Attentive Collaborative Filtering:

Multimedia Recommendation with Item- and Component-Level Attention

August 2017

Daria Walter
February 19, 2018

Motivation: attention in multimedia recommendation

- Implicit feedback: blurred users' preferences
- A lot of contextual information: images, videos

- What items does a user **really** like and what **makes** a user like it?

Recommendation with implicit feedback

- ***M*** users, ***N*** items. User-item interaction matrix:

$$\mathbf{R} \in \mathbb{R}^{M \times N}$$

- Set of user-item pairs with **observed** feedback:

$$\mathcal{R} = \{(i, j | R_{ij} = \{0, 1\})\}$$

- Goal: estimate all values in user-item rating matrix

$$\widehat{R}_{ij} - \text{predicted rating for } i \text{ user }, j \text{ item}$$

Baseline methods

- Collaborative filtering:
  - Latent factor model
  - Bayesian personalized ranking
  - NSVD, SVD++

- Hybrid models: CF + contextual information
  - SVDFeature
  - DeepHybrid

Baseline methods: Latent factor model

- Users and items are mapped to the shared latent space

$$\mathbf{u_i} - \text{ user latent vector}, \mathbf{v_j} - \text{ item latent vector}$$

- Rating is a dot product:

$$\hat{R}_{ij} =< \mathbf{u}_i, \mathbf{v}_j >= \mathbf{u}_i^T \mathbf{v}_j.$$

- Objective function:

$$\arg\min_{\mathbf{U},\mathbf{V}} \sum_{(i,j)\in\mathcal{R}} (R_{ij} - \hat{R}_{ij})^2 + \lambda(||\mathbf{U}||^2 + ||\mathbf{V}||^2)$$

Baseline methods: Bayesian personalized ranking

- Latent factor model + pairwise loss

$$\hat{R}_{ij} =< \mathbf{u}_i, \mathbf{v}_j >= \mathbf{u}_i^T \mathbf{v}_j.$$

- Training triples (user ***i***, clicked item ***j***, not clicked item ***k***) :

$$\mathcal{R}_B = \{(i,j,k) | j \in \mathcal{R}(i) \wedge k \in \mathcal{I} \backslash \mathcal{R}(i)\}$$

- Pairwise loss to address *implicitness* of feedback

$$\arg\min_{\mathbf{U},\mathbf{V}} \sum_{(i,j,k) \in \mathcal{R}_B} - \ln \sigma(\hat{R}_{ij} - \hat{R}_{ik}) + \lambda(||\mathbf{U}||^2 + ||\mathbf{V}||^2)$$

Tweaking latent factor model

- NSVD - user parametrization with item vectors

$$\widehat{R}_{ij} = v_j^T u_i \longrightarrow \widehat{R}_{ij} = v_j^T \underbrace{\left(\frac{1}{|\mathcal{R}_{(i)}|} \sum_{l \in \mathcal{R}_{(i)}} p_l\right)}_{\text{user}}$$

- SVD++

$$\widehat{R}_{ij} = v_j^T \underbrace{\left(u_i + \frac{1}{|\mathcal{R}_{(i)}|} \sum_{l \in \mathcal{R}_{(i)}} p_l\right)}_{\text{user}}$$

or

$$\widehat{R}_{ij} = \overbrace{v_j^T u_i}^{\text{latent factor model}} + \underbrace{\left(\frac{1}{|\mathcal{R}_{(i)}|} \sum_{l \in \mathcal{R}_{(i)}} v_j^T p_l\right)}_{\text{neighbourhood model}}$$

Multimedia recommendation: problem statement

- Item ***j*** is represented as a vector of feature vectors - components (e.g. frames of video)

