Gated Attentive-Autoencoder for Content-Aware Recommendation

Chen Ma[1], Peng Kang[1], Bin Wu[2], Qinglong Wang[1] and Xue Liu[1]

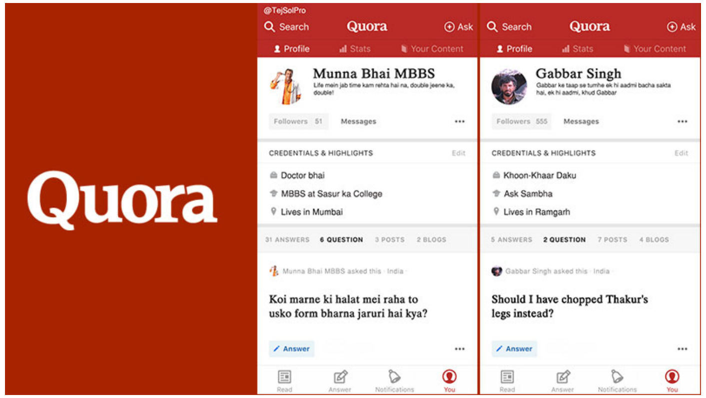
[1]McGill University, Montreal, Canada
[2]Zhengzhou University, Zhengzhou, China

WSDM2019, Melbourne, Australia

Background

The rapid growth of Internet services allows users to access millions of online products, such as movies, articles.

The large amount of user-item data facilitates a promising and practical service – the **personalized recommendation**.



Background

The rapid growth of Internet services allows users to access millions of online products, such as movies, articles.

movie \ user	1	2	3	4	5
1			✓		✓
2	✓				
3		✓			✓
4			✓		
5				✓	✓

Background

The rapid growth of Internet services allows users to access millions of online products, such as movies, articles.

movie \ user	1	2	3	4	5
1			✓		✓
2	✓				
3		✓			✓
4			✓		
5				✓	✓

Background

The rapid growth of Internet services allows users to access millions of online products, such as movies, articles.

movie \ user	1	2	3	4	5
1			✓		✓
2	✓				
3		✓			✓
4			✓		
5				✓	✓

I

II

III

Background

The rapid growth of Internet services allows users to access millions of online products, such as movies, articles.

movie \ user	1	2	3	4	5
1			✓		✓
2	✓				
3		✓			✓
4			✓		
5				✓	✓

I

II

III

Content helps

Less privacy issue

Related Work

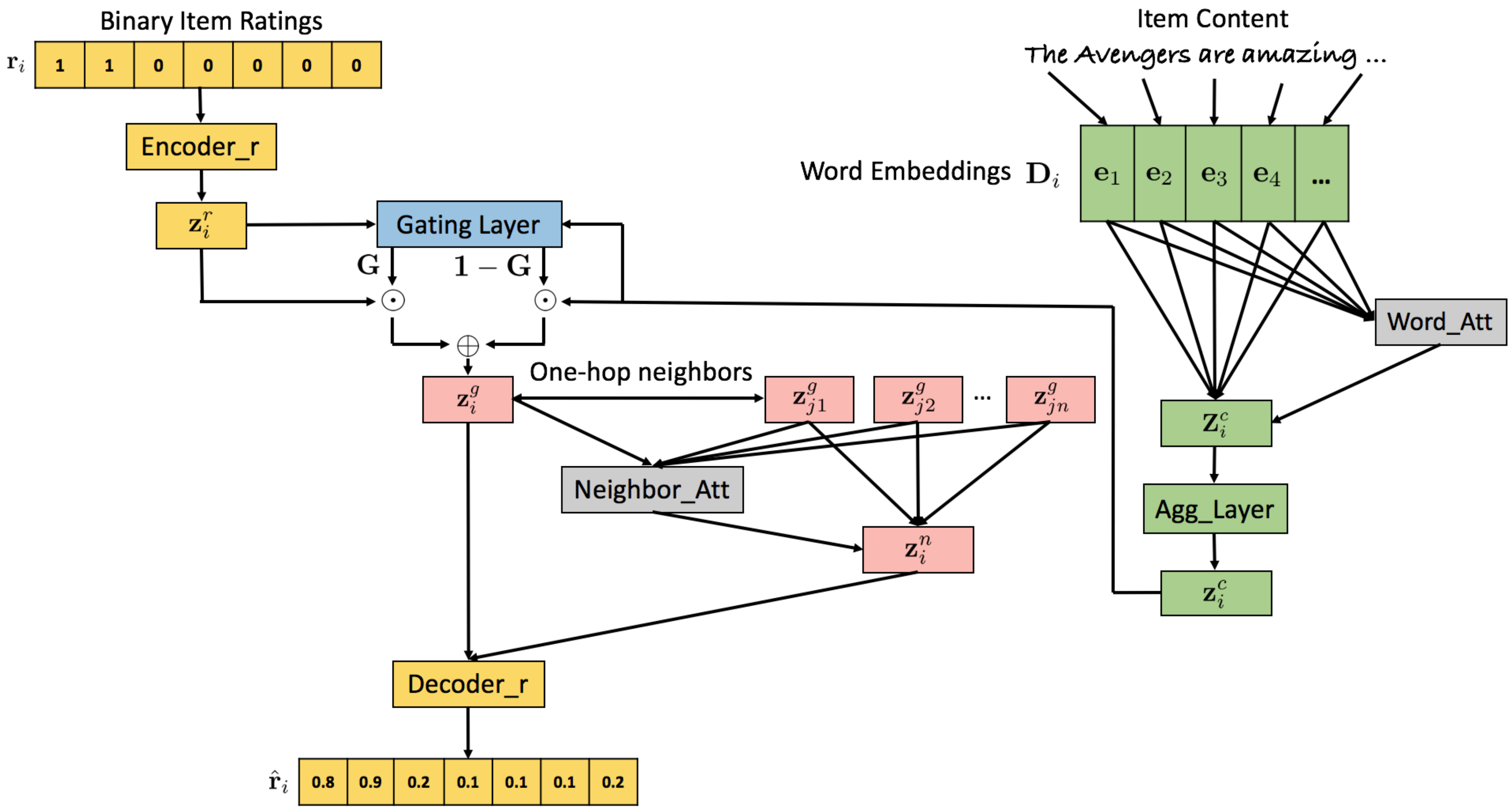
Models	Algorithms
CTR (*Wang et al., SIGKDD' 2011*)	MF + LDA
SVDFeature (*Chen et al., JMLR' 2012*)	Feature-based MF
HFT (*Julian et al., RecSys' 2013*)	LFM + LDA
CDL (*Wang et al., SIGKDD' 2015*)	MF + SDAE
ConvMF (*Kim et al., RecSys' 2016*)	MF + CNN
CVAE (*Li et al., SIGKDD' 2017*)	MF + VAE

