ICLR
International Conference On Learning Representations

ContextGNN: Beyond Two-Tower Recommendation Systems


Yiwen Yuan

Zecheng Zhang

Xinwei He

Akihiro Nitta

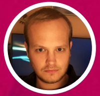
Weihua Hu

Manan Shah

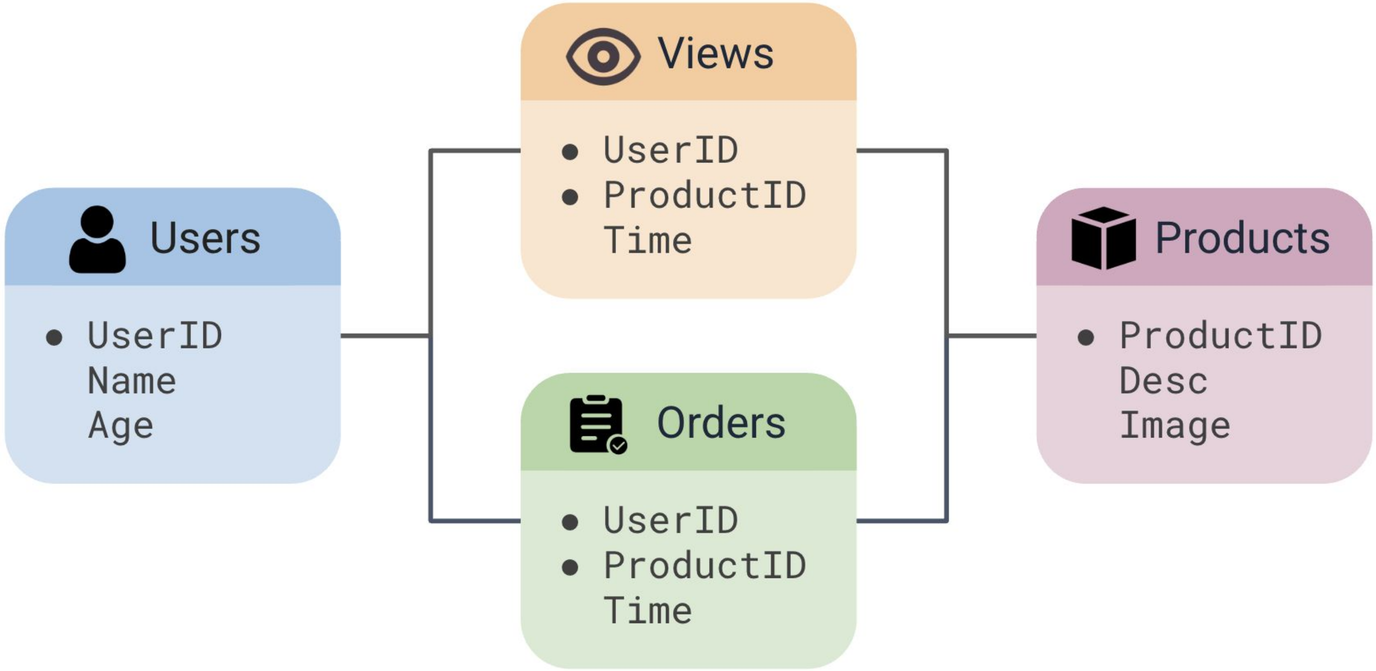
Blaz Stojanovic

Shenyang Huang

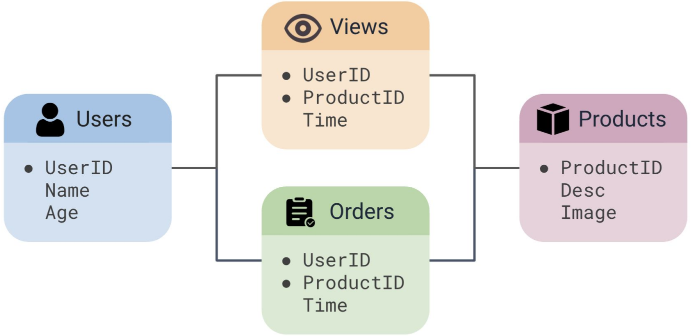
Jan Eric Lenssen

Jure Leskovec

Matthias Fey

RECOMMENDATION ON RELATIONAL DATA

