# DSC250: Advanced Data Mining

## Recommeder System

**Zhiting Hu**

Lecture 18, November 30, 2023

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

# Logistics

- Mid-quarter evaluation
  - Google form: see Piazza
  - Please complete by EOD Monday Dec.4

# Outline

- Recommender System

- 5 paper presentations
  - Jiongli Zhu, Shuying Li
  - Henry Wang, James Chen
  - Neha Mittal, Rijul Sherathia
  - Akansha Lalwani, Divyansh Srivastava
  - Harin, Golokesh

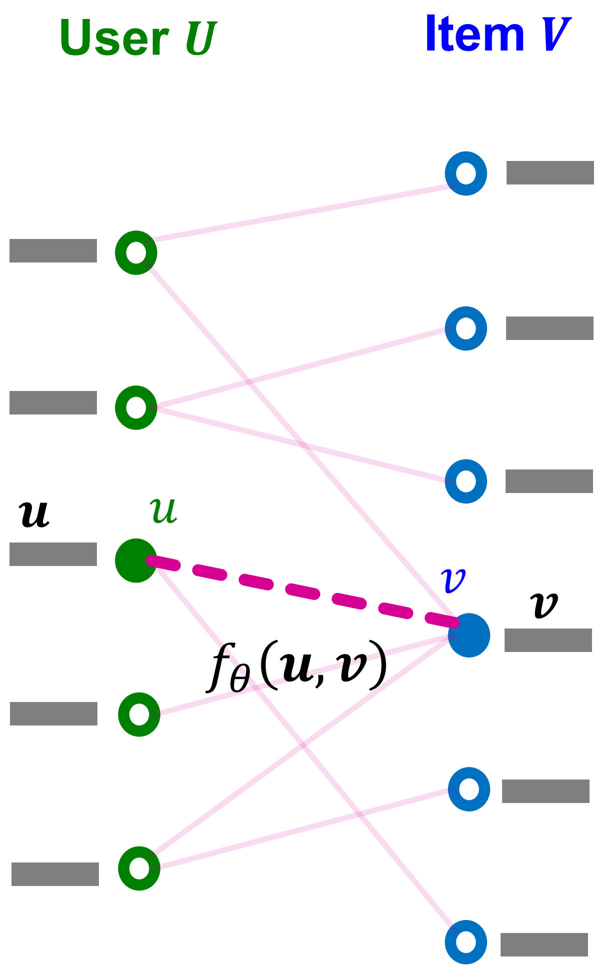
# Recommender System (RecSys)

Slides adapted from:

- Y. Sun, CS 247: Advanced Data Mining
- Jure Leskovec, Stanford CS224W: Machine Learning with Graphs

# Recap: Embedding-Based Models

- We consider **embedding-based models** for scoring user-item interactions.
  - For each user $u \in \boldsymbol{U}$, let $\boldsymbol{u} \in \mathbb{R}^D$ be its $D$-dimensional embedding.
  - For each item $v \in \boldsymbol{V}$, let $\boldsymbol{v} \in \mathbb{R}^D$ be its $D$-dimensional embedding.
  - Let $f_\theta(\cdot,\cdot): \mathbb{R}^D \times \mathbb{R}^D \to \mathbb{R}$ be a parametrized function.
  - Then, $\boxed{\text{score}(u, v) \equiv f_\theta(\boldsymbol{u}, \boldsymbol{v})}$