MF: Matrix Factorization
LDA: Latent Dirichlet Allocation
LFM: Latent Factor Model
SDAE: Stacked Denoising AutoEncoder
VAE: Variational AutoEncoder

- Equally treat item content
- Combine the rating and content information by regularization
- Not explicitly utilize the item-item relations

Model Overview

An **autoencoder**-based model:



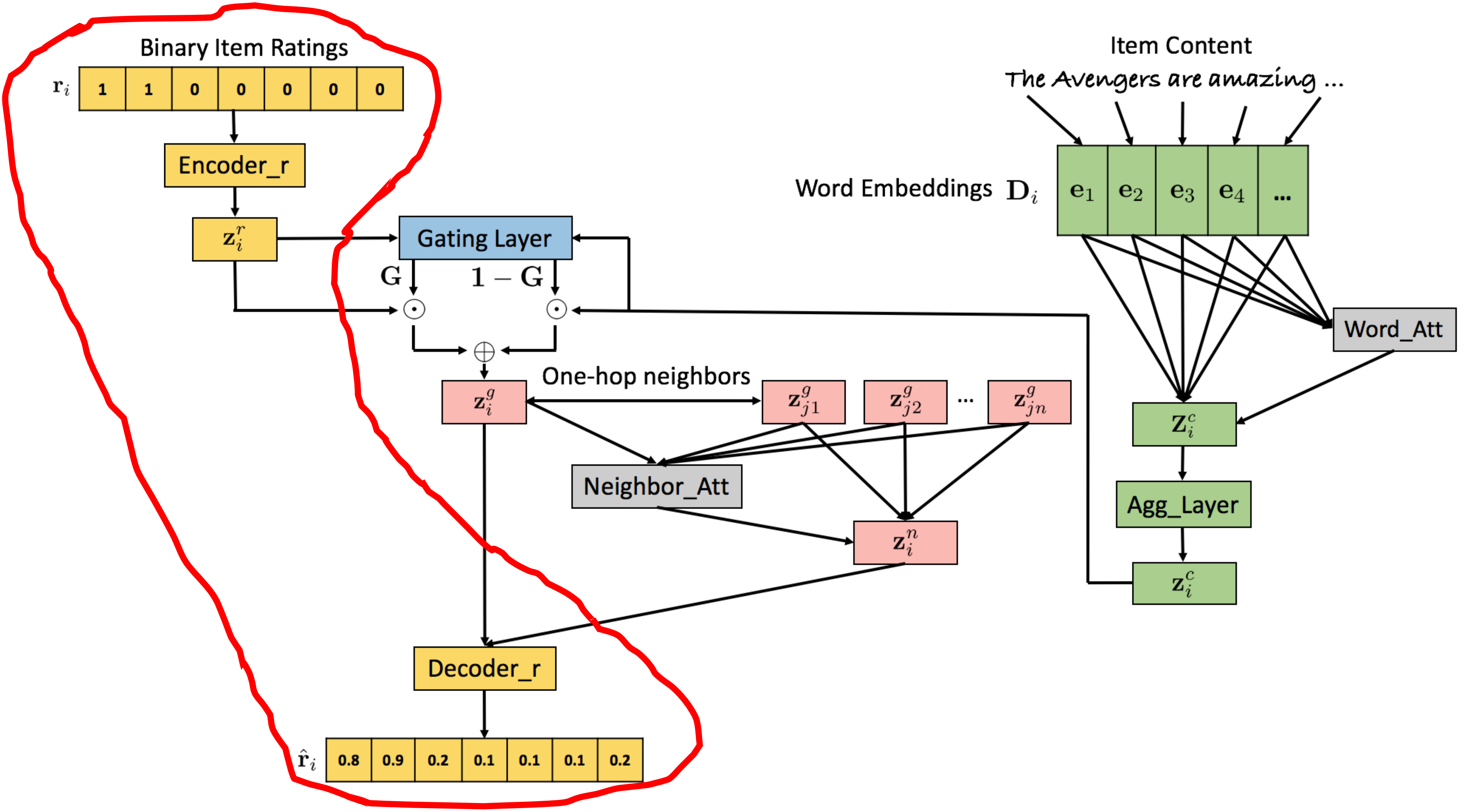
Word-attention

Gating layer

Neighbor-attention

Model Overview

An **autoencoder**-based model:



Word-attention

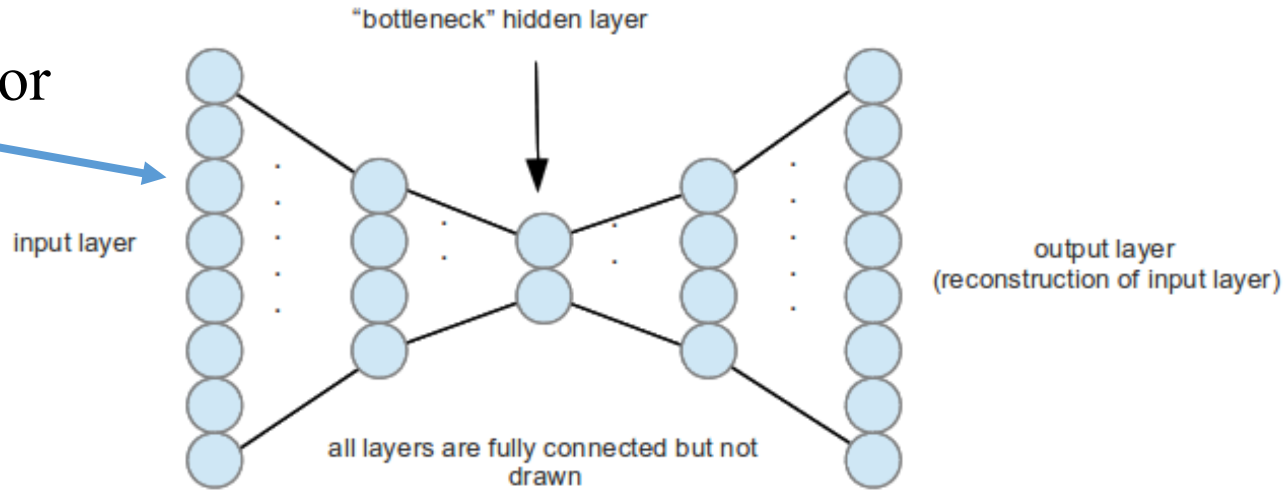
Gating layer

Neighbor-attention

Autoencoder

- Autoencoder is used to learn the item hidden representations from rating information.

