheng
E-mail: zhiyong.cheng@nus.edu.sg or jason.zy.cheng@gmail.com