User-Item Matching and Auto-encoders for Cold-start Next-item Recommendation

Hanrui Wu†, Chung Wang Wong†, Yuguang Yan†, Dahai Yu‡, Michael K. Ng†

†Department of Mathematics, The University of Hong Kong ‡TCL Corporate Research Hong Kong

Problem

Cold-start next-item recommendation (CSNIR), i.e., how to recommend a user his/her next preferred item, which has not yet been interacted with by any users within the system.

Introduction

We propose a novel based on zero-shot learning (ZSL), namely User-Item Matching and Auto-encoders (UIMA). Motivations are as follows:

- In [1], Li et al. verified that utilizing zero-shot learning can boost the cold-start recommendation performance.
- Existing algorithms seek to discover a latent space to bridge the users and cold-start items, like [3, 2]. However, these methods barely preserve either the interaction relationship between users and their historical selected items or the information of original input data.

Based on the above, we crystallize the relationship and setting from ZSL to CSNIR. Then, we exploit a hypergraph to learn user embedding based on the relationship between users and historical items. Besides, we use a multi-layer perceptron (MLP) network to learn item embedding. As a result, UIMA preserves the relationship between users and historical items and the information of original items.

ZSL vs. CSNIR

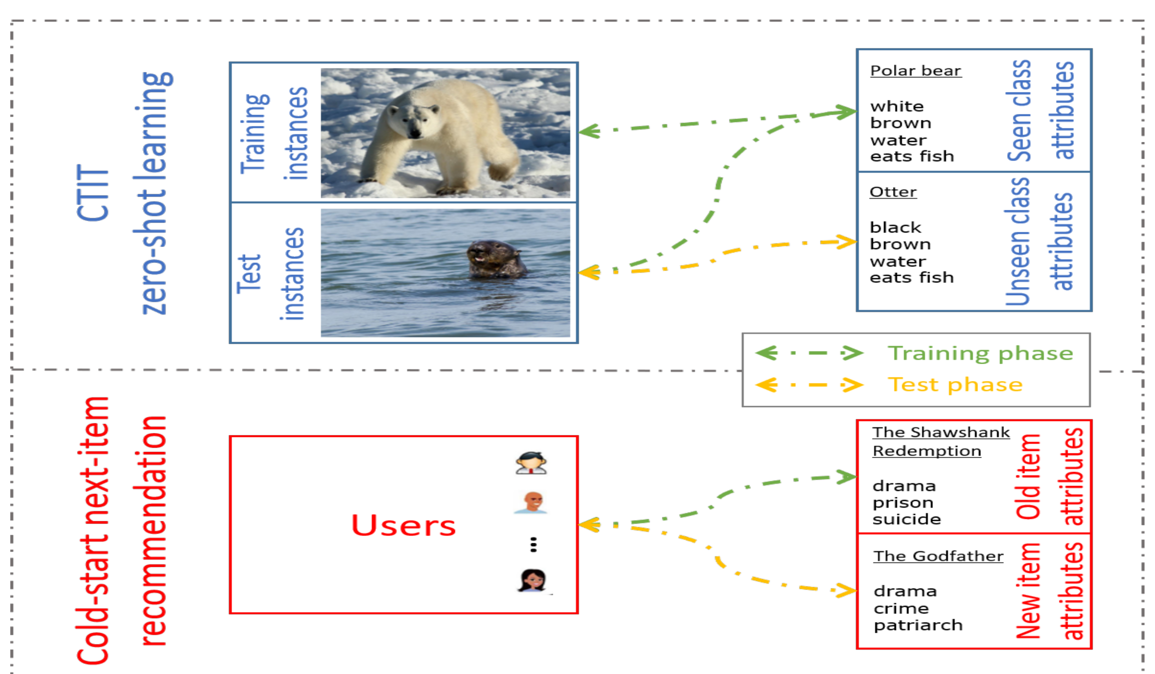

Fig. 1: Relationship between ZSL and CSNIR

- User (sample) and item (class) are given in advance.
- Class-item: seen classes in ZSL corresponding to warm items in CSNIR, unseen classes corresponding to new items.
- Sample-user: training and test samples in ZSL corresponding to users in CSNIR.

