Unifying Knowledge Graph Learning and Recommendation: Towards a Better Understanding of User Preferences

Yixin Cao
National University of Singapore
caoyixin2011@gmail.com

Xiang Wang
National University of Singapore
xiangwang1223@gmail.com

Xiangnan He*
University of Science and Technology of China
xiangnanhe@gmail.com

Zikun Hu
National University of Singapore
zikunhu1016@gmail.com

Tat-Seng Chua
National University of Singapore
dcscts@nus.edu.sg

51194506043
杨紫荆

Translation-based User Preference model (TUP)

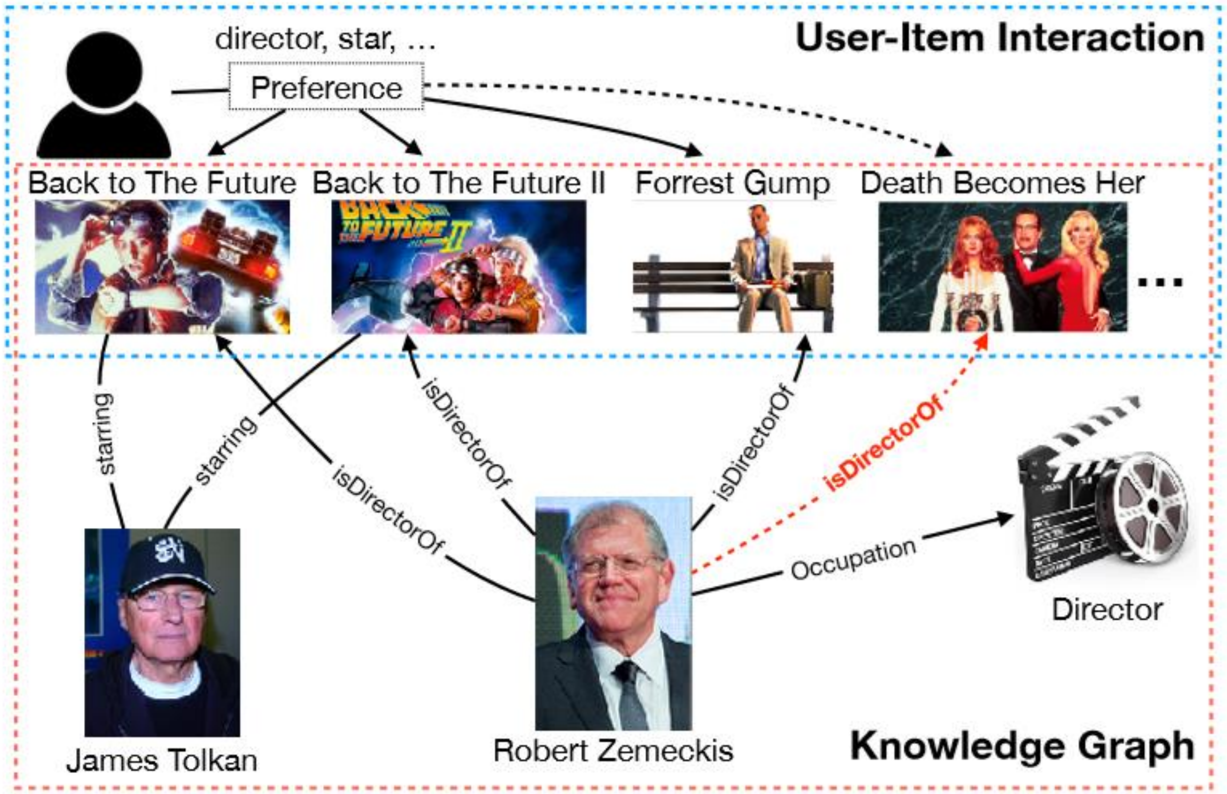


Figure 1: An illustrative example on the necessity of considering the missing relations in KG for recommendation.