$$\{x_{j,1}, \ldots, x_{j,m}\}$$

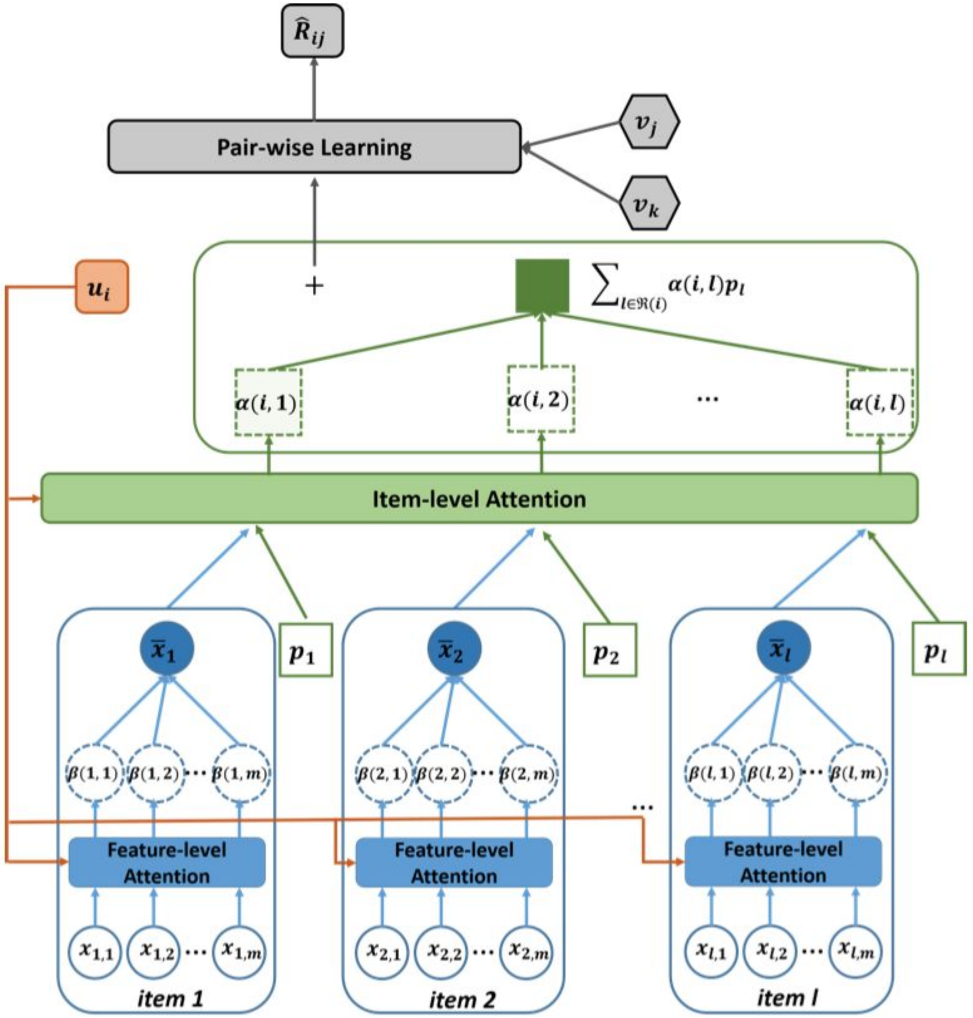
- User ***i*** is associated with a set of items $\mathcal{R}(i)$ from the browsing history
- Item representation: weighted sum of components
- User representation: weighted sum of items

- How to assign weights ?

Architecture

- Trainable parameters:
  - $\mathbf{u}_i$ - user latent vector
  - $\mathbf{v}_j$ - item latent vector
  - $\mathbf{p}_j$ - item auxiliary latent vector
  - W_1, W_2 - parameters in two-level attention modules

$$\hat{R}_{ij} = \left(\mathbf{u}_i + \sum_{l \in \mathcal{R}(i)} \alpha(i,l)\mathbf{p}_l\right)^T \mathbf{v}_j.$$

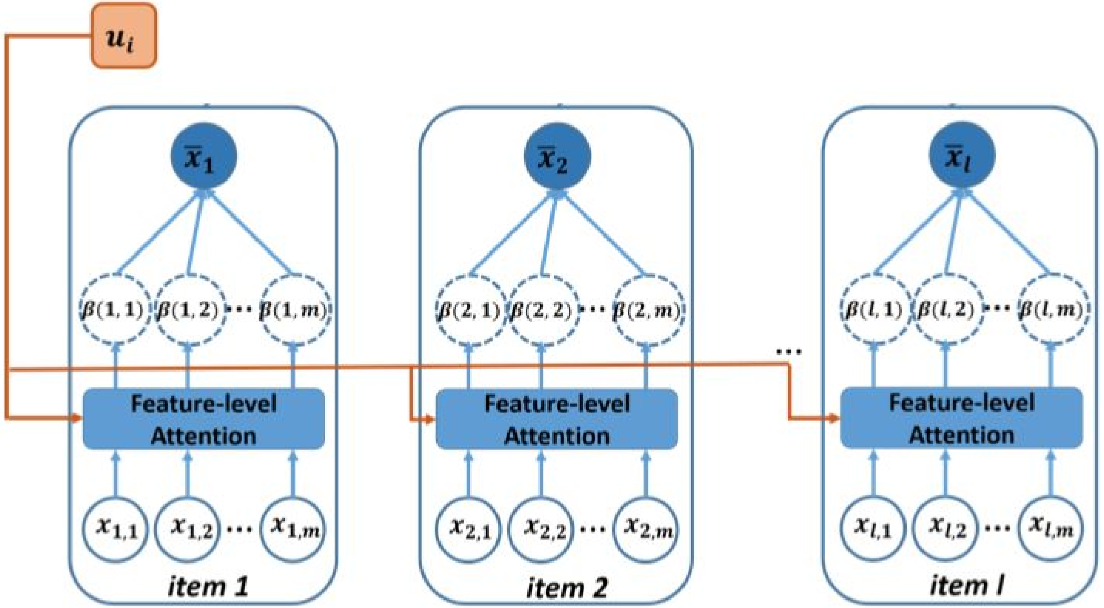


Component-level attention

$$b(i,l,m) = \mathbf{w}_2^T \phi(\mathbf{W}_{2u}\mathbf{u}_i + \mathbf{W}_{2x}\mathbf{x}_{lm} + \mathbf{b}_2) + \mathbf{c}_2$$ - two-layer neural network

