

图神经网络推荐算法



讲师: Houye



- ① 图和推荐的关系
- ② 基于图神经网络的推荐算法

01 图和推荐的关系

图和推荐的关系

推荐问题：

给定用户的历史行为，预测
用户与商品的交互行为(点击，
购买)

注：

- (1)预测是否点击0、1
- (2)预测用于对商品评分1~5

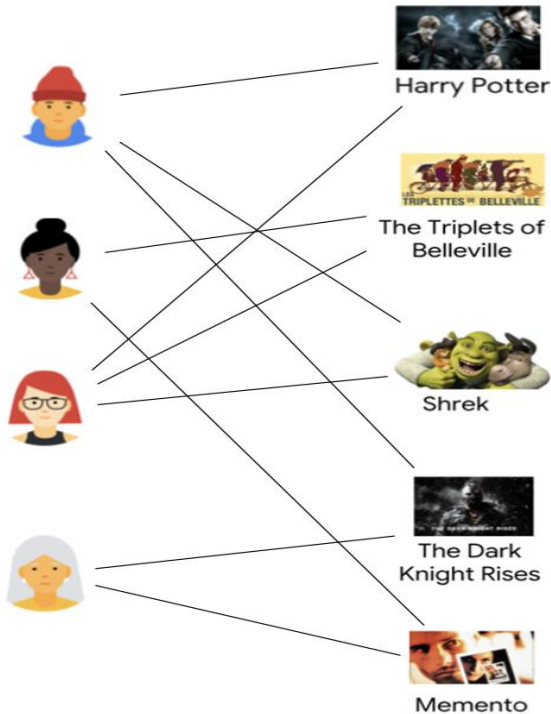
					
	✓		✓	✓	
		✓			✓
	✓	✓	✓		
				✓	✓

链路预测：

给定图上的部分边，预测可
能存在的边

注：

- (1)预测边是否存在边0、1
- (2)预测边的权重1~5(带权图)



模型



The Triplets of
Belleville



Memento

图和推荐的联系

图表示学习->推荐系统

用户U是否喜欢商品V可以用它们的向量表示的相似度来衡量。

$$\min_{U \in \mathbb{R}^{m \times d}, V \in \mathbb{R}^{n \times d}} \sum_{(i,j) \in \text{obs}} (A_{ij} - \langle U_i, V_j \rangle)^2.$$

					
	✓		✓	✓	
		✓			✓
	✓	✓	✓		
				✓	✓

≈

		.9	-1	1	1	-9
		-2	-8	-1	.9	1
1	.1	.88	-1.08	0.9	1.09	-0.8
-1	0	-0.9	1.0	-1.0	-1.0	0.9
.2	-1	0.38	0.6	1.2	-0.7	-1.18
.1	1	-0.11	-0.9	-0.9	1.0	0.91

图表示学习



The Triplets of Belleville



Memento

推荐系统

图和推荐的关系

图表示学习->推荐系统

用户、商品向量表示是推荐的基础，直接决定了推荐的效果

简单图表示学习模型(矩阵分解)

随机初始化，浅层模型进行训练，得到 U_i, V_j

$$\min_{U \in \mathbb{R}^{m \times d}, V \in \mathbb{R}^{n \times d}} \sum_{(i,j) \in \text{obs}} (A_{ij} - \langle U_i, V_j \rangle)^2.$$

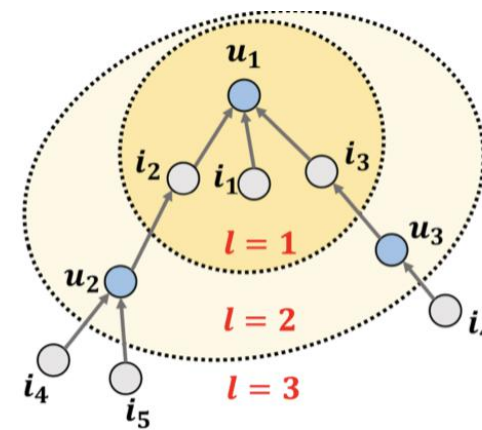
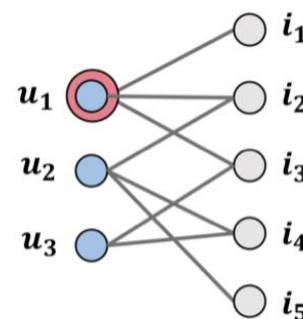
深度图表示学习模型(图神经网络)