Related Work—Item Recommendation

- similarity based methods: CF, matrix factorization techniques
- similarity-based methods with NN: LSTM, RNN
- content-based methods: reviews, relational data, knowledge graph
- translation-based recommendation

Knowledge Graph：

- embedding-based
- path-pased

Related Work—KG Completion

- translational distance models
- semantic matching models.

Relationship between Two Tasks

- translational distance models
- semantic matching models.

Tasks and Notations

- Item Recommendation :
  - Input: Y={(u,i)} ; targrt user:u
  - Output: top-N items
- KG Completion:
  - predict the missing entity e_h or e_t for a triple (e_h,e_t,r)
- TUP
  - item recommendation
  - Input:Y   Output: g(u,i:p)
- KTUP
  - multi-task architecture
  - Input: KG, Y, A={(i,e)}
  - Output: g(u,i:p) f(e_h,e_t,r)

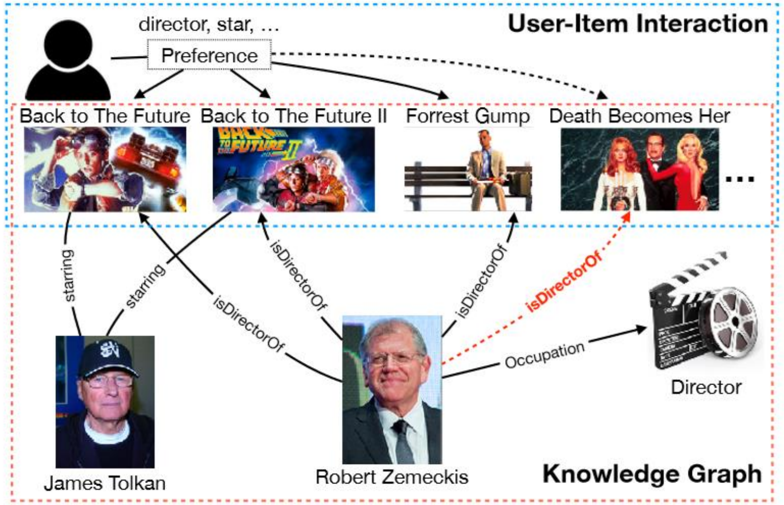


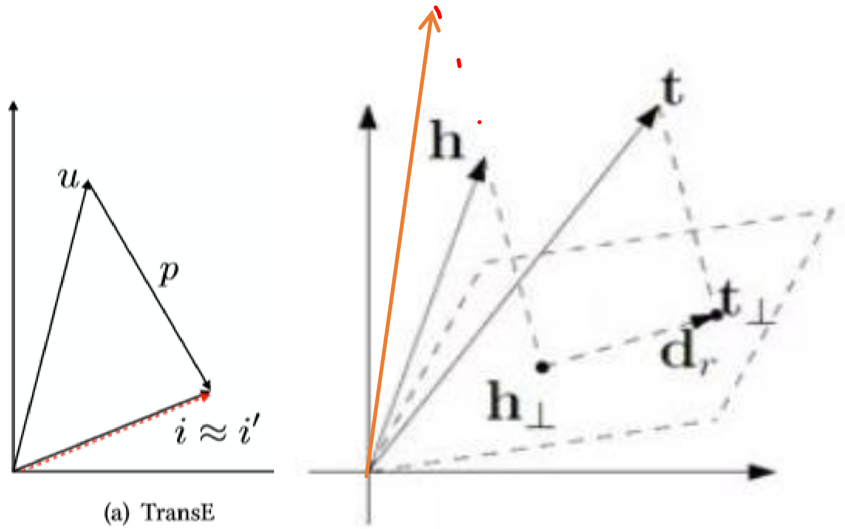
Figure 1: An illustrative example on the necessity of considering the missing relations in KG for recommendation.