$$\beta(i,l,m) = \frac{exp\,(b(i,l,m))}{\sum_{n=1}^{|\{\mathbf{x}_{l*}\}|} exp\,(b(i,l,n))}$$

$$\boxed{\overline{\mathbf{x}}_l = \sum_{m=1}^{|\{\mathbf{x}_{l*}\}|} \beta(i,l,m) \cdot \mathbf{x}_{lm}}$$



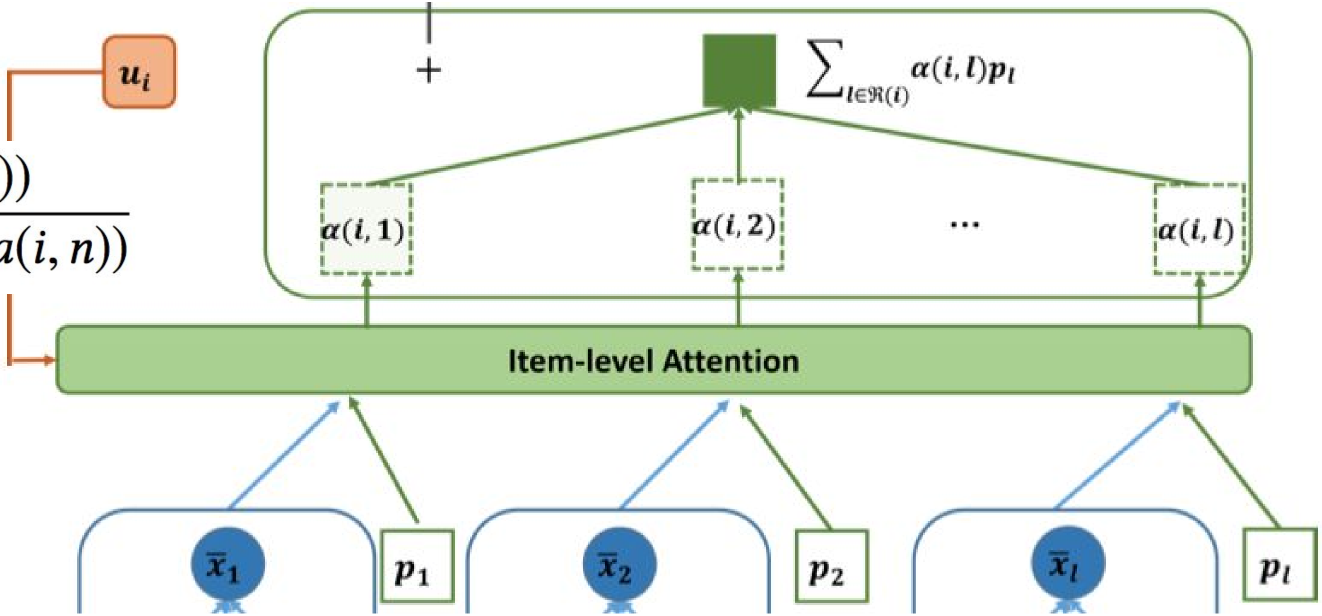
Item-level attention

two-layer neural network:

$$a(i,l) = \mathbf{w}_1^T \phi(\mathbf{W}_{1u}\mathbf{u}_i + \mathbf{W}_{1v}\mathbf{v}_l + \mathbf{W}_{1p}\mathbf{p}_l + \mathbf{W}_{1x}\overline{\mathbf{x}}_l + \mathbf{b}_1) + \mathbf{c}_1$$

