Mutually-Regularized Dual Collaborative Variational Auto-encoder for Recommendation Systems

Authors: Yaochen Zhu[1,2] and Zhenzhong Chen[1]

1. WHU-RS Institute of Artificial Intelligence and Machine Vision
2. Media Lab, Tencent

To be presented at TheWebConf 2022, Virtual, April 29, 2022

Background Knowledge

■ What is it like for an offline recommender system?

Fig. 1 User-item interaction matrix

An offline recommender system is just like

1. Matter is __________ you can see and touch.
2. ________ and _______ take the shape of a container.
3. _______ can easily be held.
4. ________ can be poured.
5. Gasses have ____ of space between particles.
6. Solids have particles that are _____ together.
7. Liquids have particles that _____ past each other.
8. A _____ is an example of a solid.
9. _____ is an example of a gas.
10. _____ is an example of a liquid.

Fig. 2 Filling-in Blanks

Different recommenders provide different solutions to the same missing value prediction problem.

Background Knowledge

■ Solution 1. matrix factorization (MF)

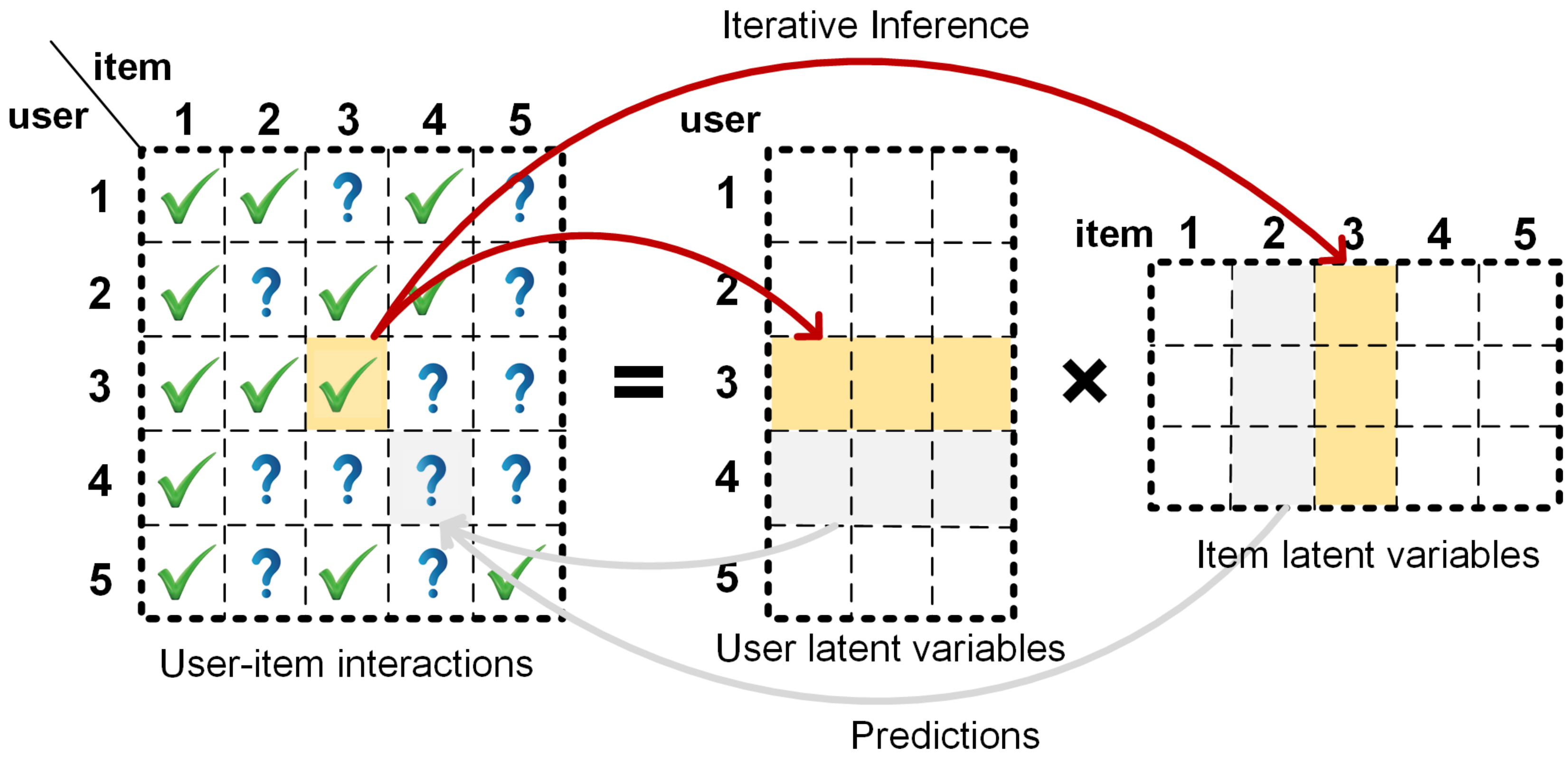


Fig. 3 Matrix factorization-based recommender systems

Item 1

Item 4

Item 2

Item 5

Item 3

Linearly factorizing user-item interaction matrix into the user and item parts.

Background Knowledge

■ Pros and cons of matrix factorization (MF)