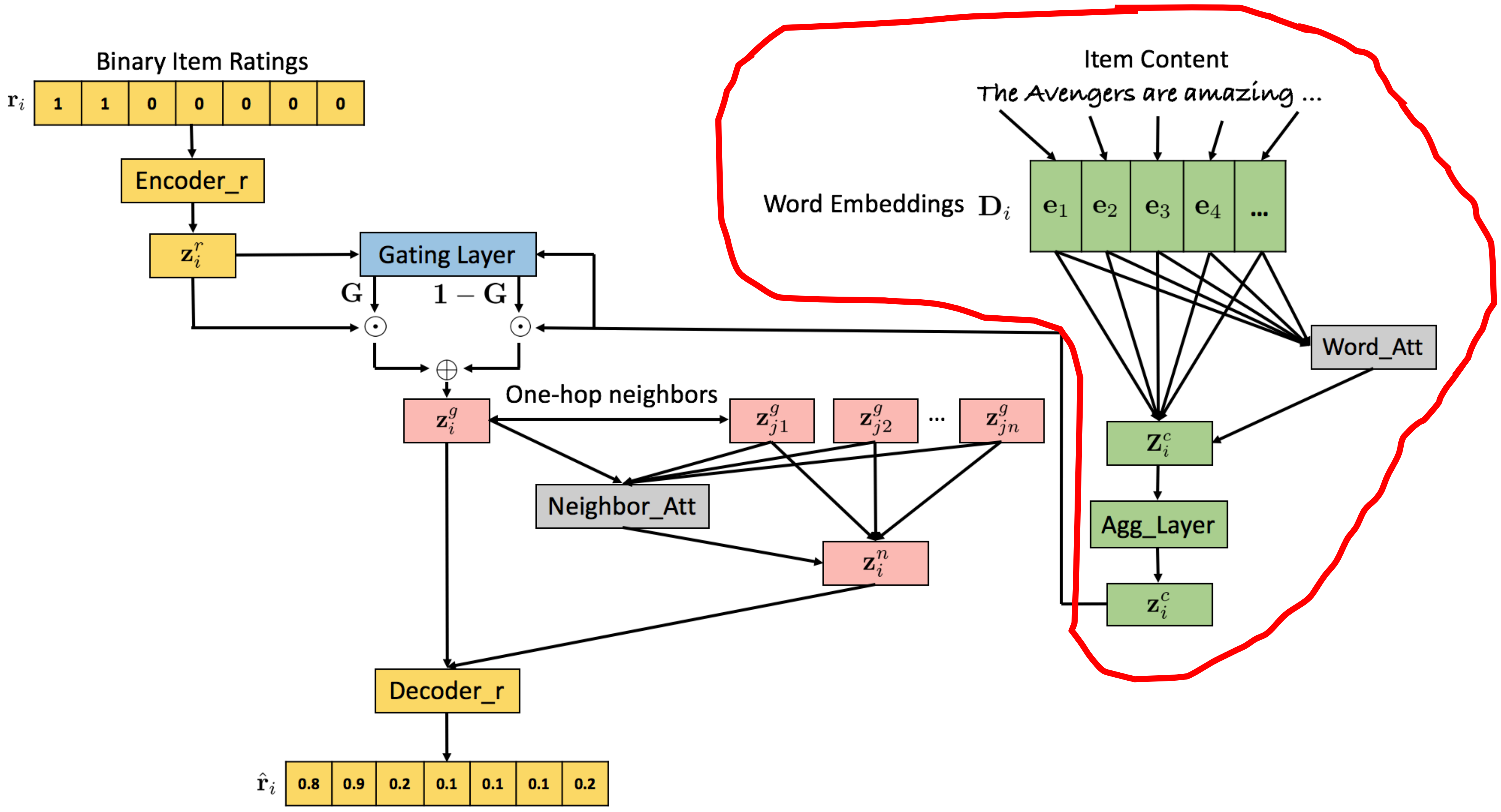
binary item rating vector



$$enc : \begin{cases} \mathbf{z}_i^{(1)} = a_1(\mathbf{W}_1\mathbf{r}_i + \mathbf{b}_1) \\ \mathbf{z}_i^{r} = a_2(\mathbf{W}_2\mathbf{z}_i^{(1)} + \mathbf{b}_2) \end{cases} \quad dec : \begin{cases} \mathbf{z}_i^{(3)} = a_3(\mathbf{W}_3\mathbf{z}_i^{r} + \mathbf{b}_3) \\ \hat{\mathbf{r}}_i = a_4(\mathbf{W}_4\mathbf{z}_i^{(3)} + \mathbf{b}_4) \end{cases} \quad loss : \mathcal{L} = \sum_{i=1}^{M} ||\mathbf{x}_i - \hat{\mathbf{x}}_i||_2^2$$

http://nghiaho.com/?p=1765

Word-attention Module



Word-attention Module

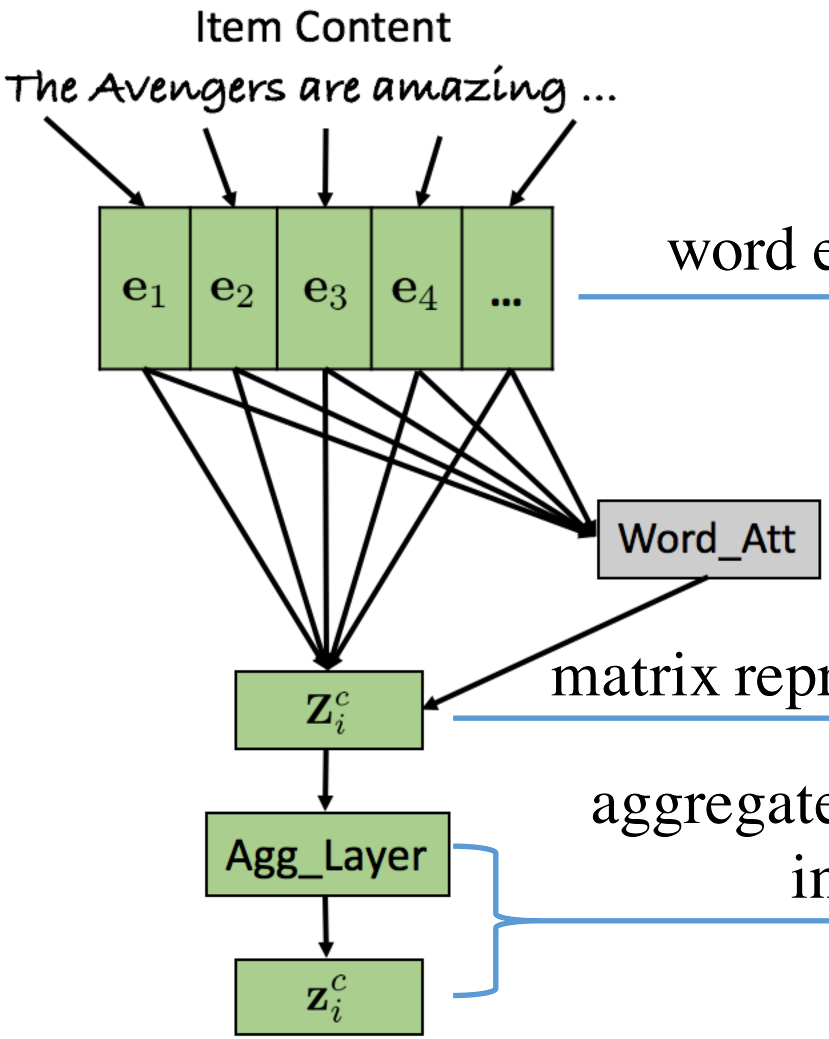
- Previous works do not discriminate the word importance for describing a certain item
- Some informative words are more representative than others and should contribute more to characterize a certain item

  E.G. great food and good service .... what else can you ask for

- **We utilize an attention model to learn the item representation from content information**.