The Proposed Model

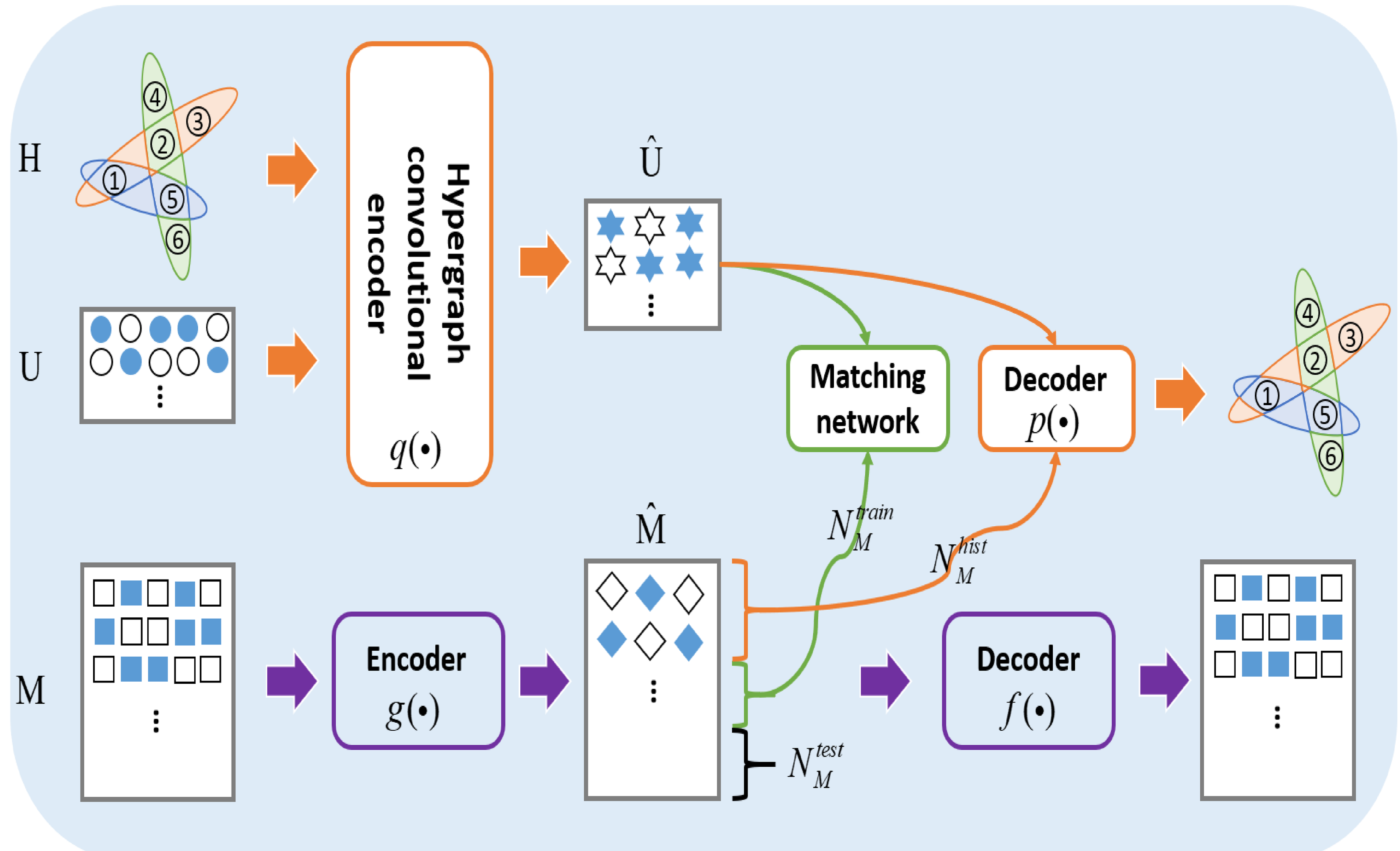

Fig. 2: The proposed network architecture

1. Item auto-encoder (in purple): learning item embedding while preserving original data information based on MLP:
$$(g, f) = \arg\min_{g,f} \mathcal{L}_{\mathrm{M}}(\mathbf{M}, f(g(\mathbf{M}))),$$
where $\mathbf{M}$ represents item (warm and cold-start) initial features and $g(\cdot)$ and $f(\cdot)$ mean item encoder and decoder, respectively.
2. User auto-encoder (in orange): learning user embedding while preserving user historical interaction information based on hypergraph convolutional network:
$$(q, p) = \arg\min_{q,p} \mathcal{L}_{\mathrm{U}}(\mathbf{H}, p(\hat{\mathbf{H}}|q(\hat{\mathbf{U}}|\mathbf{U}, \mathbf{H}), \hat{\mathbf{M}}^{hist})),$$
where $\mathbf{H}$ means user historical interactions, $\mathbf{U}$ is user initial features, $\hat{\mathbf{M}}^{hist}$ is warm item embedding learned by item auto-encoder, $\hat{\mathbf{H}}$ is the reconstructed hypergraph, and $q(\cdot)$ and $p(\cdot)$ are user encoder and decoder, respectively.
3. Matching network (in green): matching user and next-item embeddings:
$$(q, g) = \arg\min_{q,g} \mathcal{L}_{\mathrm{UM}}(q(\hat{\mathbf{U}}|\mathbf{U}, \mathbf{H}), \mathbf{Y}g(\mathbf{M}^{train})),$$
where $\mathbf{Y}$ represents the training next-item matrix and $\mathbf{M}^{train}$ is the initial features of next items.

Overall objective function:
$$\mathcal{L}(g, f, q, p) = \mathcal{L}_{\mathrm{UM}} + \gamma\mathcal{L}_{\mathrm{M}} + \rho\mathcal{L}_{\mathrm{U}}, \quad (1)$$
where γ and ρ are parameters. By solving Problem (1), we obtain user embedding $\hat{\mathbf{U}}$ and cold-start item embedding $\hat{\mathbf{M}}^{test}$. Then, we compute the Euclidean distance between each user and cold-start items to recommend the nearest item.

Experiment