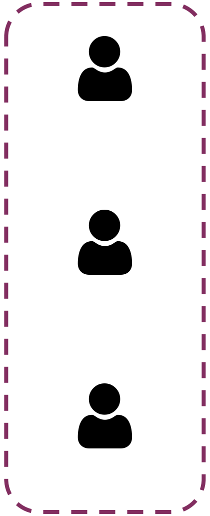


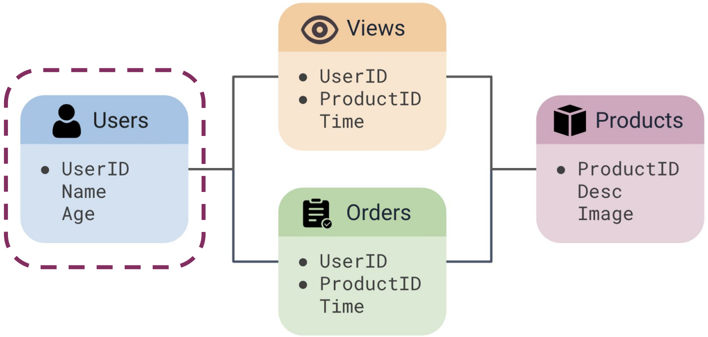
RECOMMENDATION ON RELATIONAL DATA



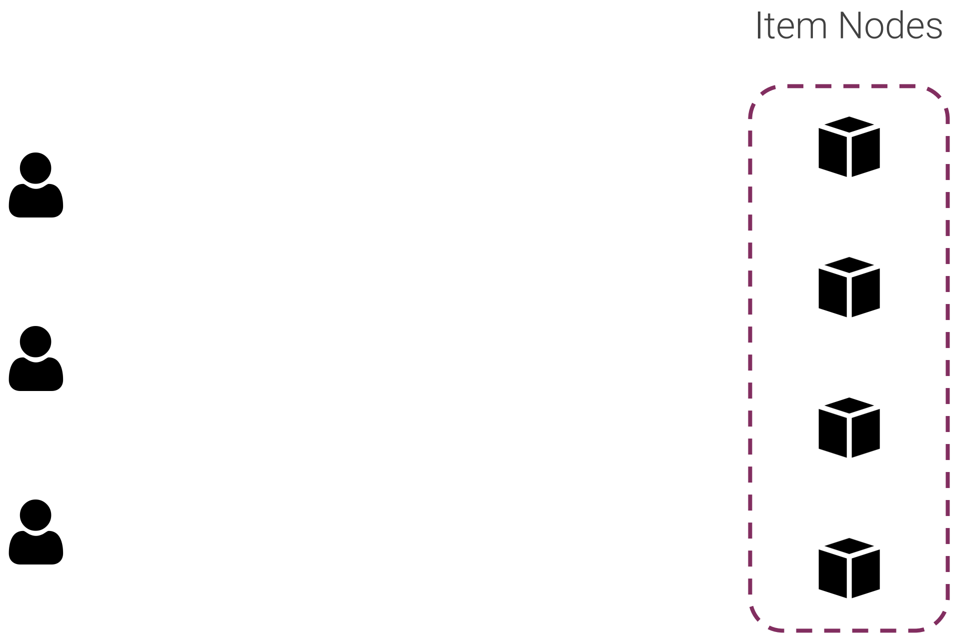
RECOMMENDATION ON RELATIONAL DATA

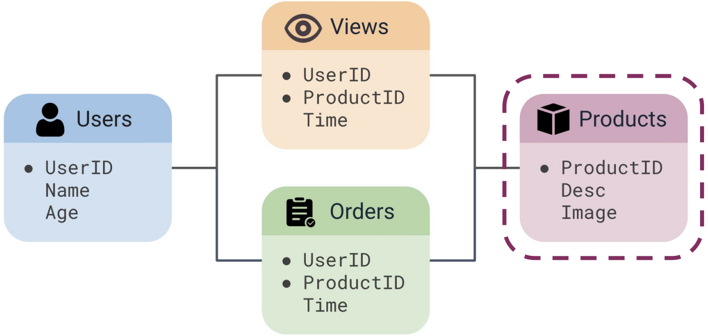
User Nodes



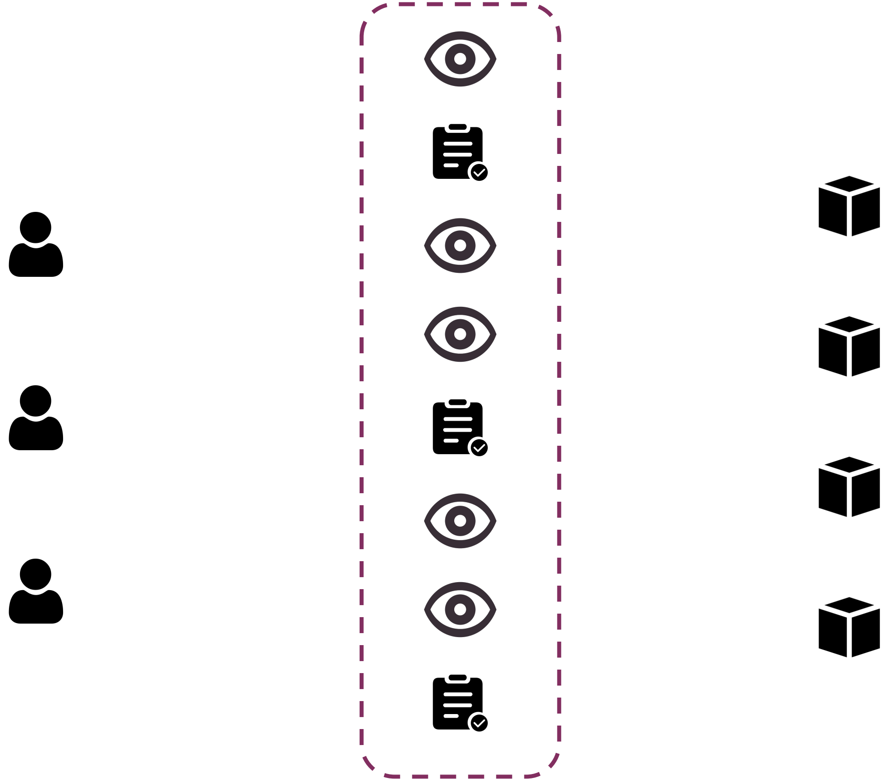


RECOMMENDATION ON RELATIONAL DATA

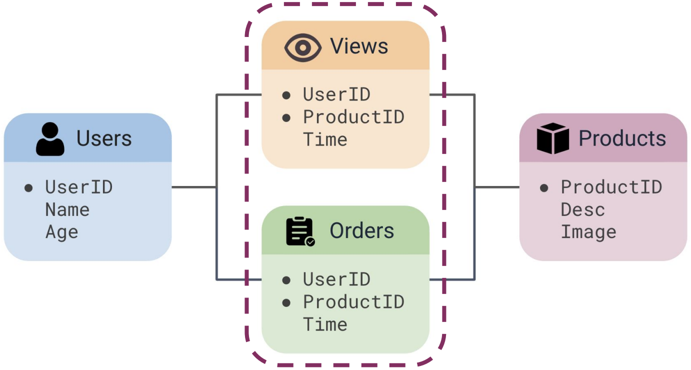




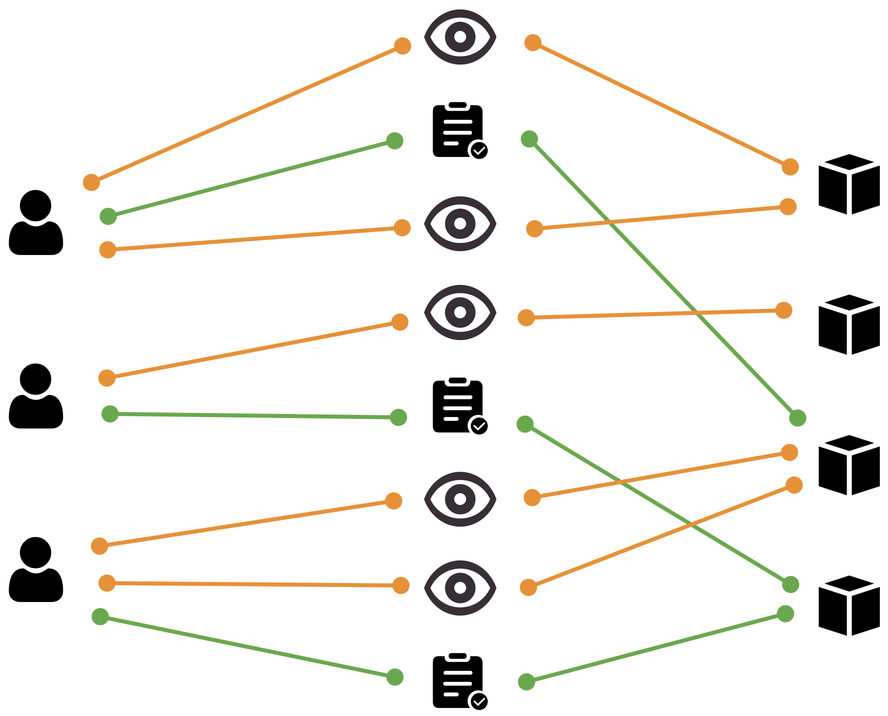