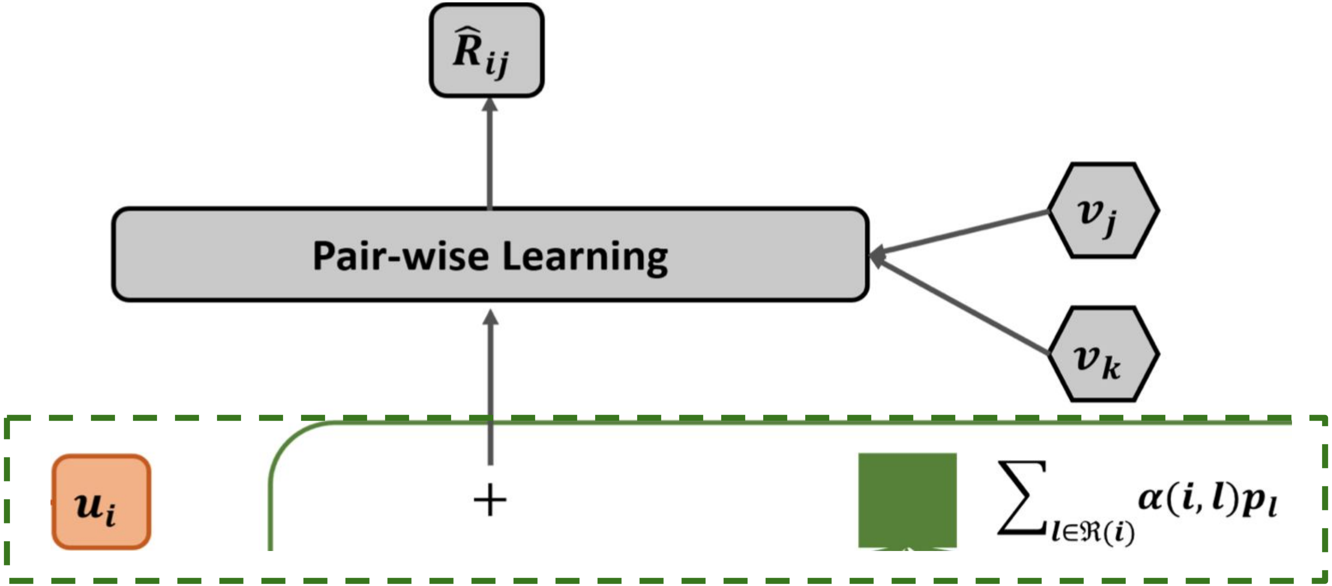
$$\alpha(i,l) = \frac{exp\,(a(i,l))}{\sum_{n\in\mathcal{R}(i)} exp\,(a(i,n))}$$

$$\mathbf{u}_i + \sum_{l\in\mathcal{R}(i)} \alpha(i,l)\mathbf{p}_l$$



Objective function

$$\arg\min_{\mathbf{U},\mathbf{V},\mathbf{P},\Theta} \sum_{(i,j,k)\in\mathcal{R}_B} -\ln\sigma\left\{\underbrace{\left(\mathbf{u}_i + \sum_{l\in\mathcal{R}(i)}\alpha(i,l)\mathbf{p}_l\right)^T \mathbf{v}_j}_{\widehat{R}_{ij}} - \underbrace{\left(\mathbf{u}_i + \sum_{l\in\mathcal{R}(i)}\alpha(i,l)\mathbf{p}_l\right)^T \mathbf{v}_k}_{\widehat{R}_{ik}}\right\} + \lambda(||\mathbf{U}||^2 + ||\mathbf{V}||^2 + ||\mathbf{P}||^2),$$



Training

- mini-batch SGD

Inference

$$\hat{R}_{ij} = \left(\mathbf{u}_i + \sum_{l \in \mathcal{R}(i)} \alpha(i,l)\mathbf{p}_l \right)^T \mathbf{v}_j.$$

Algorithm 1: Attentive Collaborative Filtering

Input: User-item interaction matrix $\mathbf{R}$. Each item l is represented by a set of component features $\{x_{l*}\}$.

Output: Latent feature matrix $\mathbf{U}$, $\mathbf{V}$, $\mathbf{P}$ and parameters in attention model Θ

Initialize $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{P}$ with Gaussian distribution. Initialize Θ with xavier [17].

repeat

  draw (i, j, k) from $\mathcal{R}_B$

  For each item l in $\mathcal{R}(i)$:

    For each component m in $\{\mathbf{x}_{l*}\}$:

      Compute $\beta(i, l, m)$ according to Eqns. (10) and (11)

    Compute $\overline{\mathbf{x}}_l$ according to Eqn. (12)

  Compute $\alpha(i, l)$ according to Eqns. (8) and (9)

  $\mathbf{u}_i' \leftarrow \mathbf{u}_i + \sum_{l \in \mathcal{R}(i)} \alpha(i,l)\mathbf{p}_l$