➢ **Pro:** Efficient to fuse U/I side-information

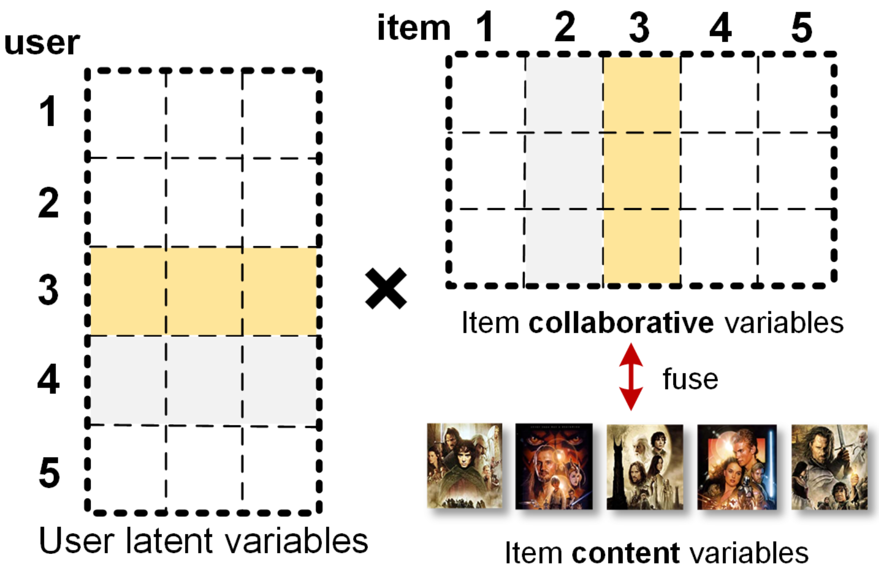


Fig. 4 Pros of MF-based recommenders

E.g., Through Introducing mutual regularization between user/item collaborative variables and content embeddings.

➢ **Con:** Inefficient to fold-in new users

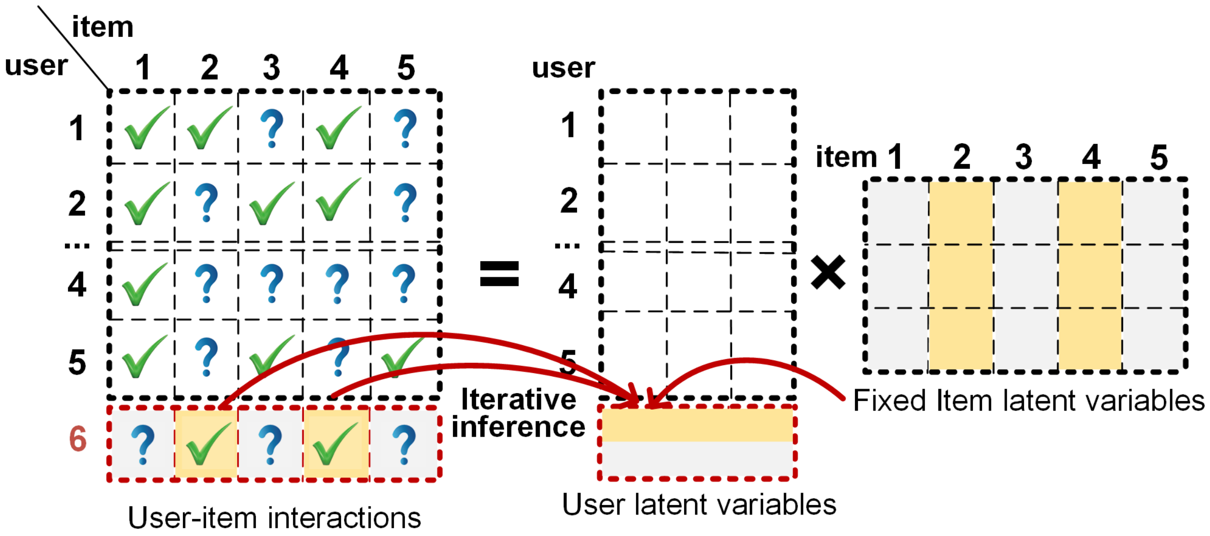


Fig. 5 Cons of MF-based recommenders

Note: Fold-in means inferring user latent variable and making recommendations for new users with interactions

Background Knowledge

- Solution 2. user-oriented auto-encoders (UAE)

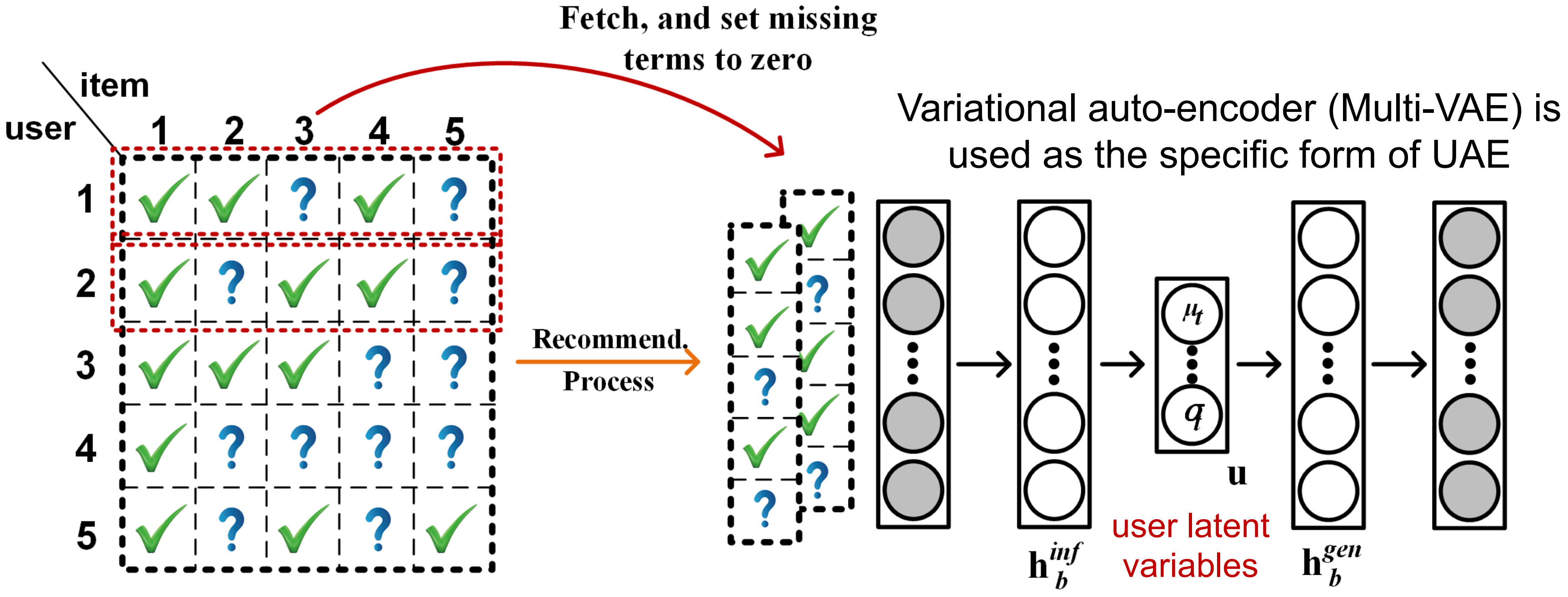


Fig. 6 User-oriented auto-encoder-based recommender systems

Take user interactions, embed into user latent variables, and from them direct reconstruct ratings.