RECOMMENDATION ON RELATIONAL DATA



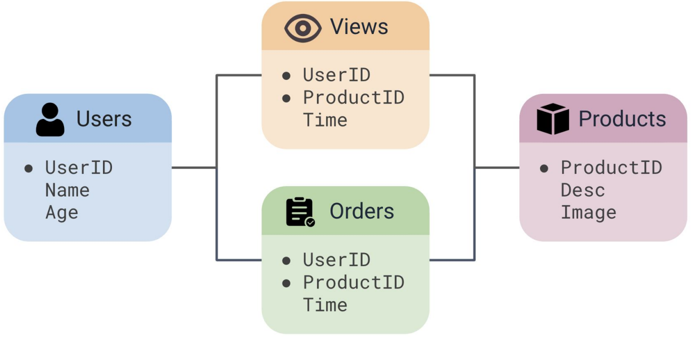
User interaction nodes



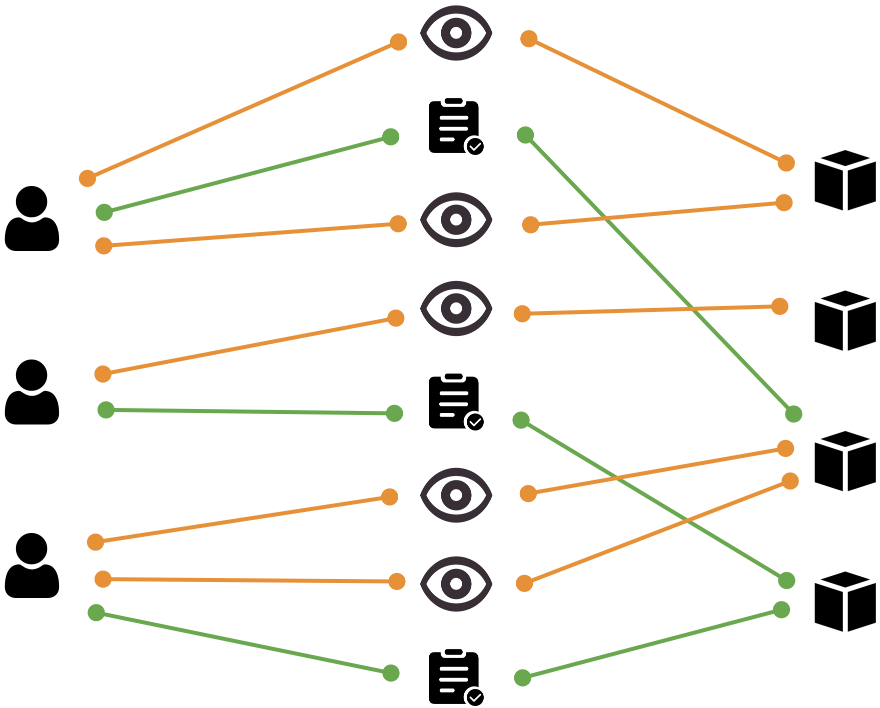
RECOMMENDATION ON RELATIONAL DATA



Matched pkey-fkey
defines an edge



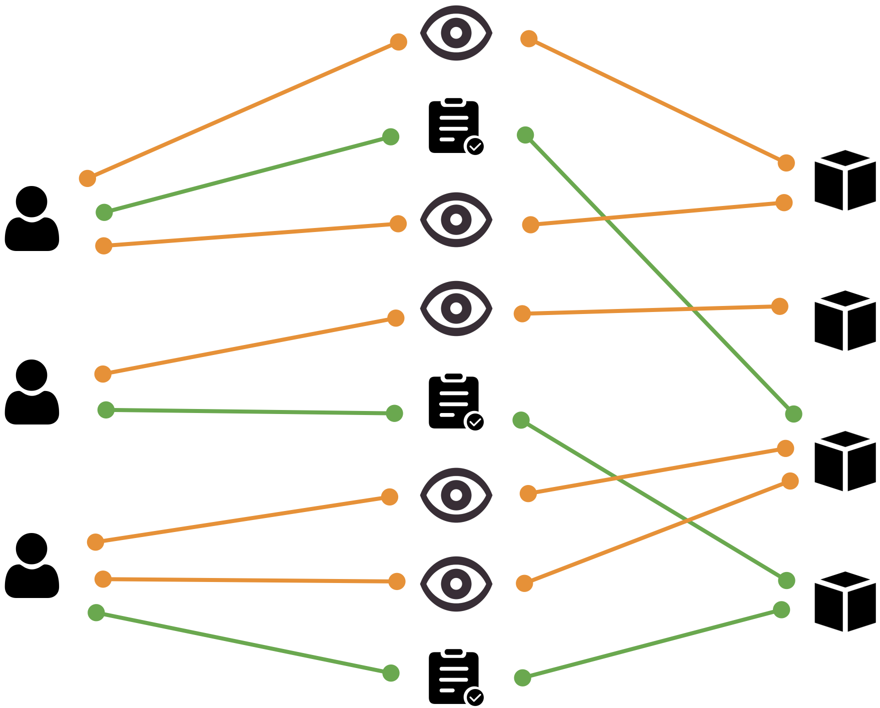
RECOMMENDATION ON RELATIONAL DATA: PROBLEM FORMULATION



Recommendation on relational data:

def: ***link prediction*** *on a* ***bipartite graph*** *of users and item nodes, past interactions