  $\hat{R}_{ijk} \leftarrow \mathbf{u}_i'\mathbf{v}_j - \mathbf{u}_i'\mathbf{v}_k$

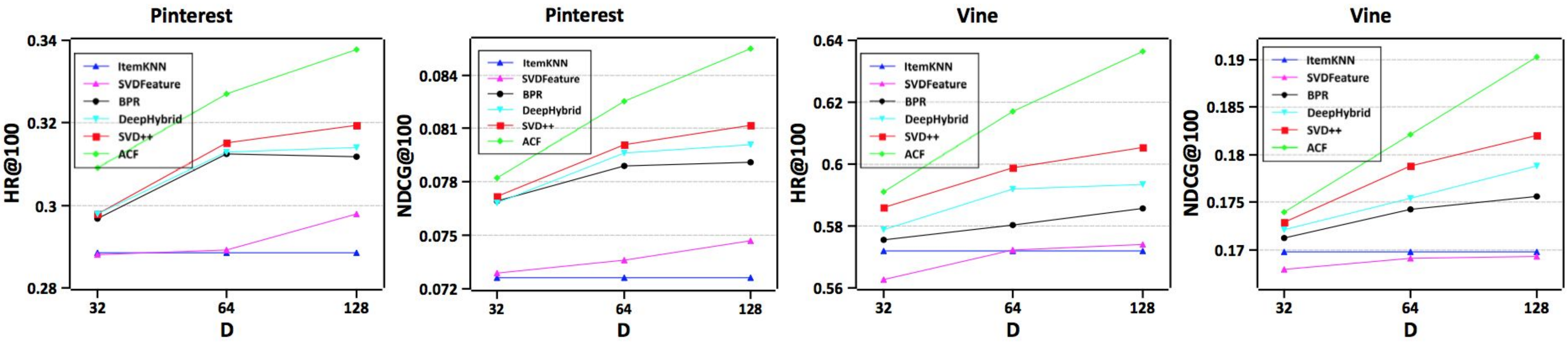
  For each parameter θ in $\{\mathbf{U}, \mathbf{V}, \mathbf{P}, \Theta\}$:

    Update $\theta \leftarrow \theta + \eta \cdot (\frac{\exp^{-\hat{R}_{ijk}}}{1+\exp^{-\hat{R}_{ijk}}} \cdot \frac{\partial \hat{R}_{ijk}}{\partial \theta} + \lambda \cdot \theta)$.

until convergence

return $\mathbf{U}$, $\mathbf{V}$, $\mathbf{P}$ and Θ.

Experiments: varying latent space dimensions



(a) Pinterest-HR@100 (b) Pinterest-NDCG@100 (c) Vine-HR@100 (d) Vine-NDCG@100

Figure 2: Performance of HR@100 and NDCG@100 w.r.t. the number of predictive factors on two datasets.

Experiments: users with sparse history

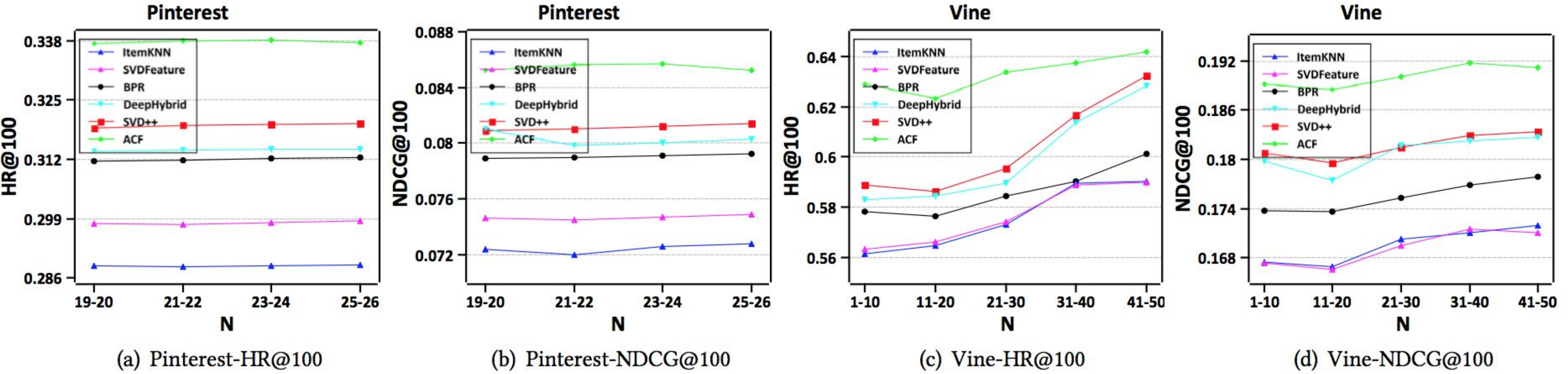


Figure 4: Performance of HR@100 and NDCG@100 w.r.t. the number of items per user on two datasets.

Experiments: effect of attention

Model	Level		Pinterest		Vine	
	Item	Comp	HR	NDCG	HR	NDCG
ACF	AVG	–	31.95%	8.12%	60.54%	18.20%
	ATT	AVG	33.21%	8.42%	62.81%	18.75%
	ATT	ATT	**33.78%***	**8.55%***	**63.65%***	**19.03%***