# Recap: Limitations of Shallow Encoders

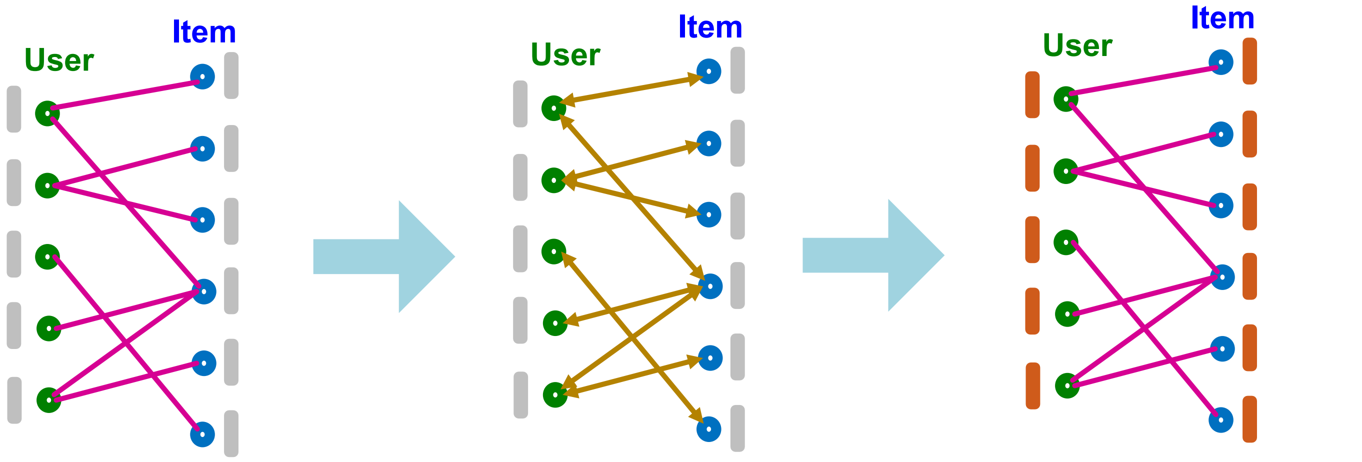
- The model itself does ***not explicitly*** capture graph structure
  - The graph structure is ***only implicitly*** captured in the training objective.
- Only the **first-order graph structure** (i.e., edges) is captured in the training objective.
  - **High-order graph structure** (e.g., $K$-hop paths between two nodes) is ***not* explicitly captured.**

# Recap: We want a model that …

- We want a model that…
  - **explicitly captures graph structure** (beyond implicitly through the training objective)
  - captures **high-order graph structure** (beyond the first-order edge connectivity structure)
- **GNNs are a natural approach to achieve both!**
  - **Neural Graph Collaborative Filtering** (NGCF) [Wang et al. 2019]
  - **LightGCN** [He et al. 2020]
    - A simplified and improved version of NGCF

# NGCF: Overview

- **Neural Graph Collaborative Filtering (NGCF)** ***explicitly*** incorporates high-order graph structure when generating user/item embeddings.
- **Key idea**: Use a GNN to generate graph-aware user/item embeddings.



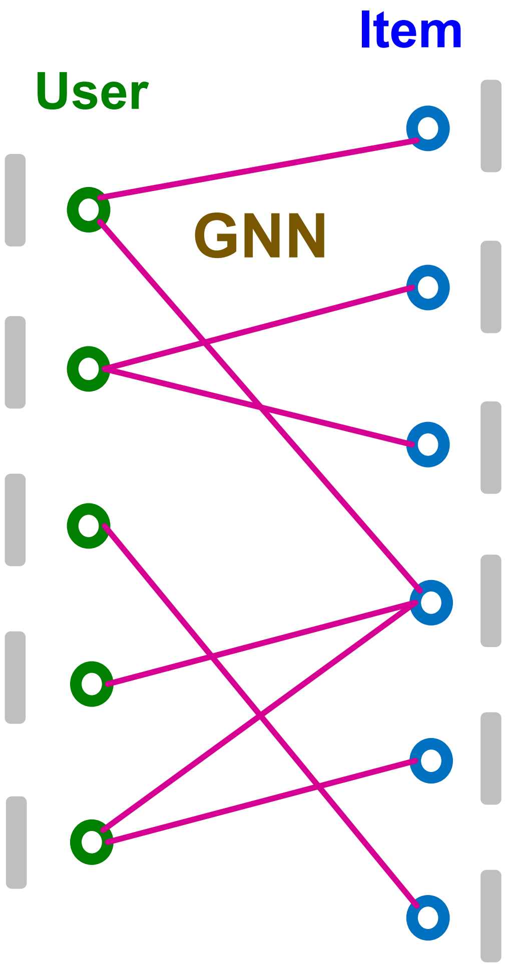

Initial shallow embeddings (not graph-aware)