Background Knowledge

- Pros and cons of user-oriented auto-encoders (UAE)

➢ **Pro:** Efficient to fold-in new users

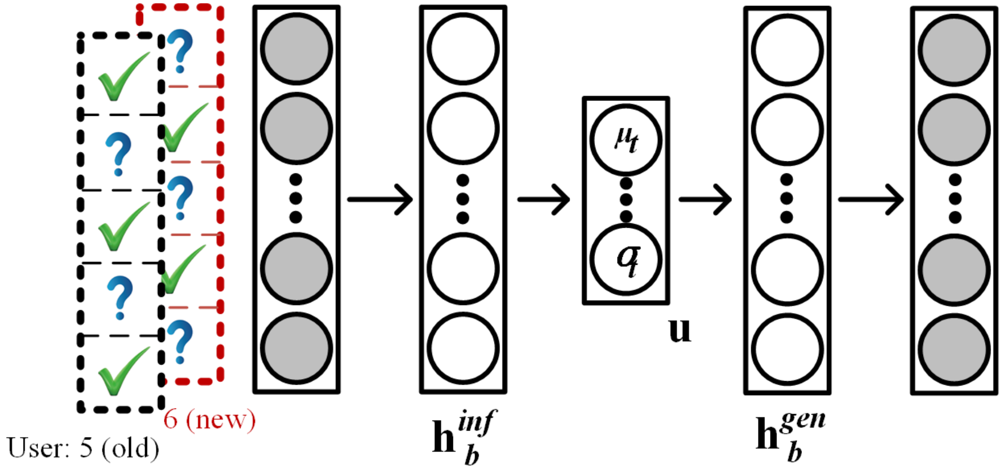


Fig. 7 Pros of UAE-based recommenders

Folding-in new users needs only one forward propagation of the UAE network.

➢ **Con:** **“impossible”** to fuse item information

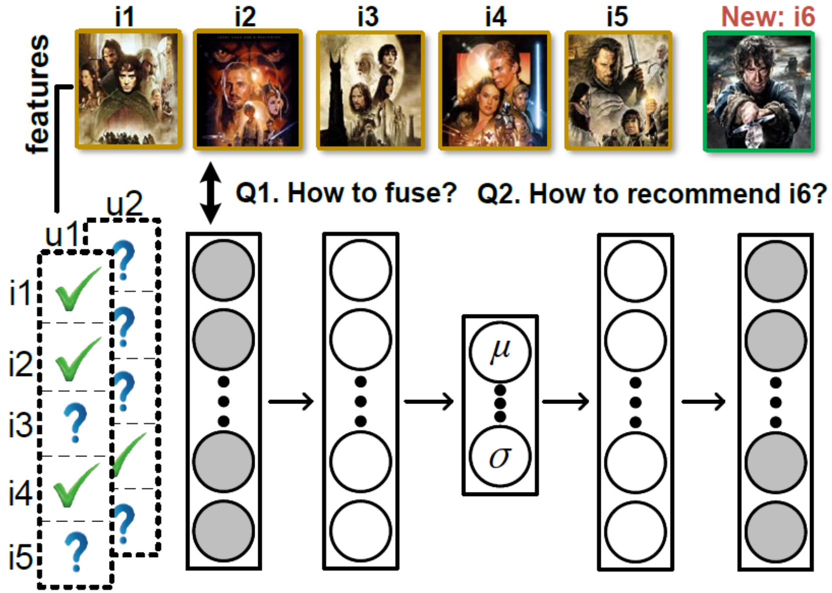


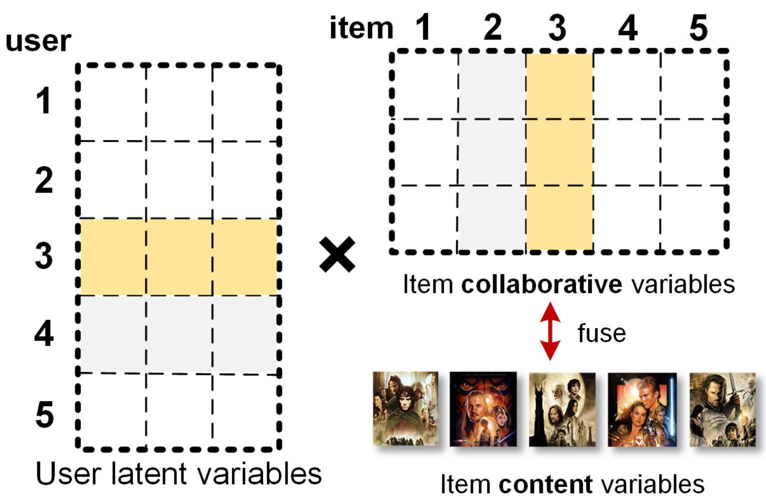
Fig. 8 Cons of UAE-based recommenders

Because: No latent item variables is considered.