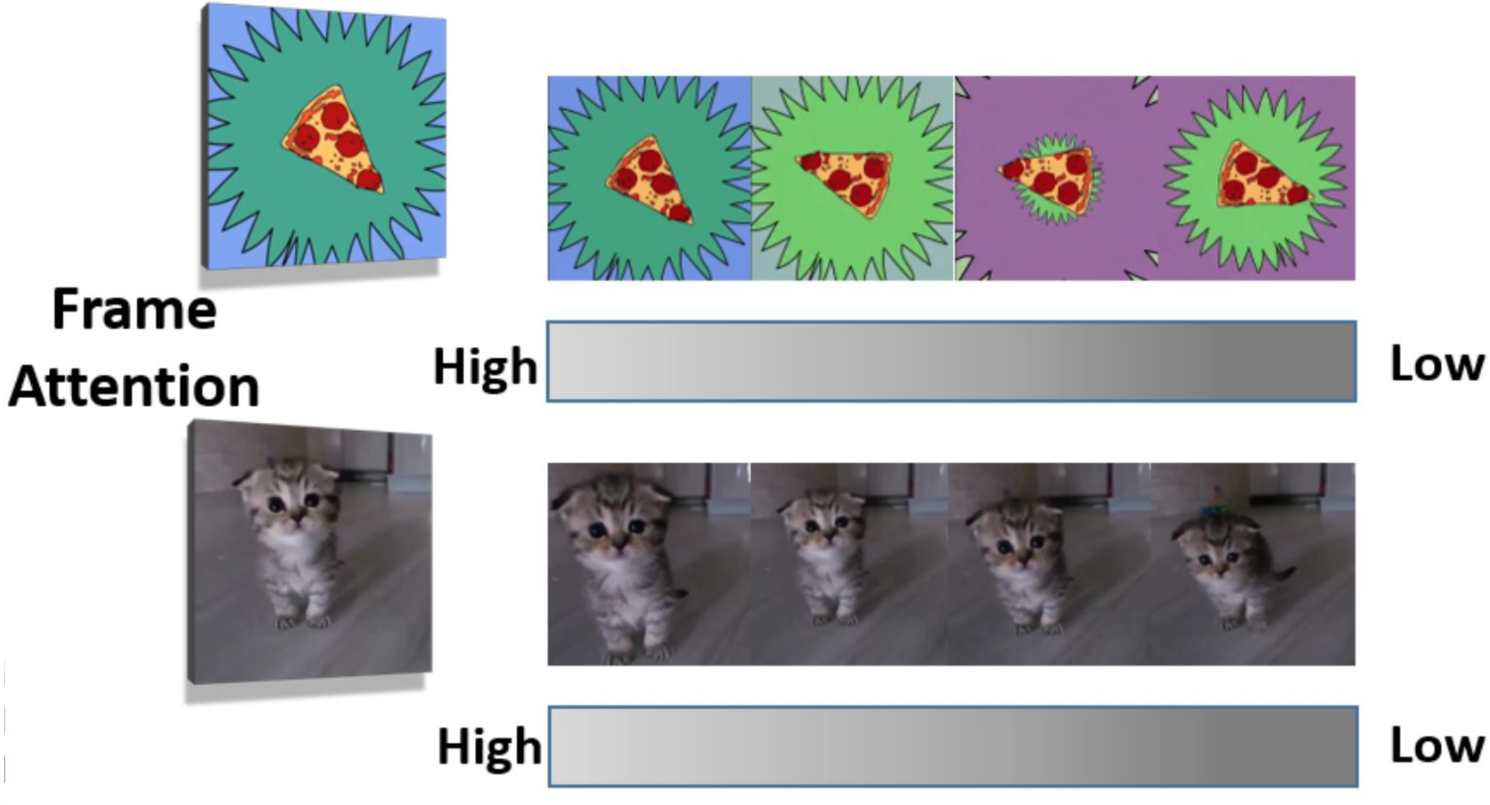
Table 2: Effect of attention mechanism on item and component (comp) level. AVG represents the average pooling strategy and ATT represents the attention mechanism. * denotes the statistical significance for $p < 0.05$.