Lin et al., A Structured Self-attentive Sentence Embedding, ICLR 2017

Word-attention Module



word embedding look-up

$$\mathbf{D}_i = \begin{bmatrix} \dots & | & | & | & \dots \\ & \mathbf{e}_{j-1} & \mathbf{e}_j & \mathbf{e}_{j+1} & \\ & | & | & | & \end{bmatrix}$$

attention score matrix

$$\mathbf{A}_i = softmax(\mathbf{W}_{a_1} tanh(\mathbf{W}_{a_2}\mathbf{D}_i + \mathbf{b}_{a_2}) + \mathbf{b}_{a_1})$$

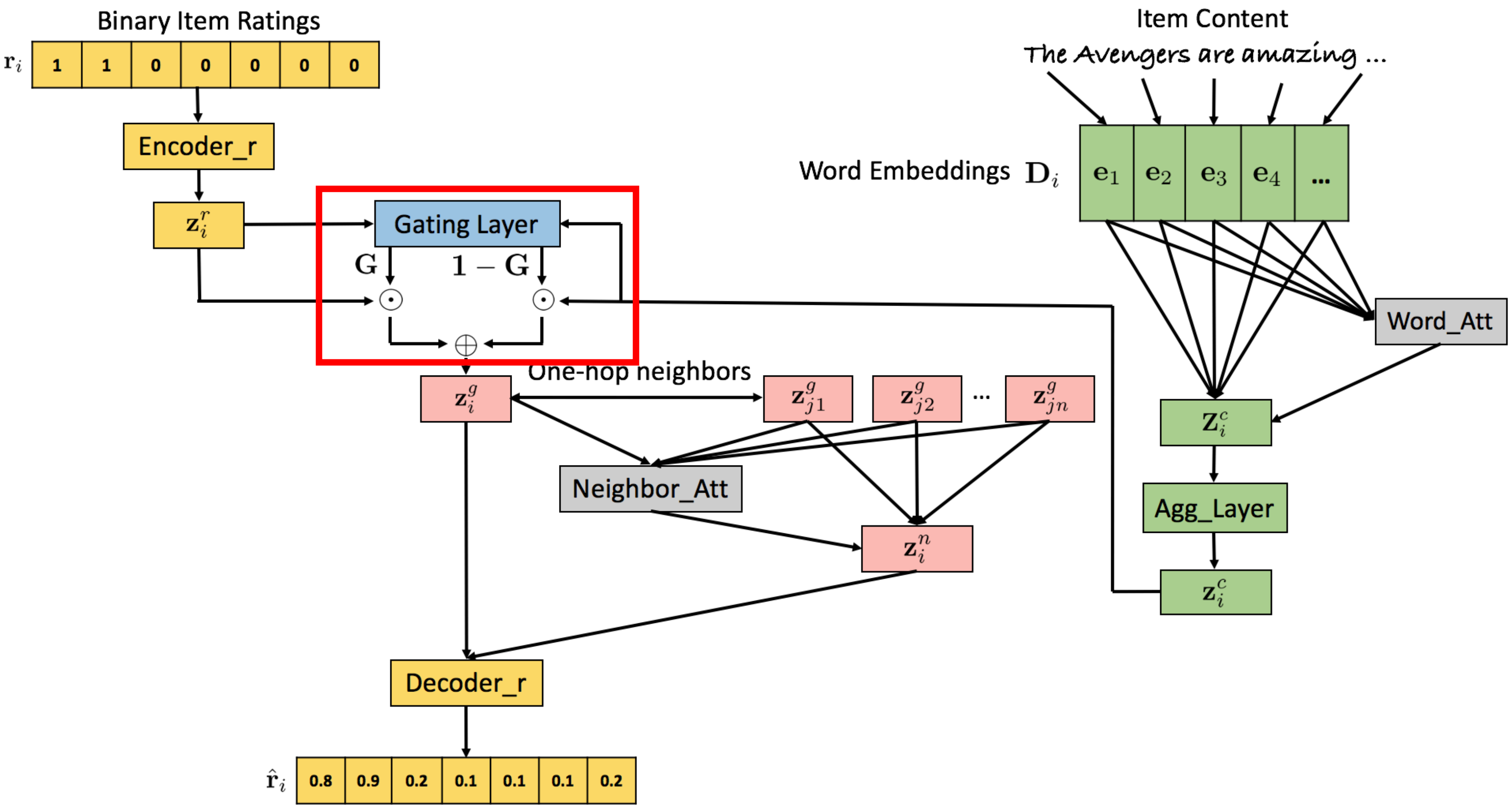
matrix representation of items

$$\mathbf{Z}_i^c = \mathbf{A}_i \mathbf{D}_i^\top$$