Challenges and Motivations

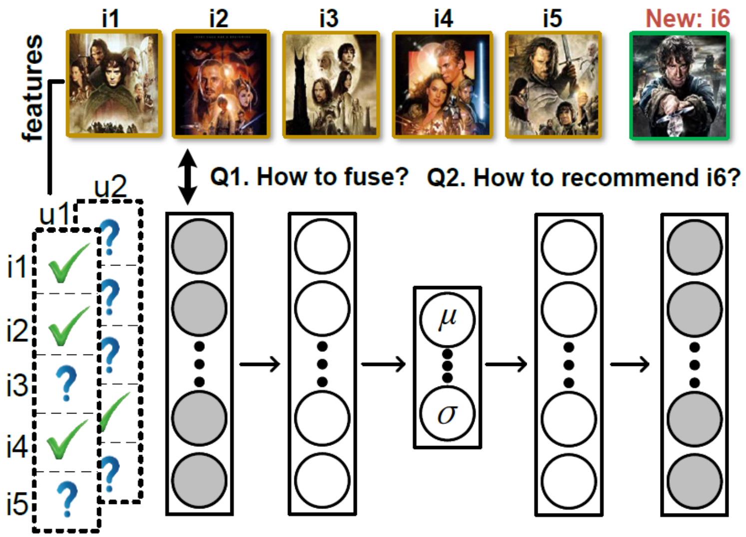
■ Summary of background knowledge

➢ **MF-based recommender system**



- Efficient to fuse side-information
- Inefficient to fold-in new users

➢ **UAE-based recommender system**



- Efficient to fold-in new users
- Inefficient to fuse item side-information

Can we combine the advantage of both worlds?

Main Contribution of MD-CVAE

- The contribution can be summarized into three folds as:

- **We have made a key observation** that**:**

  the first & last layer weights of UAE = latent item embeddings

- **We introduce a mutual regularization trick** that makes:

  item content embedding ≈ Multi-VAE weights

- **We design a symmetric inference strategy** that allows

  cold-start item recommendations without model retraining

Methodology: Mutually-Regularized CVAE

■ Generation Process of MD-CVAE

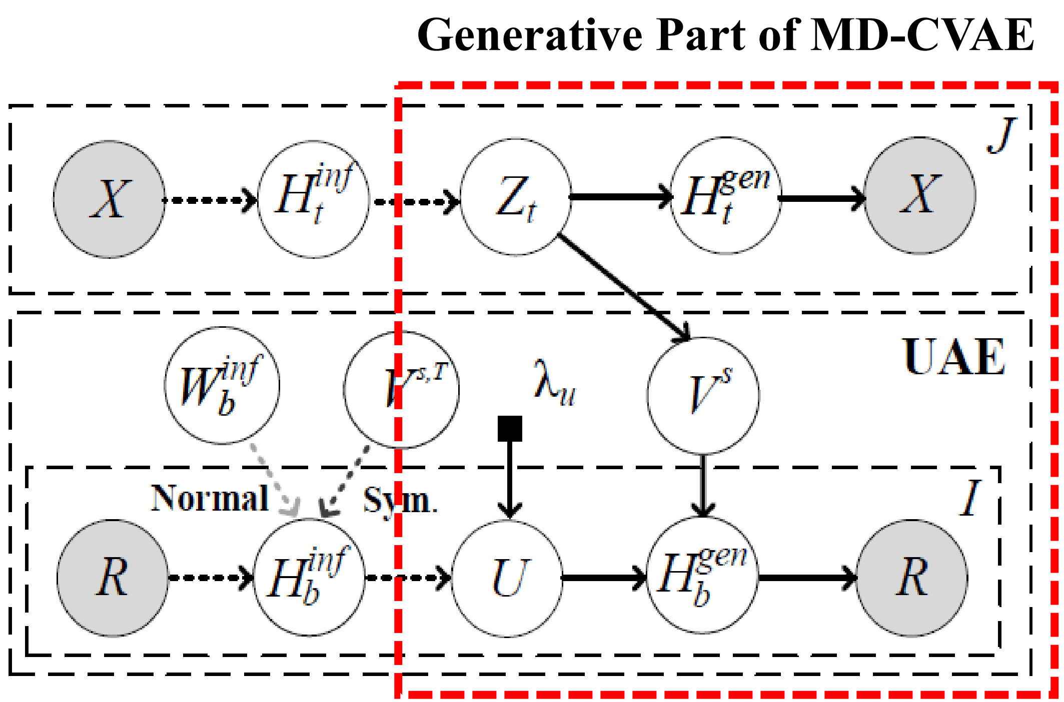


Fig. 9 PGM of MD-CVAE (Generative)

- Draw latent variable $\mathbf{u}$ as Multi-VAE from

$$\mathbf{u} \sim \mathcal{N}\left(\mathbf{0}, \lambda_u^{-1} \mathbf{I}_{K_u}\right)$$

- In addition, we draw latent item collaborative variable $\mathbf{z}_b$ and content variable $\mathbf{z}_t$ as:

$$\mathbf{z}_b \sim \mathcal{N}\left(\mathbf{0}, \lambda_v^{-1} \mathbf{I}_{K_v}\right), \quad \mathbf{z}_t \sim \mathcal{N}\left(\mathbf{0}, \mathbf{I}_{K_v}\right)$$

- We set the item latent variable $\mathbf{v}$ as:

$$\mathbf{v} = \mathbf{z}_b + \mathbf{z}_t$$

conditional on $\mathbf{z}_t$,$\mathbf{v}$ follows $\mathcal{N}\left(\mathbf{z}_t, \lambda_v^{-1} \mathbf{I}_{K_v}\right)$

**which is the key to tightly couple z_t and v*

The details for tightly coupling will be discussed later when introducing optimization.