TransH for KG Completion

- TransE： $e_h + r = e_t$
- TransH: $f(e_h, e_t, r) = \| e_{\frac{1}{h}} + r - e_{\frac{1}{t}} \|$

$$e_{\frac{1}{h}} = e_h - w_r^T e_h w_r$$

$$e_{\frac{1}{t}} = e_t - w_r^T e_t w_r$$



(a) TransE

Figure 2: Illustration of the two translation schemes for item recommendation

$$\mathcal{L}_k = \sum_{(e_h, e_t, r) \in \mathcal{KG}} \sum_{(e'_h, e'_t, r') \in \mathcal{KG}^-} [f(e_h, e_t, r) + \gamma - f(e'_h, e'_t, r')]_+$$

TUP FOR ITEM RECOMMENDATION

- Preference Induction
  - Hard Strategy: Straight-Through(ST) Gumbel SoftMax

$$\phi(p) = \frac{\exp(\log(\pi_p))}{\sum_{j=1}^{P} \exp(\log(\pi_j))}$$

$$z_p = \begin{cases} 1, & p = \arg\max_j(\log(\pi_j) + g_j) \\ 0, & \text{otherwise} \end{cases} \qquad \phi(u, i, p) = \text{Similarity}(\mathbf{u} + \mathbf{i}, \mathbf{p})$$

$$y_p = \frac{\exp((\log(\pi_p) + g_p)/\tau)}{\sum_{j=1}^{P} \exp((\log(\pi_j) + g_j)/\tau)}$$

  - Soft Strategy:

$$p = \sum_{p' \in P} \alpha_{p'} p' \qquad \alpha_{p'} - \Phi(u, i, p')$$

Hyperplane-based Translation.

$$\mathbf{u}^{\perp} = \mathbf{u} - \mathbf{w}_p^{\mathrm{T}}\mathbf{u}\mathbf{w}_p$$

$$\mathbf{i}^{\perp} = \mathbf{i} - \mathbf{w}_p^{\mathrm{T}}\mathbf{i}\mathbf{w}_p$$

$$\mathbf{w}_p = \sum_{p' \in \mathcal{P}} \alpha_{p'} \mathbf{w}_{p'}$$

$$\mathcal{L}_p = \sum_{(u,i)\in\mathcal{Y}} \sum_{(u,i')\in\mathcal{Y}'} -\log\sigma[g(u,i';p') - g(u,i;p)]$$