随机初始化/特征初始化，深度图神经网络进行训练(聚合邻居信息，更新节点表示)，得到 U_i, V_j

$$\min_{U \in \mathbb{R}^{m \times d}, V \in \mathbb{R}^{n \times d}} \sum_{(i,j) \in E} (A_{ij} - \langle U_i, V_j \rangle)^2$$

$$U_i = \textcolor{red}{GNN}(i)$$

$$V_j = \textcolor{red}{GNN}(j)$$



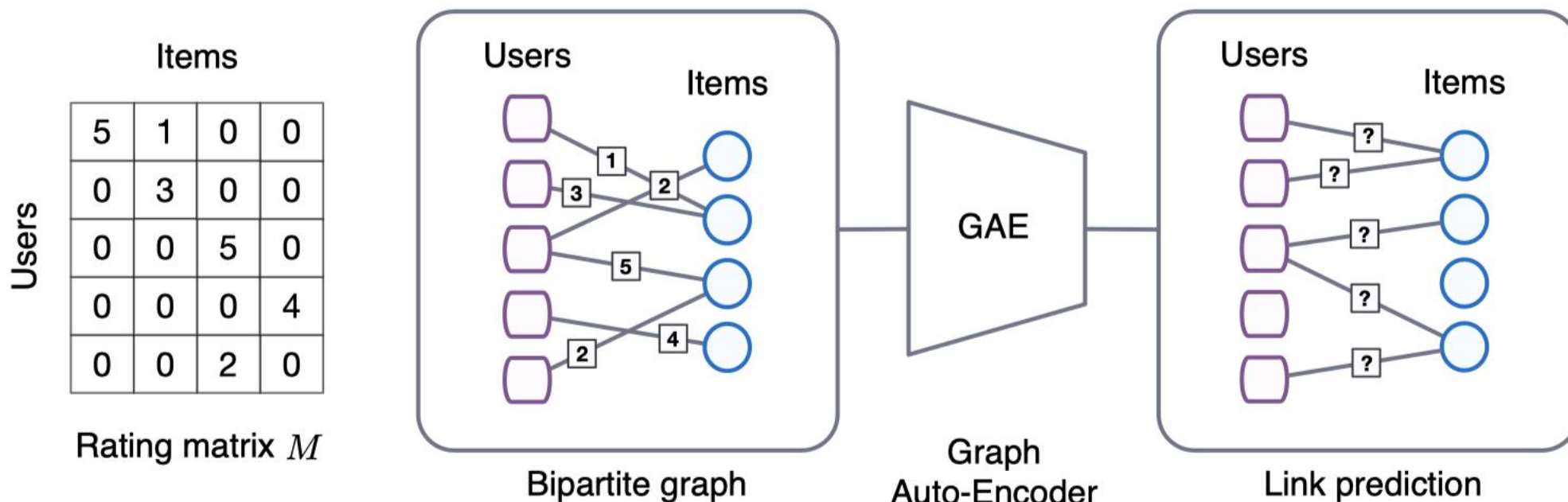
02 基于图神经网络的 推荐算法

Graph Convolutional Matrix Completion

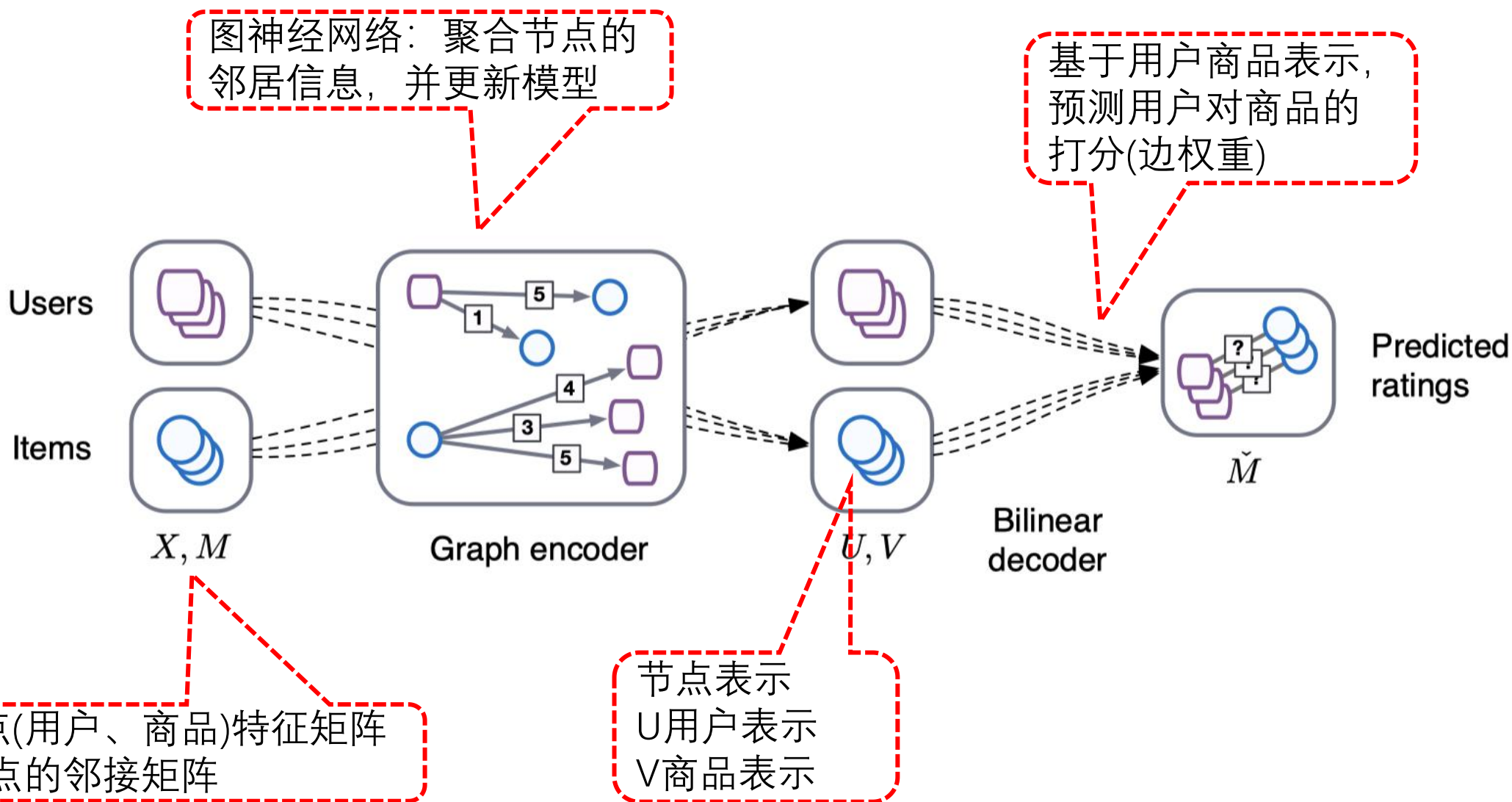
Rianne van den Berg
University of Amsterdam