Methodology: Mutually-Regularized CVAE

■ Generation Process of MD-CVAE

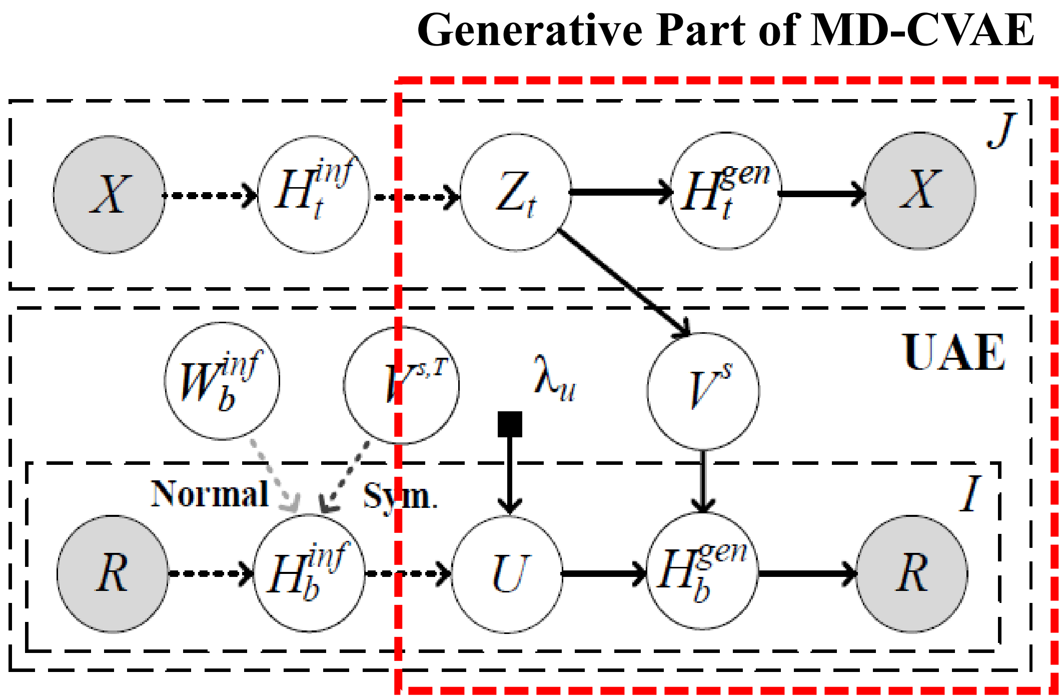


Fig. 9 PGM of MD-CVAE (Generative)

- Use an $N-1$ layer $\mathrm{MLP}_{u,gen}: \mathbb{R}^{K_u} \to \mathbb{R}^{K_v}$ on $\mathbf{u}$

$$\mathbf{h}_b^{gen}(\mathbf{u}) = MLP_{u,gen}(\mathbf{u});$$

- Def. $\mathbf{V}^s = [\mathbf{v}_1^T, \mathbf{v}_2^T, \cdots, \mathbf{v}_J^T]$, and draw r from

$$\mathbf{r} \sim Multi\left(softmax\left(\mathbf{V}^s \cdot \mathbf{h}_b^{gen}(\mathbf{u})\right), \#Int\right)$$

which *defs* an N-layer MLP with $MLP_{u,gen}(\mathbf{u});$

- Generate item content feature $\mathbf{x}$ from $\mathbf{z}_t$.

The gen. model of MD-CVAE: $p_{\boldsymbol{\theta}}(R, X^s, U, V^s, Z_t^s) = p_{\boldsymbol{\theta}_r, V^s}(R \mid U) \cdot P_{\boldsymbol{\theta}_x}(X^s \mid Z_t^s) \cdot p(V^s \mid Z_t^s) \cdot p(Z_t^s) \cdot p(U)$

Methodology: Mutually-Regularized CVAE

■ Difference between MD-CVAE and Multi-VAE decoder

➢ The vanilla Multi-VAE:

- Its decoder can be formulated as :

$$\mathbf{r} \sim Multi\left(softmax\left(W \cdot \mathbf{h}_b^{gen}(\mathbf{u})\right), \#Int\right)$$

where:

- ☐ W is the randomly initialized last layer weight
- ☐ No item content information is fused.
- ☐ Bad for sparse or cold-start items

➢ The proposed MD-CVAE

- Its decoder can be formulated as :

$$\mathbf{r} \sim Multi\left(softmax\left(\mathbf{V}^s \cdot \mathbf{h}_b^{gen}(\mathbf{u})\right), \#Int\right)$$

where:

- ☐ $\mathbf{V}^s$ is the stacked latent item variables
- ☐ $\mathbf{v}$ is tightly coupled with $\mathbf{z}_t$ via $\mathbf{v} = \mathbf{z}_b + \mathbf{z}_t$
- ☐ Item content info. can be used for rec.

In summary, MD-CVAE principledly unifies Multi-VAE and item content module.

Methodology: Mutually-Regularized CVAE

■ Inferential Process

➢ The variational posterior of the latent variables U, V^s and Z_t^s :

$$q_\phi(U, V^s, Z_t^s, \mid R, X^s) = q_\phi(U \mid R) \cdot q_\phi(Z_t^s \mid X^s) \cdot q(V^s \mid Z_t^s),$$

UAE encoder

Item content encoder

Gaussian with fixed var. that couples UAE and item VAE

➢ Optimization: Evidence Lower Bound (ELBO)

$$\begin{aligned}\mathcal{L} &= \mathrm{E}_{q_\phi}\left[\log p_0(R, X^s, U, V^s, Z_t^s) - \log q_\phi(U, V^s, Z_t^s \mid R, X^s)\right] \\ &= \mathrm{E}_{q_\phi}[\log p_0(R \mid U) + \log p(V^s \mid Z_t^s) + \log p_\theta(X^s \mid Z_t^s)] \\ &\quad - \mathrm{KL}\left(q_\phi(Z_t^s \mid X^s) \parallel p(Z_t^s)\right) - \mathrm{KL}\left(q_\phi(U \mid R) \parallel p(U)\right) + C\end{aligned}$$

The entropy of $q(V^s \mid Z_t^s)$ is constant due to its fixed variance.