are links, and the goal is to predict* ***which links are going to occur*** *in the future.*

RECOMMENDATION ON RELATIONAL DATA: PROBLEM FORMULATION

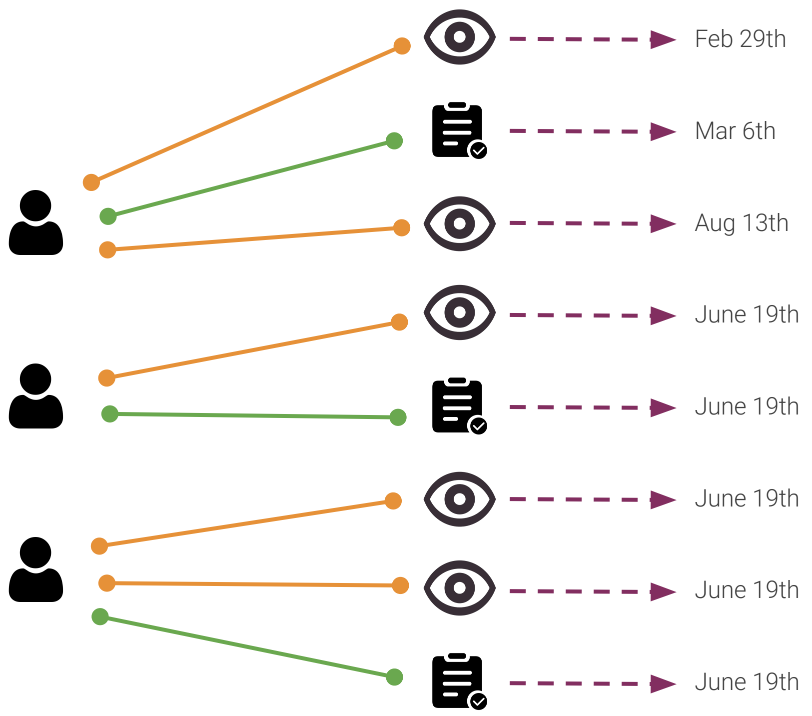


Recommendation on relational data:

*def: **link prediction** on a **bipartite graph** of users and item nodes, past interactions are links, and the goal is to predict **which links are going to occur** in the future.*

Challenges:

RECOMMENDATION ON RELATIONAL DATA: CHALLENGES



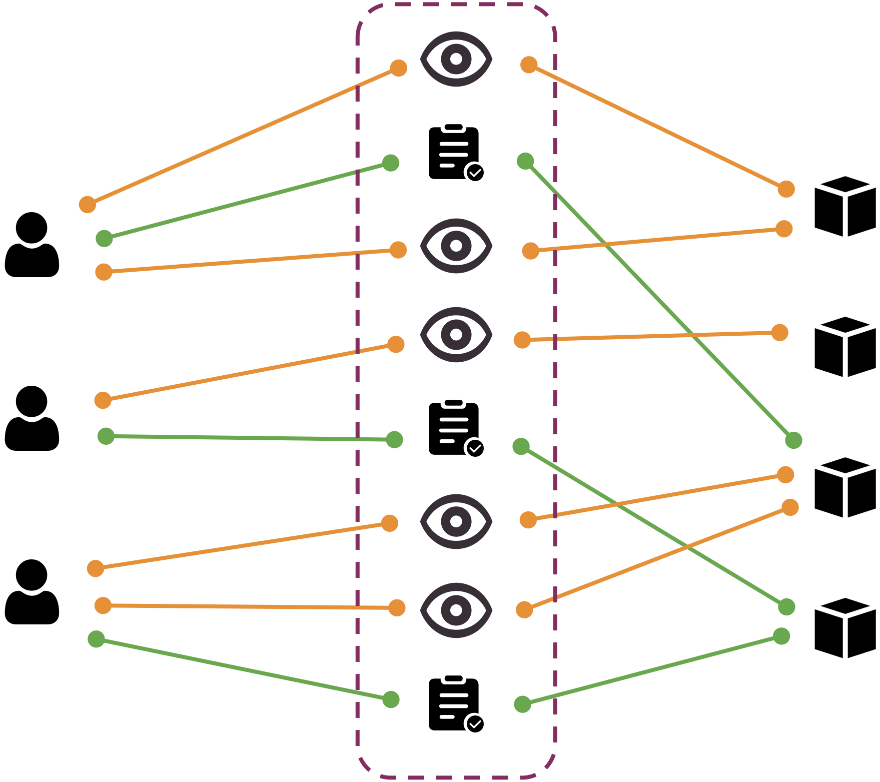
Recommendation on relational data:

*def: **link prediction** on a **bipartite graph** of users and item nodes, past interactions are links, and the goal is to predict **which links are going to occur** in the future.*

Challenges:

- ***Temporal***

RECOMMENDATION ON RELATIONAL DATA: CHALLENGES



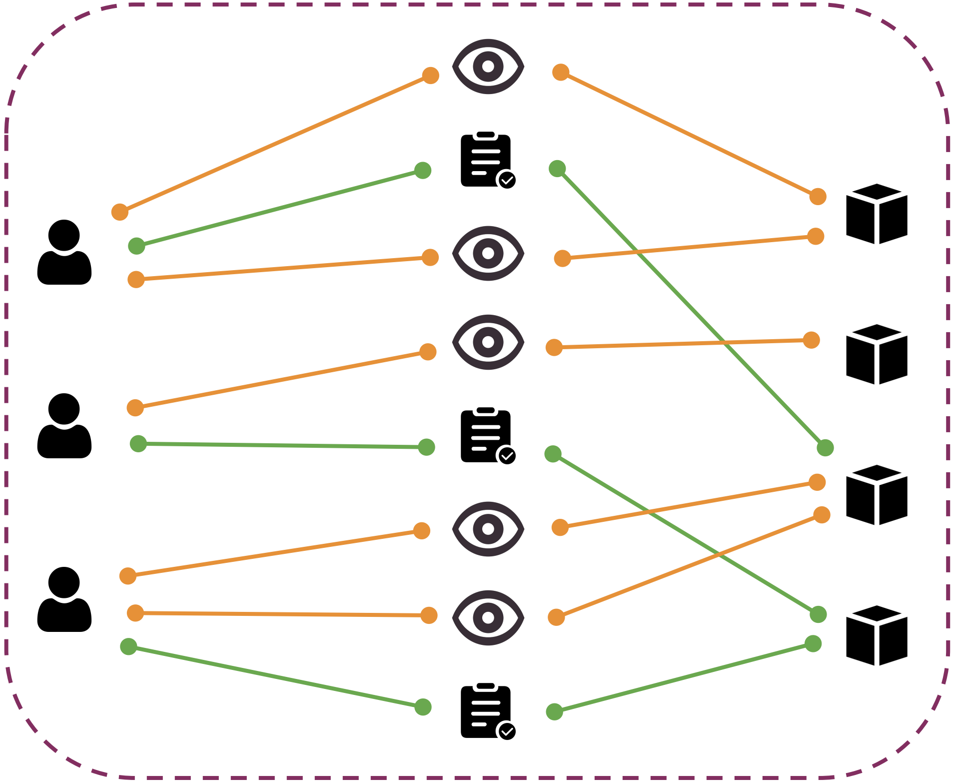
Recommendation on relational data:

*def: **link prediction** on a **bipartite graph** of users and item nodes, past interactions are links, and the goal is to predict **which links are going to occur** in the future.*

Challenges:

- Temporal
- ***Multi-behavioral***

RECOMMENDATION ON RELATIONAL DATA: CHALLENGES



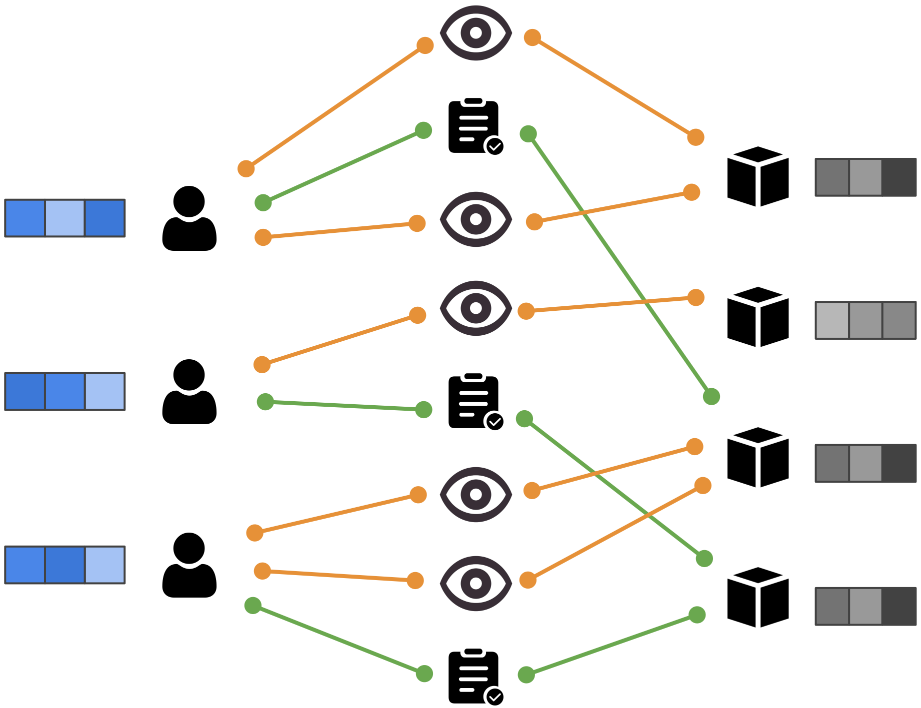
Recommendation on relational data:

*def: **link prediction** on a **bipartite graph** of users and item nodes, past interactions are links, and the goal is to predict **which links are going to occur** in the future.*

Challenges:

- Temporal
- Multi-behavioral
- ***Large Scale***

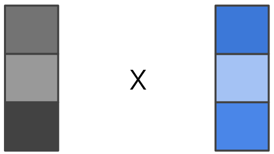
EXISTING APPROACHES: TWO-TOWER REPRESENTATION



The *industry standard* approach to recommendation systems is based on a two-tower paradigm

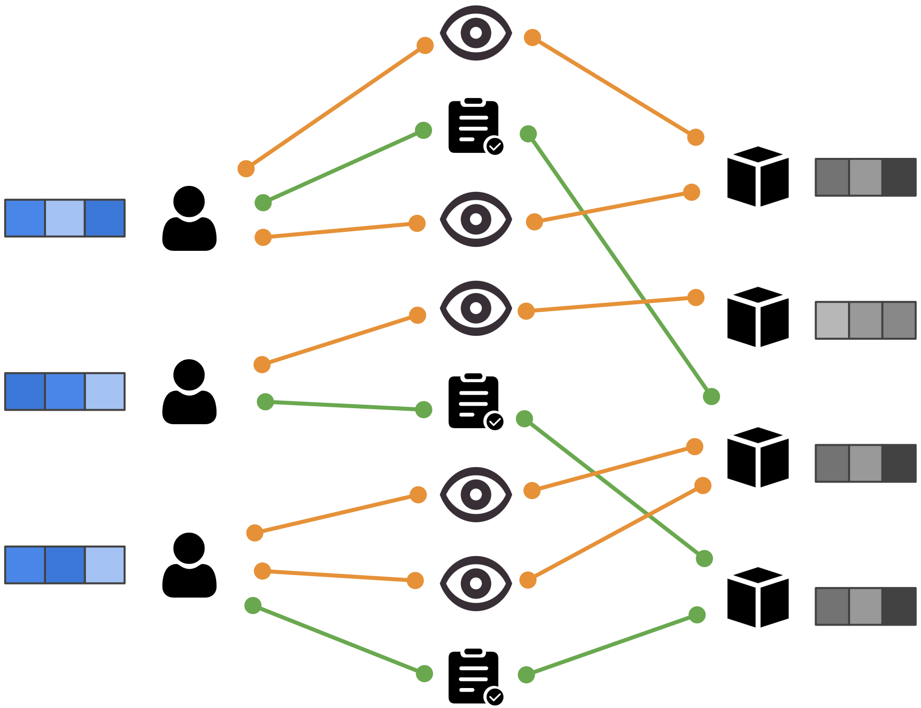
1. One tower embeds ***users***
2. One tower embeds ***items***

Ranked via inner product decoder



Inner product

EXISTING APPROACHES: TWO-TOWER LIMITATIONS



Limitation:

Two tower architectures learn a *pair-agnostic* representation of users and items

- Item representations do not capture the uniqueness of user's view on the items
- Fine-grained context, *e.g.*, of repeated purchase patterns, cannot be captured by two independent representations
- Unable to distinguish between familiar or repeated purchases *vs.* exploratory purchases

EXISTING APPROACHES: PAIR-WISE REPRESENTATION

Pair-wise representations $h(v, w)$, that incorporate the knowledge about the pair they are making predictions for, are able to contextualize the prediction.

Generating pair-wise contextualized predictions for *all* possible user-item pairs is ineffable due to its quadratic complexity.

EXISTING APPROACHES: PAIR-WISE REPRESENTATION

Pair-wise representations $h(v, w)$, that incorporate the knowledge about the pair they are making predictions for, are able to contextualize the prediction.

Generating pair-wise contextualized predictions for *all* possible user-item pairs is ineffable due to its quadratic complexity.