aggregate item representations into one aspect

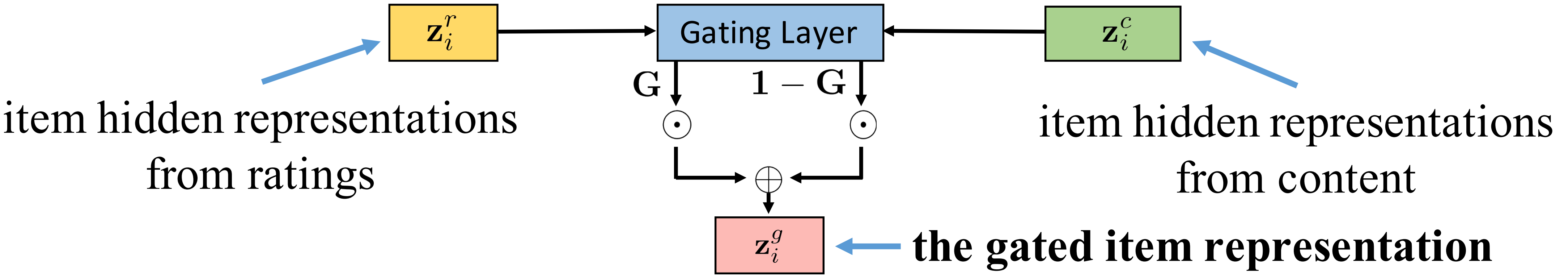
$$\mathbf{z}_i^c = a_t(\mathbf{Z}_i^{c\top} \mathbf{w}_t)$$

Gating Layer



Gating Layer

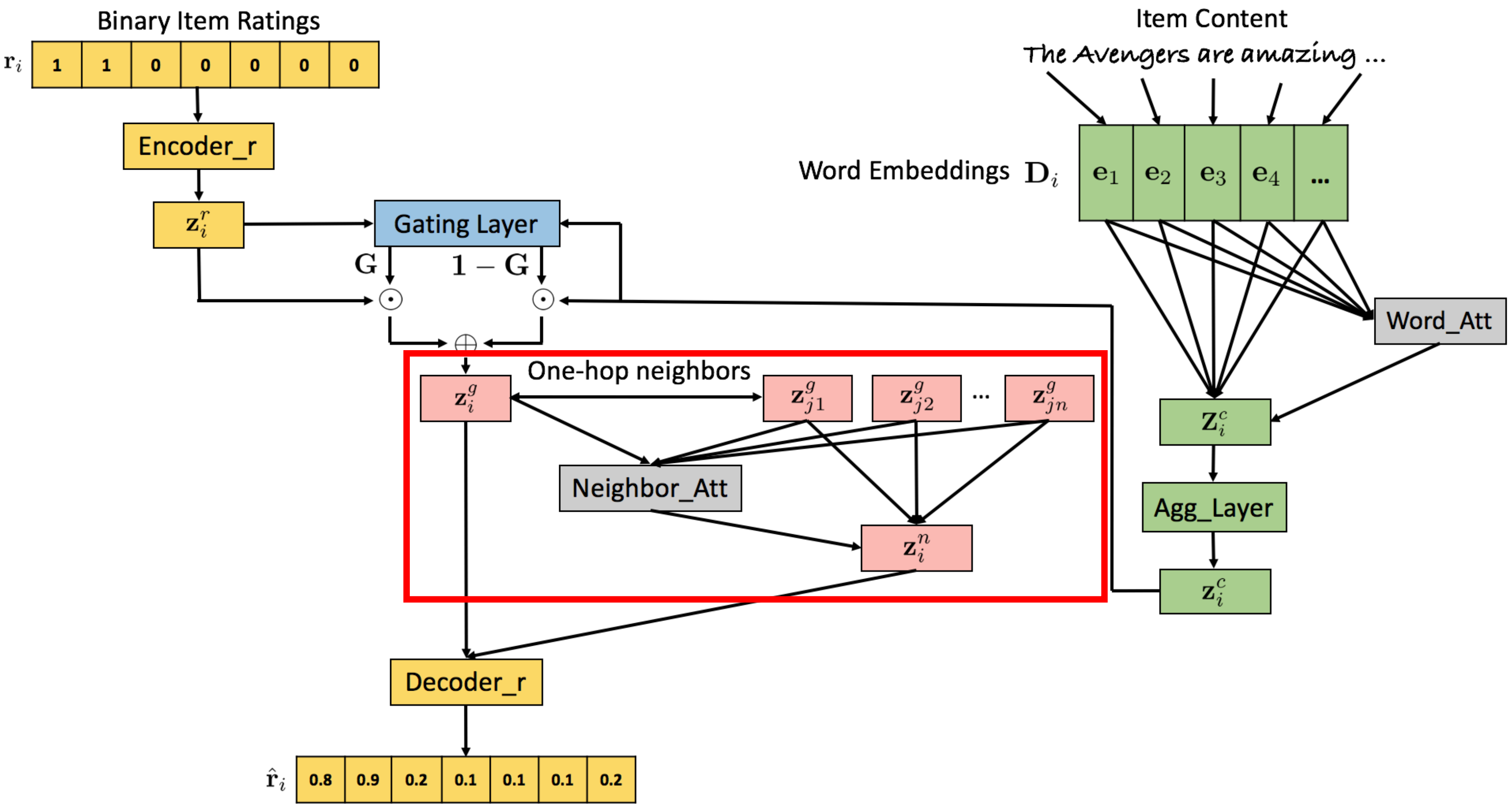
- Adaptively fuse the hidden representations from two heterogeneous data sources
- Avoid tedious hyper-parameter tuning by the regularization term



$$\mathbf{G} = sigmoid(\mathbf{W}_{g_1}\mathbf{z}_i^r + \mathbf{W}_{g_2}\mathbf{z}_i^c + \mathbf{b}_g)$$ → adaptively learn the gate