Experiments: effect of latent parameters

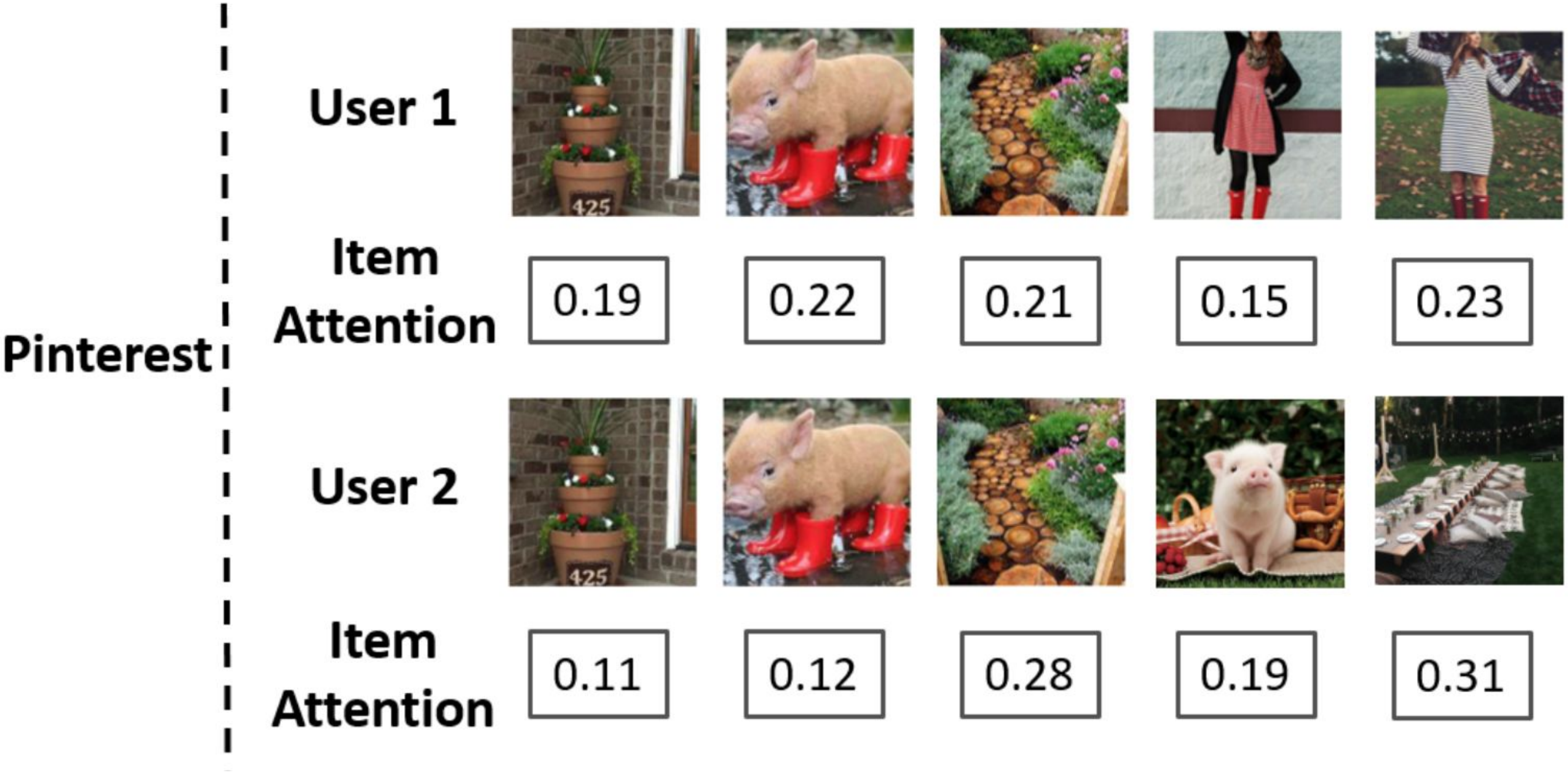
Model	Attention Type	Pinterest		Vine	
		HR	NDCG	HR	NDCG
ACF	None	31.95%	8.12%	60.54%	18.20%
	U+V	32.17%	8.31%	61.68%	18.36%
	U+P	32.69%	8.34%	62.37%	18.65%
	U+V+P	32.96%	8.32%	62.60%	18.71%
	U+V+P+X	**33.78%***	**8.55%***	**63.65%***	**19.03%***

Table 3: Effect of user, item and content attention mechanisms. U, V and P represents the user, item, and the auxiliary item information in Eqn. (5) respectively, and X indicates the content information of the item in Eqn. (8). * denotes the statistical significance for $p < 0.05$.