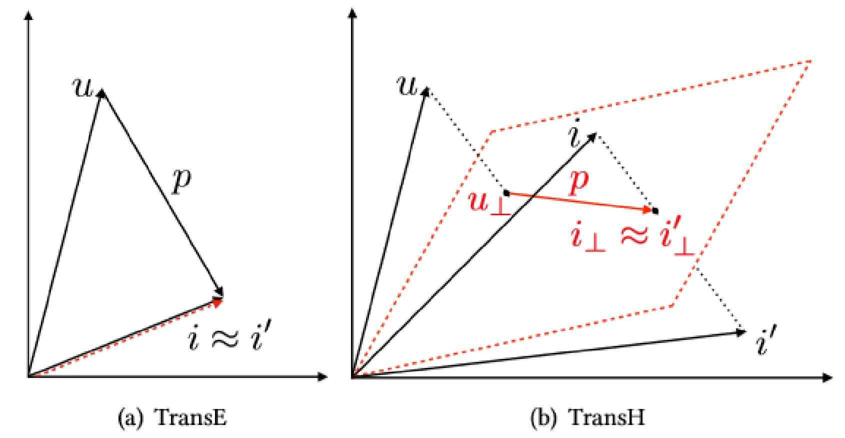


Figure 2: Illustration of the two translation schemes for item recommendation

JOINT LEARNING VIA KTUP FOR TWO TASKS

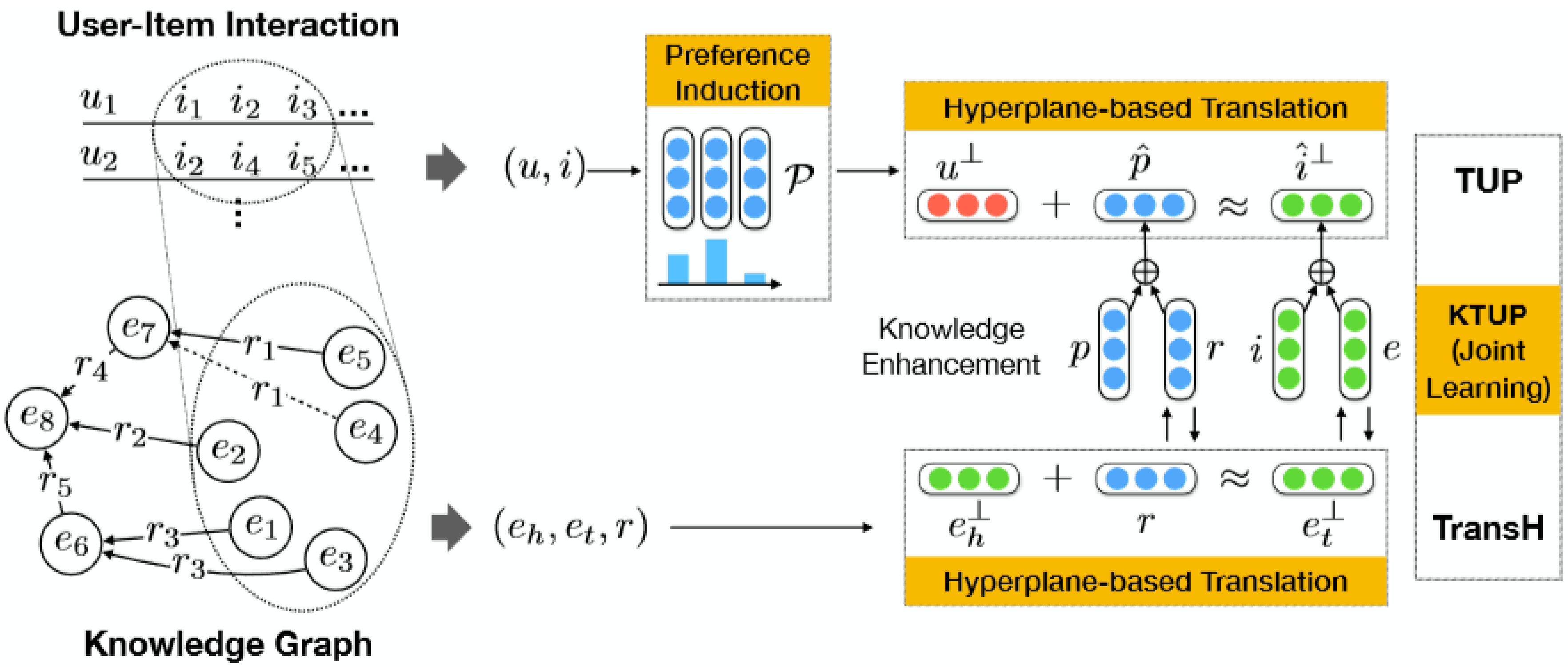


Figure 3: Framwork of KTUP. At the top is TUP for item recommendation including two components: preference induction and hyperplane-based translation. KTUP jointly learns TUP and TransH to enhance the item and preference modeling by transfering knowledge of entities as well as relations.

KTUP

- knowledge enhanced TUP translation function： $g(u, i; p) = \| \mathbf{u}^{\perp} + \hat{\mathbf{p}} - \hat{\mathbf{i}}^{\perp} \|$

 # Entity

- projected: $\hat{\mathbf{i}}^{\perp} = \hat{\mathbf{i}} - \hat{\mathbf{w}}_p^{\mathrm{T}} \hat{\mathbf{i}} \hat{\mathbf{w}}_p$
- knowledge enhanced: $\hat{\mathbf{i}} = \mathbf{i} + \mathbf{e}, \ (i, e) \in \mathcal{A}$

 # Relation

- translation vector: $\hat{\mathbf{p}} = \mathbf{p} + \mathbf{r}$
- projection vector: $\hat{\mathbf{w}}_p = \mathbf{w}_p + \mathbf{w}_r$