Methodology: Mutually-Regularized CVAE

■ EM Like Optimization

➢ UAE part of MD-CVAE: b-step optimization function

$$\mathcal{L}_{b\_step}^{MAP} = E_{q_\phi(U|R)}[\log p_\theta(R \mid U)] - \frac{\lambda_v}{2} \cdot \| V^s - \hat{Z}_t^s \|_F^2$$
$$-KL\left(q_\phi(U \mid R) \,\|\, p(U)\right) - \frac{\lambda_W}{2} \cdot \sum_l \| \mathbf{W}_b^{(l)} \|_F^2$$

➢ Dual item content VAE: t-step optimization function

$$\mathcal{L}_{t\_step}^{MAP} = E_{q_\phi(Z_t|X)}\left[\log p_\theta(X \mid Z_t) - \frac{\lambda_v}{2} \cdot \| \hat{V} - Z_t \|_2^2\right]$$
$$-KL\left(q_\phi(Z_t \mid X) \,\|\, p(Z_t)\right) - \frac{\lambda_W}{2} \cdot \sum_l \| \mathbf{W}_t^{(l)} \|_F^2$$

$$\mathbf{E}_{q_\phi}[\log p(V^s \mid Z_t^s)]$$

In ELBO

*which constrains $\mathbf{v}_j$ and $\mathbf{z}_{t,j}$ to be close to each other

Form: $\underbrace{\text{Expected Loglikelihood + Regularization}}_{\text{Common for all VAEs}}$ + $\underbrace{\text{Mutual regularization}}_{\text{Unique for MD-CVAE}}$

Methodology: Mutually-Regularized CVAE

■ Symmetric Inference Process

➢ The weights of the first Multi-VAE layer can also be viewed as item embeddings

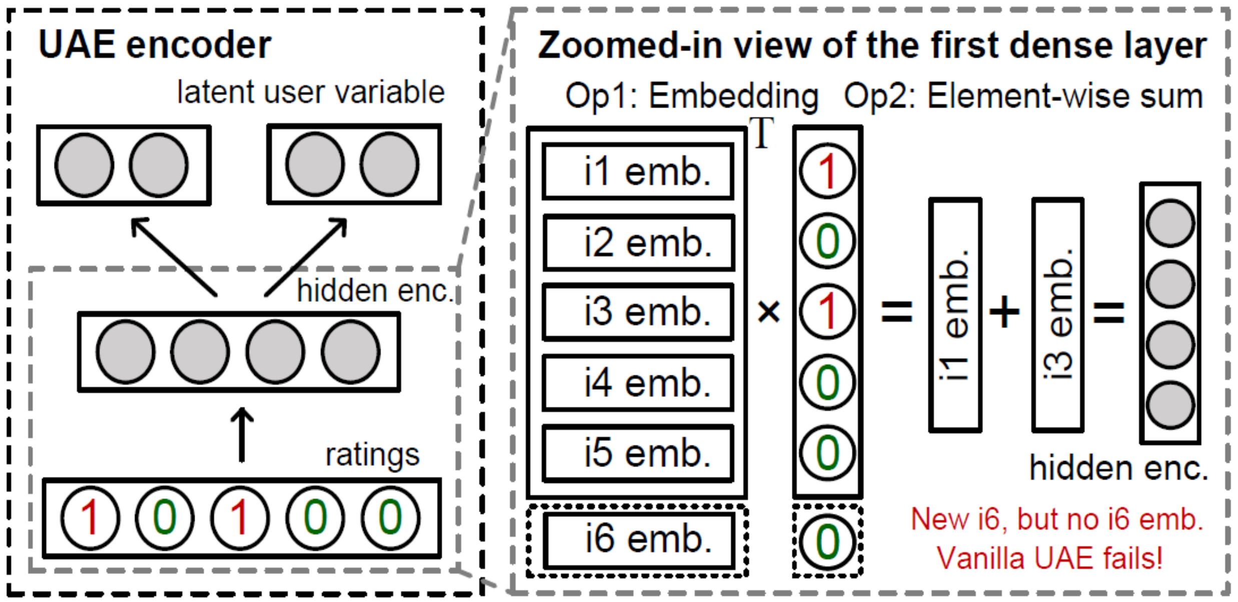


Fig. 10 Zoomed-in view of the first Multi-VAE layer

- The first dense layer:

$$\mathbf{h}_b^{inf} = \mathbf{W}_b^{inf} \cdot \mathbf{r} = \sum_j \mathbb{I}(r_j = 1) \cdot \mathbf{w}_{b,j}^{inf}$$

Op1. embedding   Op2. element-wise sum

- For mutual regularization, we can

reuse $\mathbf{V}$ in decoder by setting $\mathbf{W}_b^{inf} = \mathbf{V}^{s,T}$

which we name as **MDsym-CVAE**

Methodology: Mutually-Regularized CVAE

■ Prediction for normal items

➢ This can be done by using only the UAE part of MD-CVAE

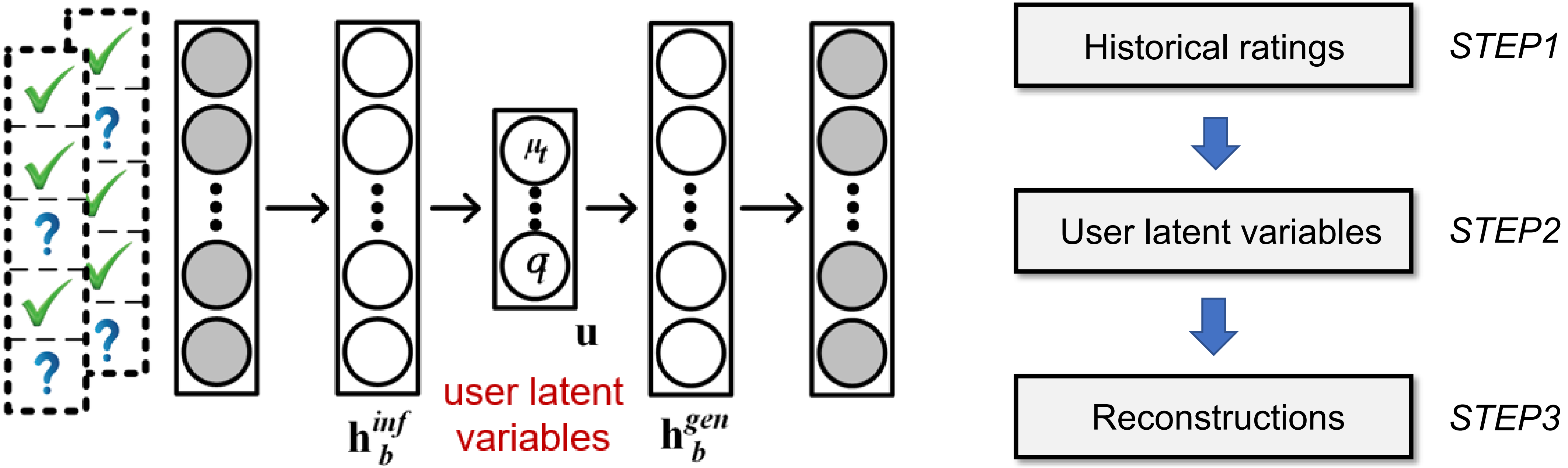


Fig. 11 Recommendation of MD-CVAE procedure for normal items

Highly efficient where three steps can be done with a single forward propagation

Methodology: Mutually-Regularized CVAE

■ Prediction for cold-start items

➢ What happens if cold-start items exist for Multi-VAE?