Use a GNN to propagate embeddings

NGCF's graph-aware embeddings

# NGCF

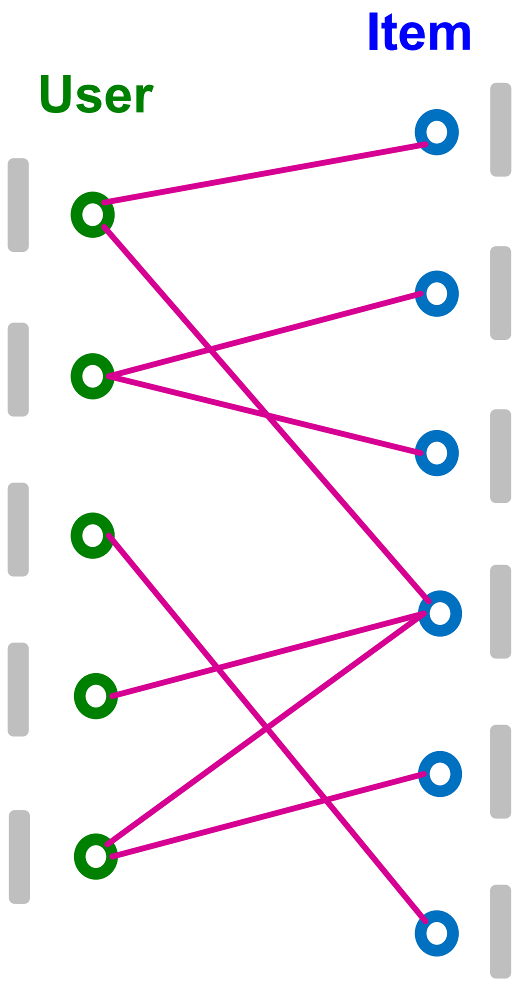
- **Given**: User-item bipartite graph.
- **NGCF framework:**
  - Prepare shallow learnable embedding for each node.
  - Use multi-layer GNNs to propagate embeddings along the bipartite graph.
    - High-order graph structure is captured.
  - Final embeddings are *explicitly* graph-aware!
- **Two kinds of learnable params are jointly learned:**
  - Shallow user/item embeddings
  - GNN’s parameters




Shallow user/item embeddings (learnable)

# NGCF: Initial Node Embeddings

- Set the shallow learnable embeddings as the initial node features:
  - For every user $u \in \boldsymbol{U}$, set $\boldsymbol{h}_u^{(0)}$ as the user's shallow embedding.
  - For every item $v \in \boldsymbol{V}$, set $\boldsymbol{h}_v^{(0)}$ as the item's shallow embedding.




Learnable shallow user/item embeddings

# NGCF: Neighbor Aggregation

- Iteratively update node embeddings using neighboring embeddings.

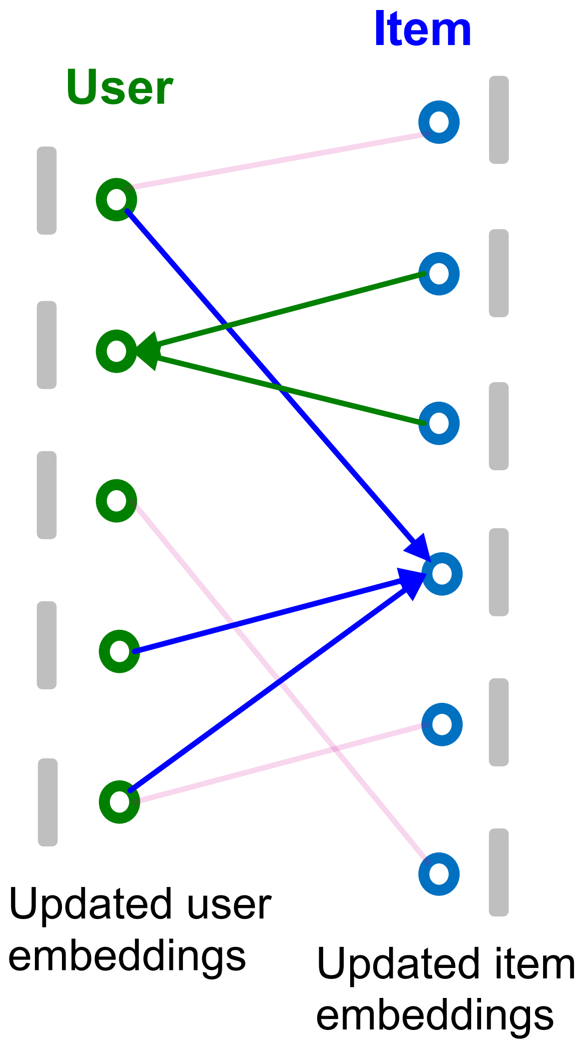
$$\boldsymbol{h}_v^{(k+1)} = \text{COMBINE}\left(\boldsymbol{h}_v^{(k)}, \text{AGGR}\left(\left\{\boldsymbol{h}_u^{(k)}\right\}_{u \in N(v)}\right)\right)$$