➡ Only generate pair-wise contextualized predictions for a given candidate set

Effectivity of the model is now *bounded* by the recall of the **candidate generation procedure**

ContextGNN: MAKING HYBRID RECOMMENDATIONS

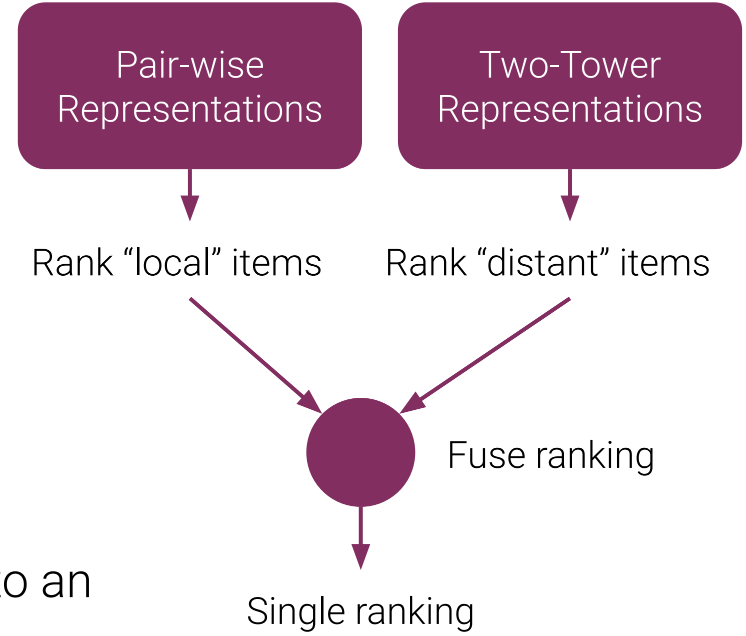
Idea:

1. Contextualize the prediction for the area of items for which a user has **rich past interactions**

   Pair-wise representations are able to capture fine-grained patterns of past user-item interactions

2. Fall back to two-tower representations for **“distant”/exploratory items**

3. **Fuse recommendations** from both models into an *end-to-end* training procedure

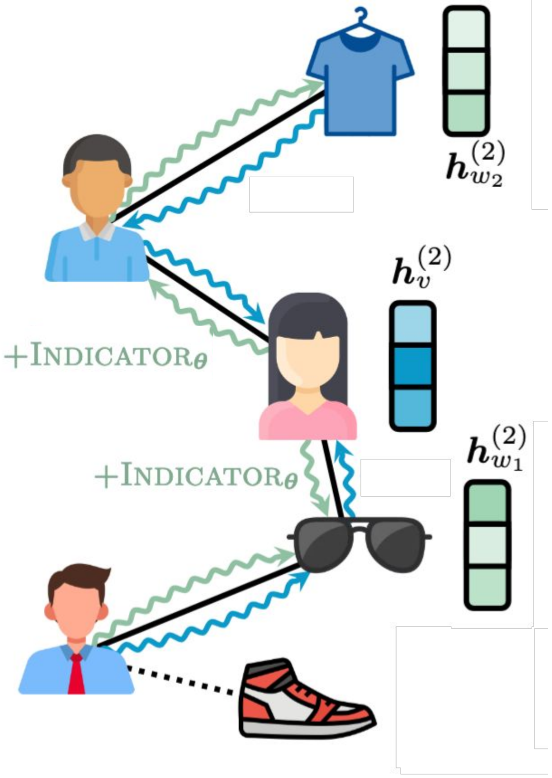


ContextGNN: PAIR-WISE MODEL

Our pair-wise model learns a **pair-wise prediction** *by* **conditioning the item representation on a user-specific subgraph**

1. Sample a *k*-hop subgraph around user *v*
2. Add an INDICATOR$_\theta$ embedding to user *v*
3. Apply a GNN
4. **Readout user representation** *v and* **item representations** *w* for *all* items within the user-specific subgraph

➡ Item representations *w* now depend on user *v*!