Training

$$\mathcal{L} = \lambda \mathcal{L}_p + (1 - \lambda)\mathcal{L}_k$$

Relationship to SOTA Models - Baseline

- Item Recommendation
  - Typical similarity-based methods:FM, BPRMF
  - CFKG
  - CKE
  - CoFM

- Implicity of User Preference
- Variety of User Preference
- Transfered Knowledge from KG

EXPERIMENTS

- Datasets:
  - MovieLens-1m DBPedia
  - DBbook20142

Table 1: Statistics of MovieLens-1m and DBbook2014

		MovieLens-1m	DBbook2014
User-Item Interactions	# Users	6,040	5,576
	# Items	3,240	2,680
	# Ratings	998,539	65,961
	# Avg. ratings	165	12
	Sparsity	94.9%	99.6%
KG	# Entity	14,708	13,882
	# Relation	20	13
	# Triple	434,189	334,511
Multi-Tasks	# Item-Entity Alignments	2,934	2,534
	Coverage	90.6%	94.6%

Item Recommendation

- Metrics: Precision@N, Recall@N, F1score@N, Hitratio@N, nDCG@N

Table 2: Overall performance on Item Recommendation

	MovieLens-1m (@10, %)					DBbook2014 (@10, %)				
	Precision	**Recall**	**F1**	**Hit**	**NDCG**	**Precision**	**Recall**	**F1**	**Hit**	**NDCG**
FM	29.28	11.92	13.81	81.06	59.48	3.44	21.55	5.75	30.15	20.10
BPRMF	30.81	12.95	14.84	83.18	61.02	3.56	22.46	5.96	31.26	21.01
CFKG	29.45	12.49	14.23	82.24	58.97	3.17	19.69	5.30	28.09	19.87
CKE	38.67	16.65	18.94	88.36	67.05	3.92	23.41	6.51	33.18	**27.78**
CoFM (share)	32.08	13.02	15.12	83.30	58.69	3.41	20.78	5.67	29.84	20.92
CoFM (reg)	31.74	12.74	14.87	82.67	58.66	3.32	20.54	5.54	28.96	20.53
TUP (hard)	37.29	17.07	18.98	**89.60**	67.40	3.40	21.11	5.67	29.56	20.19
TUP (soft)	37.00	16.79	18.76	89.47	67.02	3.62	22.81	6.06	31.42	21.54
KTUP (hard)	40.87	17.24	19.79	88.97	69.65	4.04	24.48	6.71	34.49	27.38
KTUP (soft)	**41.03**	**17.25**	**19.82**	89.03	**69.92**	**4.05**	**24.51**	**6.73**	**34.61**	27.62