$$\boldsymbol{h}_u^{(k+1)} = \text{COMBINE}\left(\boldsymbol{h}_u^{(k)}, \text{AGGR}\left(\left\{\boldsymbol{h}_v^{(k)}\right\}_{v \in N(u)}\right)\right)$$

**High-order graph structure is captured through iterative neighbor aggregation.**

Different architecture choices are possible for AGGR and COMBINE.

- $\text{AGGR}(\cdot)$ can be $\text{MEAN}(\cdot)$
- $\text{COMBINE}(\boldsymbol{x}, \boldsymbol{y})$ can be $\text{ReLU}(\text{Linear}(\text{Concat}(\boldsymbol{x}, \boldsymbol{y})))$



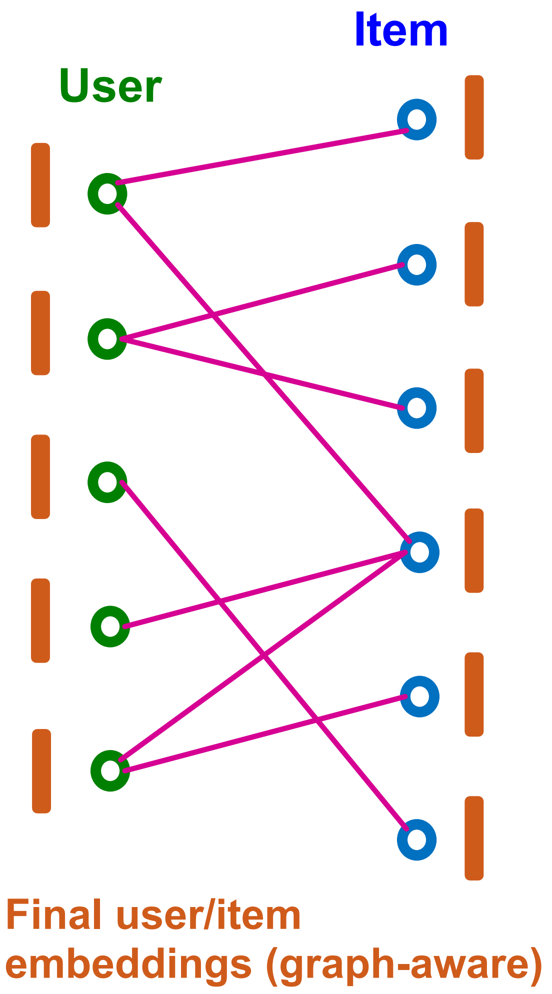
# NGCF: Final Embeddings and Score Function

- After $K$ rounds of neighbor aggregation, we get the **final user/item embeddings** $\boldsymbol{h}_u^{(K)}$ and $\boldsymbol{h}_v^{(K)}$.
- For all $u \in \boldsymbol{U}, v \in \boldsymbol{V}$, we set

$$\boldsymbol{u} \leftarrow \boldsymbol{h}_u^{(K)}, \boldsymbol{v} \leftarrow \boldsymbol{h}_v^{(K)}.$$

- Score function is the inner product

$$\text{score}(u, v) = \boldsymbol{u}^T \boldsymbol{v}$$



# NGCF: Summary

- Conventional collaborative filtering uses shallow user/item embeddings.
  - The embeddings do ***not explicitly* model graph structure**.
  - The training objective **does not model high-order graph structure.**
- **NGCF uses a GNN to propagate the shallow embeddings.**
  - The embeddings are **explicitly aware of high-order graph structure.**

# Issues of Collaborative Filtering

- **Cold Start**: There needs to be enough other users already in the system to find a match.
- **Sparsity**: If there are many items to be recommended, even if there are many users, the user/ratings matrix is sparse, and it is hard to find users that have rated the same items.
- **First Rater**: Cannot recommend an item that has not been previously rated.
  - New items
  - Esoteric items
- **Popularity Bias**: Cannot recommend items to someone with unique tastes.
  - Tends to recommend popular items.