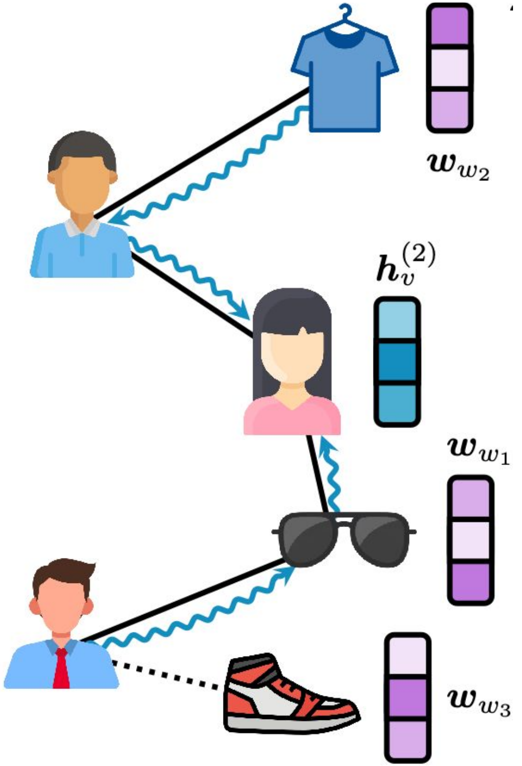


ContextGNN: TWO-TOWER MODEL

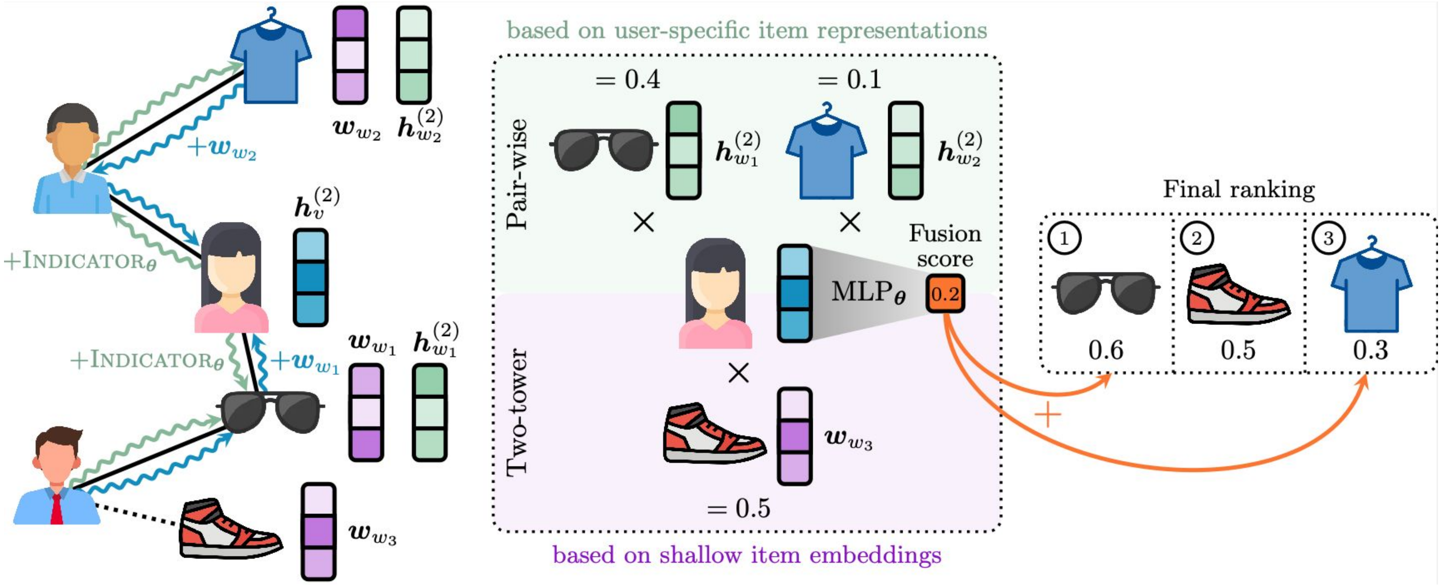
Our two-tower model learns a **pair-agnostic prediction based on a GNN-based user representation** *and* **a shallow item representation**

1. **Re-use user representation** from the pair-wise model
2. **Query a shallow embedding matrix** for item representations

We found shallow item representations to be effective enough to capture key signals while allowing us to scale to a large corpus of negative samples during training!



ContextGNN: PUTTING IT ALL TOGETHER



We *fuse* both the pair-wise and the two-tower representations by computing a user-specific fusion-score and using it to adjust the final ranking accordingly.

ContextGNN: EVALUATION - RELBENCH

Table 2: **Recommendation results** (MAP, higher is better, in %) on RELBENCH.

Task	LIGHT GBM	MULTI VAE	GRAPH SAGE	NGCF	NBFNET	SHALLOW ITEM	CONTEXT GNN
			rel-amazon				
user-item-purchase	0.16	0.23	0.74	0.88	2.06	0.56	**2.93**
user-item-rate	0.17	0.24	0.87	0.86	1.24	0.74	**2.25**
user-item-review	0.09	0.10	0.47	0.55	1.57	0.40	**1.63**
			rel-hm				
user-item-purchase	0.38	0.28	0.80	0.75	2.81	0.40	**2.93**
			rel-stack				
user-post-comment	0.04	0.01	0.11	0.13	12.72	0.03	**13.34**
post-post-related	2.00	0.78	0.07	0.13	10.83	0.82	**11.18**
			rel-trial				
condition-sponsor-run	4.82	2.47	2.89	3.88	11.36	0.85	**11.65**
site-sponsor-run	8.40	6.17	10.70	6.54	19.06	10.66	**28.02**
Average (↑)	2.01	1.29	2.08	1.72	7.71	1.81	**9.23**

ContextGNN: TEMPORAL & STATIC LINK PREDICTION

Model	HR@1	HR@5	NDCG@5	HR@10	NDCG@10
DEEPFM (2017)	0.138	0.332	0.244	0.469	0.290
BERT4REC (2019)	0.141	0.356	0.261	0.467	0.297
CHORUS (2020a)	0.140	0.345	0.247	0.457	0.283
HYREC (2020b)	0.137	0.323	0.229	0.442	0.266
NMTR (2019)	0.141	0.360	0.254	0.481	0.304
MATN (2020)	0.142	0.375	0.273	0.489	0.309
MBGCN (2020)	0.137	0.332	0.228	0.463	0.277
TGT (2022)	0.148	0.399	0.293	0.519	0.330
CONTEXTGNN	**0.411**	**0.603**	**0.513**	**0.667**	**0.534**

Model	Recall@20	NDCG@20
NGCF (2019)	0.0337	0.0261
LIGHTGCN (2020)	0.0410	0.0318
ULTRAGCN (2021)	**0.0681**	**0.0556**
LIGHTGCL (2023)	0.0585	0.0436
SIMGCL (2022)	0.0478	0.0379
SGL (2021)	0.0468	0.0371
CONTEXTGNN	0.0451	0.0377

ContextGNN: SUMMARY

1. **Two-tower** representations are *pair-agnostic*, and are unable to fully represent the rich user-item behaviors we find industry datasets

2. **Pair-wise** representations can contextualize (user, item) rankings, but are ineffable due to the their quadratic complexity

3. **CONTEXTGNN** learns **pair-wise** and **two-tower** representations end-to-end as part of a single GNN backbone. The two representations are fused using a user-specific fusion-score

4. **CONTEXTGNN** outperforms relevant state-of-the-art recommendation system methods on *heterogeneous temporal* recommendation datasets (RELBENCH)