Influence of Training Data Sparsity

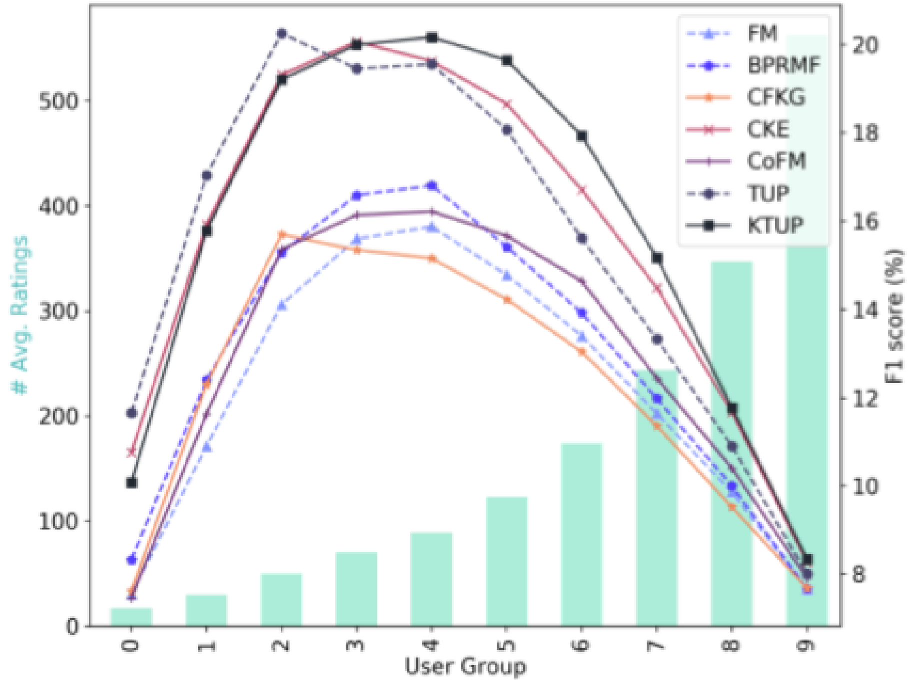


Figure 4: Influence of Different Sparsity on MovieLens-1m. The x-axis shows 10 user groups splited according to interaction number, the left y-axis corresponds to the bars indicating the number of interactions in each user group, and the right y-axis denotes F1-score of curves.

KnowledgeGraph Completion

- Metrics：
  - Hit ratio@N
  - Mean Rank

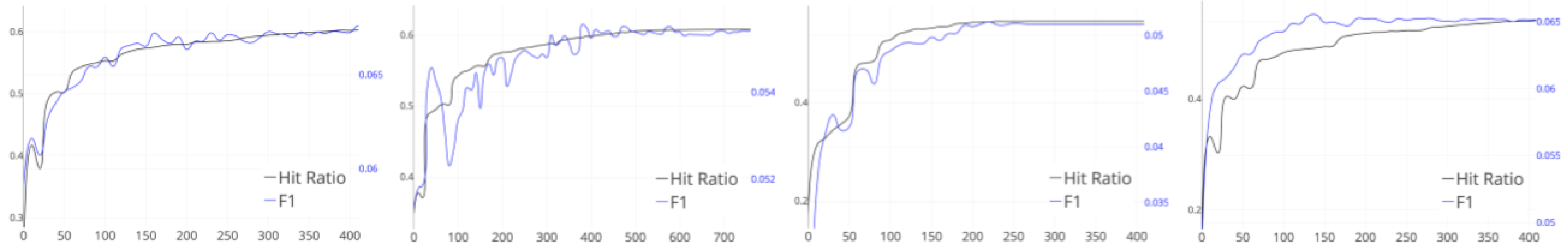
Table 4: Overall performance on KG Completion

	MovieLens-1m		DBbook2014	
	Hit@10 (%)	Mean Rank	Hit@10 (%)	Mean Rank
TransE	46.95	537	60.71	531
TransH	47.63	537	60.06	556
TransR	38.93	609	56.33	563
CFKG	41.56	523	58.83	547
CKE	34.37	585	54.66	593
CoFM (share)	46.62	515	57.01	529
CoFM (reg)	46.51	**506**	60.81	521
KTUP (hard)	48.39	525	60.53	501
KTUP (soft)	**48.90**	527	**60.75**	**499**

Table 3: Performance on MovieLens by Relation Category

Task	Prediction Head (Hits@10, %)				Prediction Tail (Hits@10, %)			
Relation Category	1-to-1	1-to-N	N-to-1	N-to-N	1-to-1	1-to-N	N-to-1	N-to-N
TransE	59.62	56.76	64.55	24.56	**65.38**	62.16	78.52	46.25
TransH	61.54	48.65	65.73	25.51	57.69	78.38	75.62	46.73
TransR	17.31	29.73	32.88	18.50	17.31	43.24	53.12	38.88
CFKG	59.62	51.35	63.31	20.30	57.69	70.27	78.56	41.22
CKE	19.23	21.62	24.16	14.81	7.69	24.32	37.83	34.82
CoFM (share)	65.38	59.46	66.13	24.42	61.54	72.97	**81.05**	45.99
CoFM (reg)	69.23	**70.27**	66.09	24.30	48.08	**86.49**	80.72	45.79
KTUP (hard)	67.31	59.46	66.42	25.67	57.69	81.08	79.22	47.24
KTUP (soft)	**75.00**	56.76	**67.16**	**26.09**	63.46	81.08	78.34	**47.65**