# Methods

- Collaborative filtering
- Content-based recommendation
- Hybrid methods

# Content-based Recommendation

- Collaborative filtering does NOT require any information about content,
  - However, it might be reasonable to exploit such information
  - E.g. recommend fantasy novels to people who liked fantasy novels in the past
- What do we need:
  - Information about the available items such as the genre ("content")
  - *user profile* describing what the user likes (the preferences)
- The task:
  - Learn user preferences
  - Locate/recommend items that are "similar" to the user preferences

# Content-based Recommendation

User profile

| Title | Genre | Author | Type | Price | Keywords |
|---|---|---|---|---|---|
| The Night of the Gun | Memoir | David Carr | Paperback | 29.90 | Press and journalism, drug addiction, personal memoirs, New York |
| The Lace Reader | Fiction, Mystery | Brunonia Barry | Hardcover | 49.90 | American contemporary fiction, detective, historical |
| Into the Fire | Romance, Suspense | Suzanne Brockmann | Hardcover | 45.90 | American fiction, Murder, Neo-nazism |
| ... | | | | | |

Item

| Title | Genre | Author | Type | Price | Keywords |
|---|---|---|---|---|---|
| ... | Fiction, Suspense | Brunonia Barry, Ken Follet, .. | Paperback | 25.65 | detective, murder, New York |

- Simple approach
  - Compute the similarity of an unseen item with the user profile based on the keyword overlap (e.g. using the Dice coefficient)
  - $\mathrm{sim}(b_i, b_j) = \frac{2 * |keywords(b_i) \cap keywords(b_j)|}{|keywords(b_i)| + |keywords(b_j)|}$
- Other advanced similarity measure

# Methods

- Collaborative filtering

- Content-based recommendation

- Hybrid methods
  - Combining both user-item interaction and other external sources of information
  - E.g., Factorization Machines

# Conclusions

# This Course: Summary

1) Text mining
   - Topic models
     - LDA, Expectation Maximization, variational inference
   - Language models
   - Text representation learning (embedding)
2) Graph/network mining
   - Node embedding
   - Graph neural networks
   - Knowledge graphs and reasoning
3) Recommender systems
   - Collaborative filtering
     - Embedding-based methods
     - Graph neural networks for recommendation
   - Content-based recommendation
   - Hybrid methods

# Data Mining Society

- 1989 IJCAI Workshop on Knowledge Discovery in Databases
  - Knowledge Discovery in Databases (G. Piatetsky-Shapiro and W. Frawley, 1991)
- 1991-1994 Workshops on Knowledge Discovery in Databases
  - Advances in Knowledge Discovery and Data Mining (U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, 1996)
- 1995-1998 International Conferences on Knowledge Discovery in Databases and Data Mining (KDD'95-98)
  - Journal of Data Mining and Knowledge Discovery (1997)
- ACM SIGKDD conferences since 1998 and SIGKDD Explorations
- More conferences on data mining
  - PAKDD (1997), PKDD (1997), SIAM-Data Mining (2001), (IEEE) ICDM (2001), etc.
- ACM Transactions on KDD starting in 2007

# Conferences and Journals on Data Mining

- KDD Conferences
  - ACM SIGKDD Int. Conf. on Knowledge Discovery in Databases and Data Mining (KDD)
  - SIAM Data Mining Conf. (SDM)
  - (IEEE) Int. Conf. on Data Mining (ICDM)
  - European Conf. on Machine Learning and Principles and practices of Knowledge Discovery and Data Mining (ECML-PKDD)
  - Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD)
  - Int. Conf. on Web Search and Data Mining (WSDM)
- Other related conferences
  - DB conferences: ACM SIGMOD, VLDB, ICDE, EDBT, ICDT, …
  - Web and IR conferences: WWW, SIGIR, WSDM
  - ML conferences: ICML, NIPS
  - PR conferences: CVPR,
- Journals
  - Data Mining and Knowledge Discovery (DAMI or DMKD)
  - IEEE Trans. On Knowledge and Data Eng. (TKDE)
  - KDD Explorations
  - ACM Trans. on KDD

# Questions?