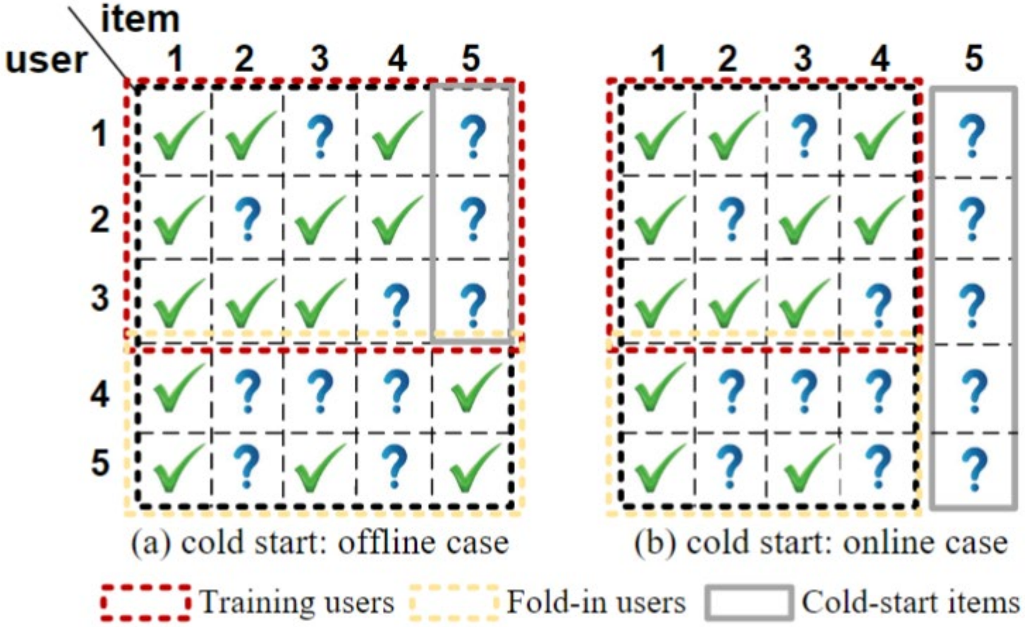


Fig. 12 Two cases of cold-start rec.

□ The offline case:

- Cold-start items are **mixed with training samples**
- Corresponding weights will never be updated
- Inferred user embeddings may contain **randomness**

□ The online case:

- Cold-start items exist **after training**
- **Dimensional mismatch** of first/last layer of Multi-VAE
- 4 expected but 5 actually inputs/outputs (Fig. 12)

Methodology: Mutually-Regularized CVAE

■ Prediction for cold-start items

➢ How does MD-CVAE and MDsym-CVAE recommend cold-start items?

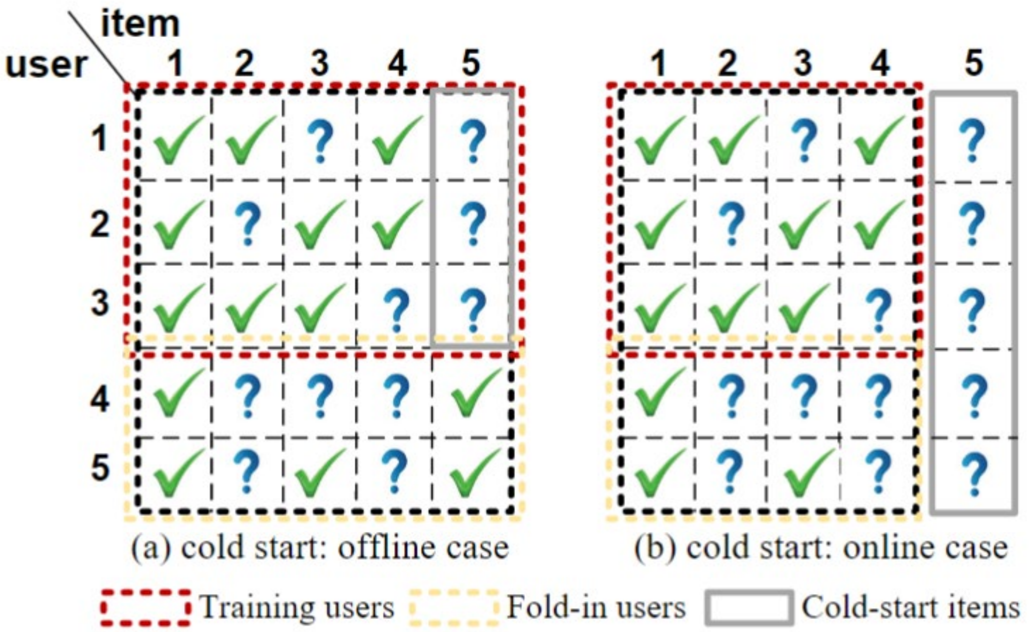


Fig. 12 Two cases of cold-start rec.

☐ The offline case:

- Mutual regularization to introduce content info.

☐ The online case (MDsym-CVAE only):

- Denote the old-weights as $\mathbf{V}^{s}_{old} \in \mathbb{R}^{J \times K_v}$
- Denote content embeddings for cold-start items as:

$$\mathbf{V}^{s}_{new} = \left[\boldsymbol{\mu}^{T}_{\mathbf{z}_{t},J+1}, \boldsymbol{\mu}^{T}_{\mathbf{z}_{t},J+2}, \cdots, \boldsymbol{\mu}^{T}_{\mathbf{z}_{t},J+J'}\right] \in \mathbb{R}^{J' \times K_v}$$

- Calculate the new weights as: $\mathbf{V}^{s}_{surr} = [\mathbf{V}^{s}_{old} || \mathbf{V}^{s}_{new}]$
- Rec. can be made the same as normal items.