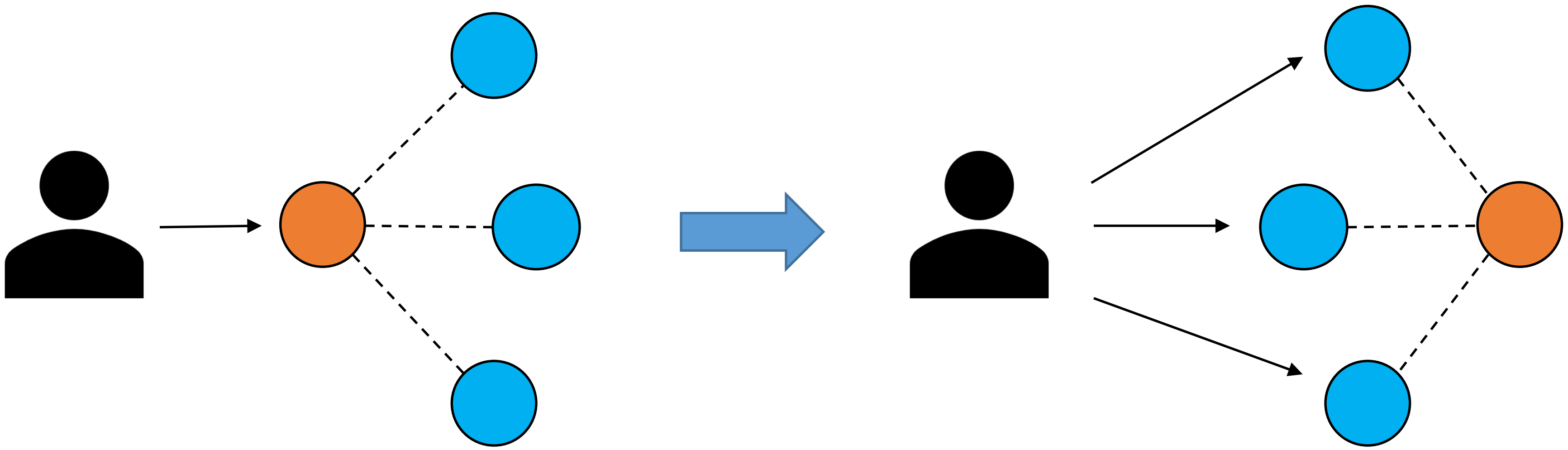
$$\mathbf{z}_i^g = \mathbf{G} \odot \mathbf{z}_i^r + (1 - \mathbf{G}) \odot \mathbf{z}_i^c$$ → combine hidden representations

Neighbor-attention Module

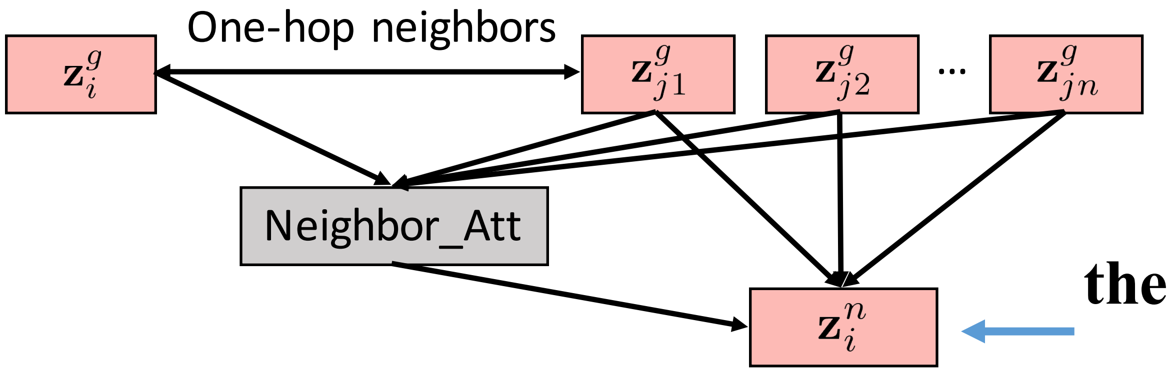


Neighbor-attention Module

- Previous works do not consider the relations between items
- Related items may share the same topic or have similar attributes: citations between articles, movies in the same genre
- **Exploring users' preferences on an item's neighbors also benefits inferring users' preferences on this item**



Neighbor-attention Module



the item neighborhood representation

$s_{i,j} = tanh(\mathbf{z}_i^{g\top} \mathbf{W}_n \mathbf{z}_j^g), \forall j \in \mathcal{N}_i$ → use a bilinear function to capture the relation

$\mathbf{a}_i = softmax(\mathbf{s}_i)$ → the attention score of item i's neighbors

$\mathbf{z}_i^n = \sum_{j \in \mathcal{N}_i} a_{i,j} \mathbf{z}_j^g$ → the neighborhood representation item i