TCL dataset:

- Collects users' video play log from Dec. 29, 2020 to Mar. 1, 2021. We treat 6 days as one week and obtain data for 10 weeks.
- Keeps videos whose channel is "电影" and keeps users whose weekly play count < 200 and play rate $> 60\%$ to build learning tasks.
- We build 6 tasks in total, each task contains 5 weeks' data, where the first and second weeks are used as historical data, the third week is treated as training label data, the fourth week is regarded as validation data, and the fifth week is adopted as test data.

Average results on the 6 tasks:

Method	MRR (↑)	Top@5 (↓)	Top@10 (↓)	Top@15 (↓)	Top@20 (↓)
KNN	0.0480	0.9447	0.9059	0.8705	0.8397
PCA	0.0180	0.9907	0.9841	0.9747	0.9661
LCE	0.0342	0.9703	0.9554	0.9438	0.9033
LLAE	0.0727	0.9205	0.8847	0.8480	0.8282
UIMA-I	0.1182	0.8493	0.7896	**0.7404**	0.7182
UIMA-II	**0.1490**	**0.8129**	**0.7737**	0.7440	**0.7041**
Improvement	104.95%	135.35%	96.27%	70.79%	72.24%

Conclusion & Future Work

- Compared with baseline methods, UIMA obtains better results on the TCL dataset, indicating the effectiveness of the proposed mode.
- In the future, we will study the collaborative convolutional operations between nodes and hyperedges in hypergraphs, further improving the recommendation performance.
- Consider recommending multiple items in the future work.

Acknowledgement

HKU-TCL Joint Research Centre for Artificial Intelligence, Hong Kong

Reference

[1] Jingjing Li et al. "From zero-shot learning to cold-start recommendation". In: *AAAI*. Vol. 33. 01. 2019, pp. 4189–4196.

[2] Yitong Meng et al. "Wasserstein Collaborative Filtering for Item Cold-start Recommendation". In: *ACM UMAP*. 2020, pp. 318–322.

[3] Martin Saveski and Amin Mantrach. "Item cold-start recommendations: learning local collective embeddings". In: *ACM RecSys*. 2014, pp. 89–96.