Experimental Results

■ Datasets and evaluation metrics

➢ Dataset descriptions

dataset	#users	#items	%density	#features
citeulike-a	5,551	16,980	0.217%	8,000
movielen-sub	10,881	7,701	0.922%	8,000
toys & games	14,706	11,722	0.072%	8,000

➢ Item textual features:

- ☐ *cituelike-a :* title + abstract
- ☐ *movielen-sub* : plot from IMDB
- ☐ *toys & games* : user reviews

➢ Evaluation metrics

Three metrics utilized in the paper are: Recall@20, Recall@40, NDCG@100

Codes and datasets are released at: https://github.com/yaochenzhu/MD-CVAE

Experimental Results

- Normal item recommendation

➢ Comparison with state-of-the-art algorithms

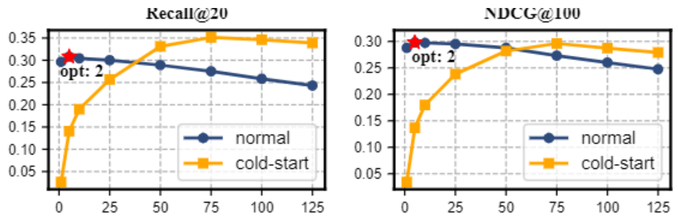
	citeulike-a			movielen-sub			toys & games		
	Recall@20	Recall@40	NDCG@100	Recall@20	Recall@40	NDCG@100	Recall@20	Recall@40	NDCG@100
MD-CVAE	**0.303**	**0.377**	**0.301**	**0.353**	**0.452**	**0.381**	0.141	0.188	0.102
MDsym-CVAE	0.295	0.374	0.297	0.347	0.449	0.377	**0.147**	**0.191**	**0.106**
FM	0.231	0.312	0.238	0.324	0.421	0.357	0.088	0.121	0.062
CTR	0.169	0.250	0.190	0.285	0.398	0.312	0.124	0.179	0.089
CDL	0.209	0.295	0.226	0.311	0.405	0.339	0.133	0.181	0.092
CVAE	0.236	0.334	0.247	0.304	0.422	0.355	0.139	0.188	0.094
Multi-VAE	0.269	0.346	0.274	0.326	0.423	0.357	0.114	0.157	0.082
CoVAE	0.247	0.338	0.260	0.338	0.436	0.367	0.120	0.174	0.085
CondVAE	0.274	0.359	0.275	0.341	0.437	0.365	0.132	0.180	0.094
DICER	0.279	0.363	0.277	0.329	0.428	0.359	0.127	0.172	0.092
DAVE	0.281	0.362	0.283	0.340	0.432	0.371	0.125	0.177	0.086

MD-CVAE improves over MF-based tightly-coupled and Multi-VAE-based recommenders

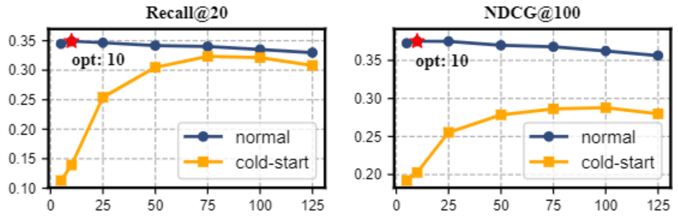
Experimental Results

■ Cold-start item recommendations

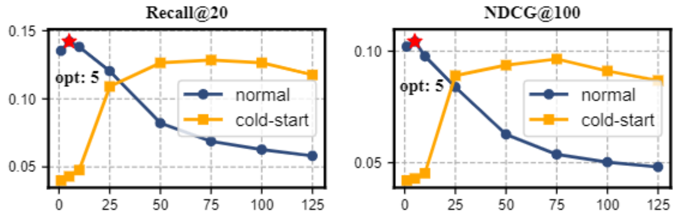
➢ Discussion of hyper-parameter



(a) *citeulike-a* dataset



(b) *movielen-sub* dataset



(c) *toys & games* dataset

➢ Comparisons with SOTA methods

(a) citeulike-a

	Recall@20 (NI / CI)	NDCG@100 (NI / CI)
MDsym-CVAE	**0.290** / **0.251**	**0.286** / **0.249**
CTR	0.169 / 0.209	0.189 / 0.207
CDL	0.206 / 0.218	0.203 / 0.214
CVAE	0.238 / 0.235	0.236 / 0.232

(b) movielen-sub

	Recall@20 (NI / CI)	NDCG@100 (NI / CI)
MDsym-CVAE	**0.351** / **0.309**	**0.369** / **0.275**
CTR	0.284 / 0.164	0.310 / 0.195
CDL	0.301 / 0.196	0.338 / 0.227
CVAE	0.318 / 0.279	0.342 / 0.240

(c) toys & games

	Recall@20 (NI / CI)	NDCG@100 (NI / CI)
MDsym-CVAE	**0.113** / **0.099**	**0.084** / **0.082**
CTR	0.089 / 0.084	0.072 / 0.069
CDL	0.096 / 0.090	0.077 / 0.079
CVAE	0.110 / 0.094	0.081 / 0.080

MDsym-CVAE improves over tightly-coupled methods for cold-start recommendations

Summary & Conclusions

- Three conclusions can be drawn for MD-CVAE as follows:

  - MD-CVAE combines the *pros* of both *MF*-based and *UAE*-based recommenders;
  - Specifically, MD-CVAE fuse *item content information* by mutual regularization while maintaining the high efficiency of Multi-VAE to *fold-in new users*;
  - Finally, with a symmetric inference strategy, MDsym-CVAE is competitive as tightly-coupled hybrid recommender for *cold-start item recommendations*.

Thank you for listening!

Codes & Dataset

WHU-RS Institute of Artificial Intelligence and Machine Vision, Wuhan University

Media Lab, Tencent