Prediction and Loss

- Modified decoder: explore users' preferences on both an item and its neighborhood

$$\mathbf{z}_i^{(3,g)} = a_3(\mathbf{W}_3\mathbf{z}_i^g + \mathbf{b}_3),$$
$$\mathbf{z}_i^{(3,n)} = a_3(\mathbf{W}_3\mathbf{z}_i^n + \mathbf{b}_3),$$
$$\hat{\mathbf{r}}_i = a_4(\mathbf{W}_4\mathbf{z}_i^{(3,g)} + \mathbf{W}_4\mathbf{z}_i^{(3,n)} + \mathbf{b}_4).$$

- Weighted loss

$$\mathcal{L}_{AE} = \sum_{i=1}^{n}\sum_{u=1}^{m} ||C_{u,i}(R_{u,i} - \hat{R}_{u,i})||_2^2 = ||\mathbf{C}^\top \odot (\mathbf{R}^\top - \hat{\mathbf{R}}^\top)||_F^2 \quad C_{u,i} = \begin{cases} \rho & \text{if } R_{u,i} = 1 \\ 1 & otherwise \end{cases}$$

Evaluation

- Four datasets

Dataset	#Users	#Items	#Ratings	#Words	Density
citeulike-a	5,551	16,980	204,986	8,000	0.217%
ML20M	138,493	18,307	19,977,049	12,397	0.788%
Books	65,476	41,264	1,947,765	27,584	0.072%
CDs	24,934	24,634	478,048	24,341	0.078%

For each user, **20%** of her viewed items are selected as testing.

- Evaluation Metrics
  - Recall@5, 10, 15, 20
  - NDCG @5, 10, 15, 20

Evaluation Baselines

WRMF: weighted regularized matrix factorization, ICDM' 2008

CDAE: collaborative denoising autoencoder, WSDM' 2016

Classical CF methods

CDL: collaborative deep learning, SIGKDD' 2015

CVAE: collaborative variational autoencoder, SIGKDD' 2015

CML+F: collaborative metric learning, WWW' 2017

Learning from bag-of-words

ConvMF: convolutional matrix factorization, RecSys' 2016

JRL: joint representation learning, CIKM' 2017

Learning from word sequence

Evaluation Results

	WRMF	CDAE	CDL	CVAE	CML+F	ConvMF	JRL	GATE	Improv.
Recall@10									
citeulike-a	0.0946	0.0888	0.1317	0.1371	0.1283	0.1153	0.1325	**0.1419**	3.50%
movielens-20M	0.1075	0.0751	0.1287	0.1303	0.1123	0.1201	0.1401	**0.1625****	15.99%
Amazon-Books	0.0553	0.0132	0.0648	0.0632	0.0756	0.0524	0.0924	**0.1133***	22.62%
Amazon-CDs	0.0779	0.0191	0.0827	0.0811	0.0824	0.0753	0.0816	**0.1057*****	27.81%
NDCG@10									
citeulike-a	0.0843	0.0736	0.0949	0.0952	0.1035	0.0914	0.0982	**0.1082**	4.54%
movielens-20M	0.1806	0.1774	0.1836	0.1939	0.2479	0.1807	0.2439	**0.2992****	20.69%
Amazon-Books	0.0377	0.0105	0.0393	0.0384	0.0456	0.0324	0.0592	**0.0708*****	19.59%
Amazon-CDs	0.0357	0.0105	0.0356	0.0349	0.0364	0.0323	0.0386	**0.0477*****	23.58%

*: $p <= 0.05$, ** $p < 0.01$, ***: $p < 0.001$

Our GATE outperforms other methods significantly on most of the datasets

Evaluation Results

- Ablation study

Architecture	*CDs*		*Books*	
	R@10	N@10	R@10	N@10
(1) stacked AE	0.0672	0.0315	0.0745	0.0484
(2) reg: AE + W_Att	0.0676	0.0318	0.0304	0.0265
(3) gating: AE + W_Att	0.0816	0.0353	0.0793	0.0515
(4) gating: AE + GRU	0.0818	0.0352	0.0789	0.0512
(5) gating: AE + CNN	0.0777	0.0335	0.0791	0.0495
(6) GATE	**0.1057**	**0.0477**	**0.1133**	**0.0708**

From (2), (3): our gating is better than regularization
From (3), (4), (5): our word-attention achieves similar performance with fewer parameters
From (3), (6): the item-item relations play an important role

Evaluation Results

- Case Study

Target	Neighbor	Score
Fluctuations in network dynamics	Genomic analysis of regulatory network dynamics reveals large topological changes (0)	0.07172
	Frequency of occurrence of numbers in the World Wide Web (10)	0.22090
	Complex networks: Structure and dynamics (16)	0.26835
	Noise in protein expression scales with natural protein abundance (**36**)	**0.43903**

The Summary of Article 16797 in *citeulike-a*
We present the first parallel implementation of the T-Coffee consistency-based multiple aligner. We benchmark it on the Amazon Elastic Cloud (EC2) and show that the parallelization procedure is reasonably effective. We also conclude that for a web server with moderate usage (10K hits/month) the cloud provides a cost-effective alternative to in-house deployment.

Conclusion

We propose an **autoencoder-based** model, which consists of a **word-attention module**, a **neighbor-attention module**, and a **gating layer** to address the content-aware recommendation task.

Experimental results show that the proposed method outperforms the state-of-the-art methods significantly for content-aware recommendation.

Thank you!

Q & A

Email: chen.ma2@mail.mcgill.ca
Code: https://github.com/allenjack/GATE
LibRec: https://www.librec.net/

Google '**LibRec**'