Mutual Benefits of Two Tasks



(a) KTUP ($\rho = 0.97$) (b) CoFM ($\rho = 0.81$) (c) CFKG ($\rho = 0.97$) (d) CKE ($\rho = 0.94$)

Figure 5: Correlation of Training Curves between Two Tasks on DBbook2014, which is denoted by the Pearson's correlation coefficient ρ. The x-axis is training epoch, the left y-axis corresponds to KG completion via hit ratio, and the right y-axis is for item recommendation through F1. (Note that we scale the values of both F1 and Hit Ratio to the same magnitude.)

Case Study

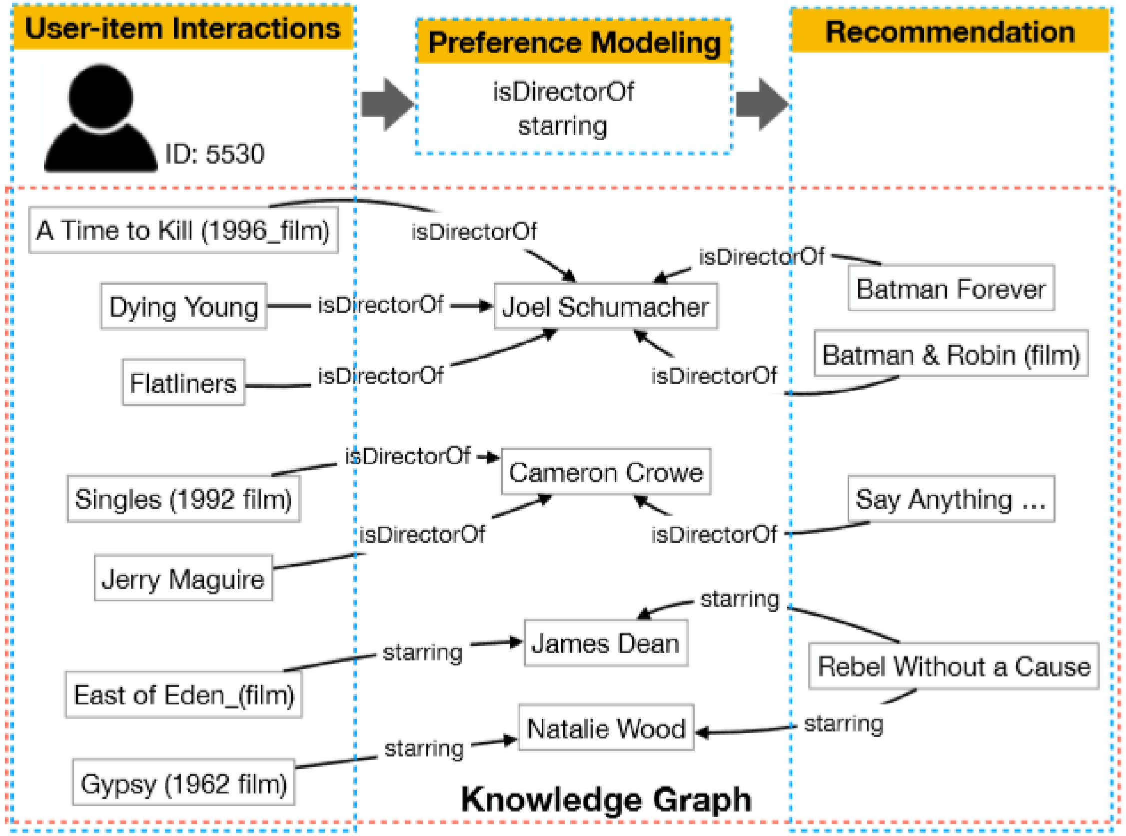


Figure 6: Real Example from MovieLens-1m

CONCLUSION

- translation-based recommender model — TUP
- TUP+KG — model explainability

FUTURE

- multi-hop entity relations — more complex user preferences
- KG reasoning