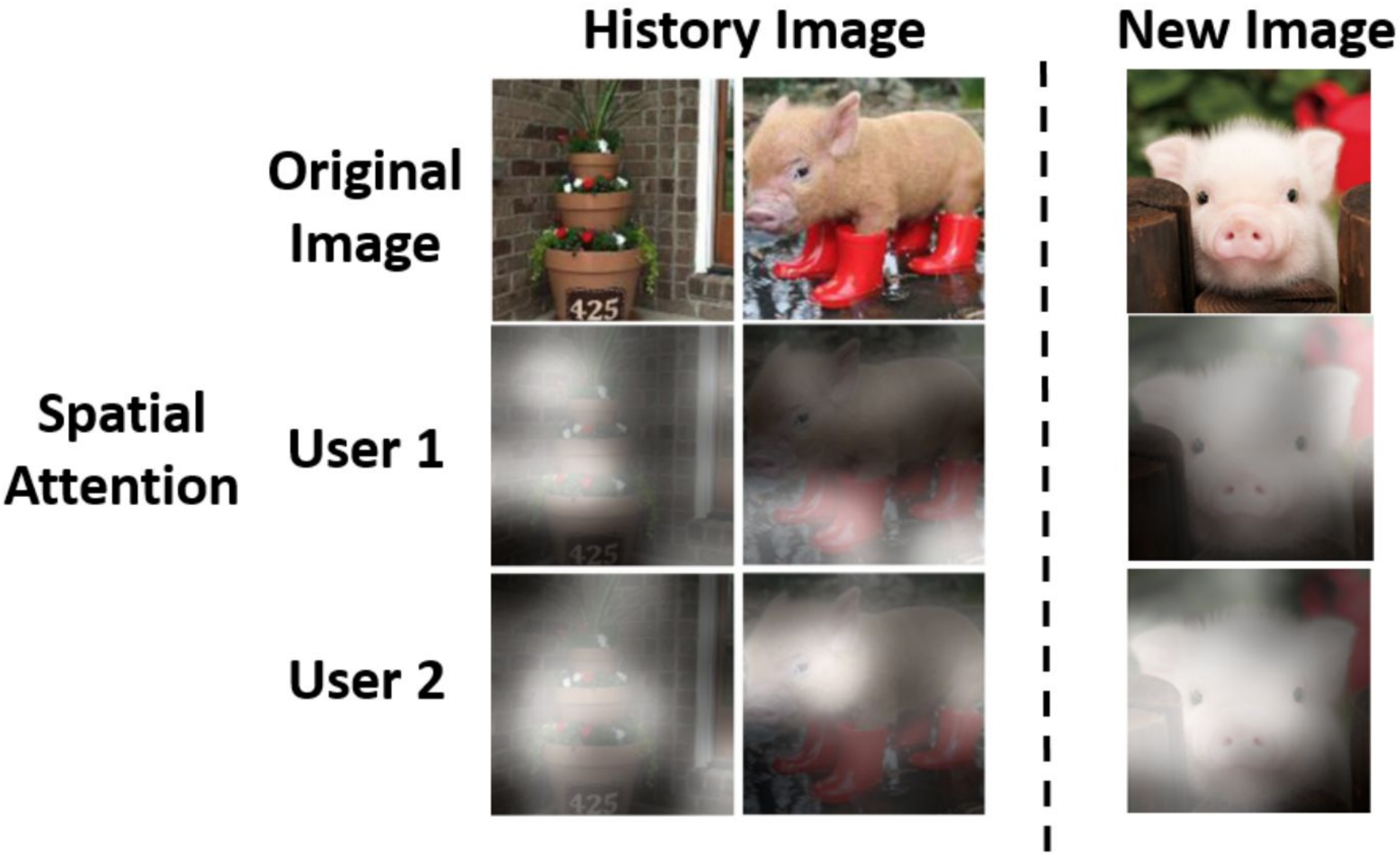
Sweet attention visualization



Sweet attention visualization



Sweet attention visualization



Hybrid recommendation: SVDFeature

- Latent factor model + contextual features
- Rating is a linear model of features + dot product of latent vectors:

$$\widehat{R}_{ij}(x,y,z) = \left(\sum_{k=1}^{m} b_k^{(i)} x_k + \sum_{k=1}^{n} b_k^{(j)} y_k + \sum_{k=1}^{s} b_k^{(o)} z_k\right) + \left(\sum_{k=1}^{m} x_k \mathbf{u}_\mathbf{k}^{(\mathbf{i})}\right)^T \left(\sum_{k=1}^{n} y_k \mathbf{v}_\mathbf{k}^{(\mathbf{j})}\right)$$

model parameters: $\Theta = (b^{(i)}, b^{(j)}, b^{(o)}, \mathbf{U}, \mathbf{V})$

$\mathbf{u}_\mathbf{k}^{(\mathbf{i})} \in \mathbb{R}^d, \mathbf{v}_\mathbf{k}^{(\mathbf{j})} \in \mathbb{R}^d -$ latent vectors

$x \in \mathbb{R}^m -$ user's properties ,
$y \in \mathbb{R}^n -$ item's properties,
$z \in \mathbb{R}^n -$ other context

Hybrid recommendation: DeepHydrid

- Idea:
  - Use matrix factorization technique to learn latent embeddings
  - Regress contextual data to item latent embeddings for rare items
- In original paper:
  - Weighted matrix factorization:

$$\min_{x_\star, y_\star} \sum_{u,i} c_{ui}(p_{ui} - x_u^T y_i)^2 + \lambda \left(\sum_u ||x_u||^2 + \sum_i ||y_i||^2 \right)$$

$$p_{ui} = I(r_{ui} > 0), \quad c_{ui} = 1 + \alpha \log(1 + \epsilon^{-1} r_{ui})$$

  - Regression of audio content with CNN

Attentive Collaborative Filtering: recap

- Contextual information modeling + latent factor model
- Jointly learnt item and user representation with two-level attention
- Component level attention highlights content interesting for user
- Item-level attention contributes most to the quality of recommendation
- Nice model interpretability

References

1) Chen, J., Zhang, H., He, X. and Nie, L. (2017). *Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention.*

2) SVD++: Yehuda Koren. (2008). *Factorization Meets the Neighborhood: a Multifaceted Collaborative Filtering Model*

3) SVDFeature: Chen T. (2012). *SVDFeature: A Toolkit for Feature-based Collaborative Filtering*

4) DeepHybrid: A. van den Oord (2013). *Deep content-based music recommendation*

Questions?