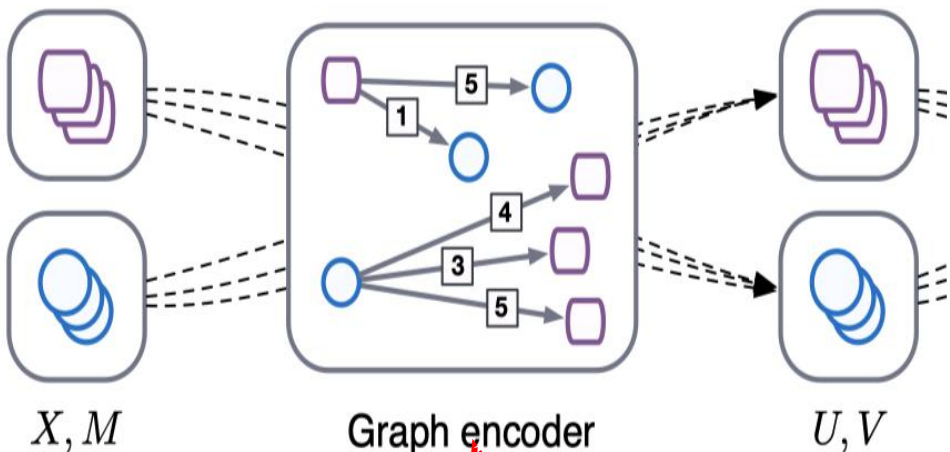
Thomas N. Kipf
University of Amsterdam

Max Welling
University of Amsterdam, CIFAR¹



基于图神经网络的推荐算法





图神经网络：聚合节点的邻居信息，并更新模型

用户表示：聚合其邻居(商品)来学习

- 用户买过的商品会有不同的打分 r ，分数 r 的高低代表了用户对商品的喜欢程度。
- 基于打分为 r 的商品聚合，得到的用户 i 的表示为 h_i^r

邻居聚合

$$h_i^r = \sum_{j \in \mathcal{N}_{i,r}} W_r x_j$$

其中， $\mathcal{N}_{i,r}$ 代表用户打分为 r 的商品集合， W_r 是针对打分 r 的投影矩阵。

- 考虑用户所有打过分的商品 $r = 1, \dots, R$ ，得到 R 个用户 i 的表示。

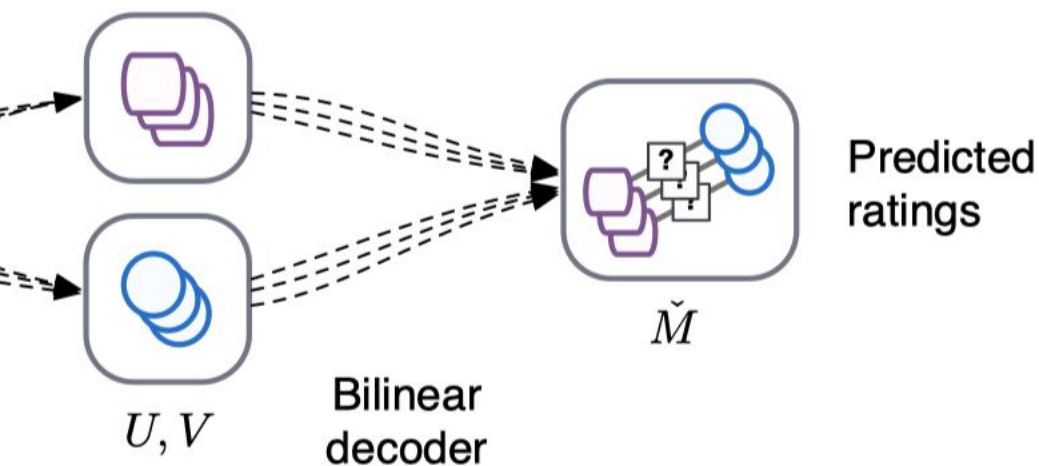
$$h_i^1, \dots, h_i^R$$

- R 个表示的融合(例如拼接)并进行非线性变换，得到最终用户 i 的表示 u_i

$$h_i = h_i^1 || \dots || h_i^R$$

$$u_i = \sigma(Wh_i)$$

商品表示：聚合其邻居(用户)来学习。与用户表示学习的方式相同。商品 j 的表示为 v_j



基于用户商品表示,
预测用户对商品的
打分(边权重)

评分预测

用户 i 的表示 u_i + 商品 j 的表示为 v_j \rightarrow 用户 i 对商品 j 打分预测分为 \tilde{M}_{ij} 。

$$\tilde{M}_{ij} = g(u_i, v_j) = \sum_{r \in R} r \cdot p(\tilde{M}_{ij} = r)$$

$$p(\tilde{M}_{ij} = r) = \frac{e^{u_i^T Q_r v_j}}{\sum_{s \in R} e^{u_i^T Q_s v_j}}$$

其中, Q_r 是针对评分 r 的参数矩阵。

模型训练

在已知用户 i -商品 j 交互数据上 ($\Omega_{ij} = 1$), 计算预测评分的损失函数

$$\mathcal{L} = - \sum_{i,j; \Omega_{ij}=1} \sum_{r=1}^R I[r = M_{ij}] \log p(\tilde{M}_{ij} = r)$$

Dataset	Users	Items	Features	Ratings	Density	Rating levels
Flixster	3,000	3,000	Users/Items	26,173	0.0029	0.5, 1, ..., 5
Douban	3,000	3,000	Users	136,891	0.0152	1, 2, ..., 5
YahooMusic	3,000	3,000	Items	5,335	0.0006	1, 2, ..., 100
MovieLens 100K (ML-100K)	943	1,682	Users/Items	100,000	0.0630	1, 2, ..., 5
MovieLens 1M (ML-1M)	6,040	3,706	—	1,000,209	0.0447	1, 2, ..., 5
MovieLens 10M (ML-10M)	69,878	10,677	—	10,000,054	0.0134	0.5, 1, ..., 5

Model	ML-1M	ML-10M
PMF [20]	0.883	—
I-RBM [26]	0.854	0.825
BiasMF [16]	0.845	0.803
GC-MC (Ours)	0.832	0.777

基于图神经网络的推荐算法

Dataset	Users	Items	Features	Ratings	Density	Rating levels
Flixster	3,000	3,000	Users/Items	26,173	0.0029	0.5, 1, ..., 5
Douban	3,000	3,000	Users	136,891	0.0152	1, 2, ..., 5
YahooMusic	3,000	3,000	Items	5,335	0.0006	1, 2, ..., 100
MovieLens 100K (ML-100K)	943	1,682	Users/Items	100,000	0.0630	1, 2, ..., 5
MovieLens 1M (ML-1M)	6,040	3,706	—	1,000,209	0.0447	1, 2, ..., 5
MovieLens 10M (ML-10M)	69,878	10,677	—	10,000,054	0.0134	0.5, 1, ..., 5