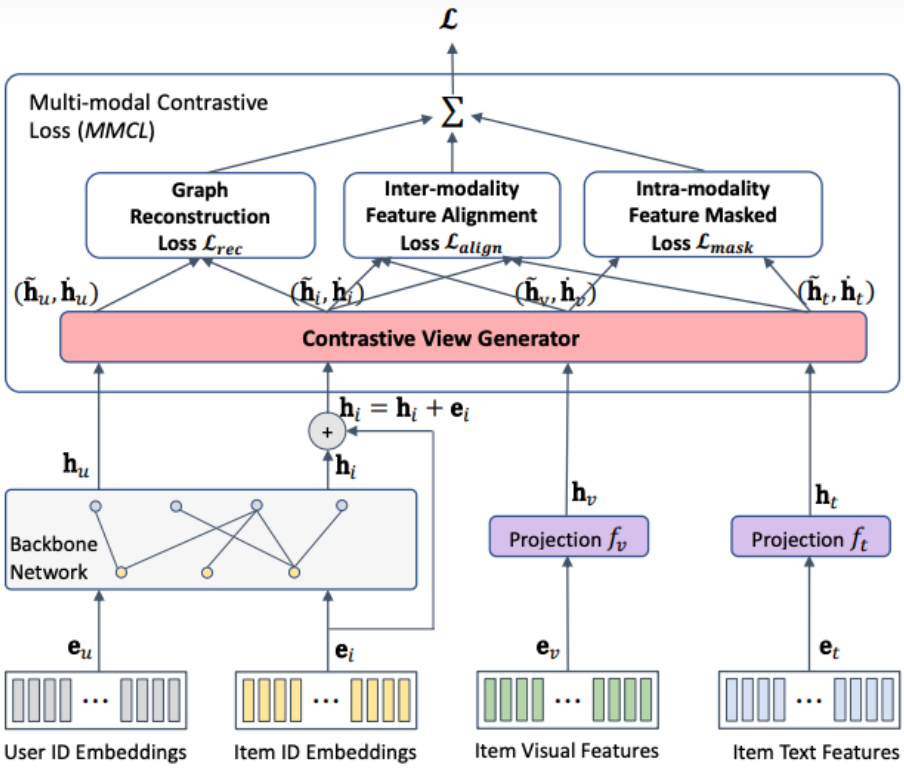
知乎



阿里-2023：BM3 基于自我监督学习的多模态推荐

SmartMindAI
专注搜索、广告、推荐、大模型和人工智能最新技术，欢迎关注我

已关注

2 人赞同了该文章

Introduction

在电子商务中，深度学习技术广泛用于推荐系统。为提高推荐准确率，近期多模态推荐工作探讨了如何将多模态信息有效地融入传统推荐范式。其中一种方法是结合物品潜在表示和多模态特征；另一种方法则是使用注意力机制捕捉用户对物品多模态特征的喜爱。同时，随着图基础推荐的研究兴起，也有学者采用图形神经网络利用多模态信息增强用户和物品表示的学习。例如，在用户-商品交互图上，使用图卷积网络分别进行传播和聚合不同多模态信息。此外，还利用辅助图结构，如用户-用户关系图和商品-商品关系图，以提升从多模态信息中学习用户和物品表示的能力。

首先，它们通常依赖于基于对偶排序损失的学习用户和项目表示（如BPR损失），其中观测到的用户-项目交互对被视为正例，随机采样的用户-项目对被视为负例，但这可能导致大规模图上的大量成本，并且可能引入训练过程中的不稳定的监督信号（例如，LightGCN中默认的均匀采样在每个周期消耗超过25%的训练时间）。其次，使用辅助图结构的方法在构建或训练大型辅助图时可能会产生无法接受的内存成本。有关现有图基多模态方法的计算复杂性的更多信息可在表1和表2中找到。SSL可以解决无负样本情况下学习用户和项目表示的问题，已经在CV和NLP等领域得到证明，且效果可与监督学习媲美甚至更好。SSL利用在线网络和目标网络两个不对称网络来最大化不同版本样本相似性，但仅使用正样本训练会导致模型陷入简单常数解。

因此， BYOL和SimSiam引入了一个额外的"预测器"网络到在线网络，并对目标网络执行特殊的"停止梯度"操作。BUIR将这种方法应用于推荐领域并在评估数据集上显示出竞争力的表现。我们提出了一种新的无监督学习方法 BM3 ，用于多模态推荐。在 BM3 中，我们使用Dropout机制来生成用户和项目视图，而不需要负样本来训练模型。为了在没有负样本来训练的情况下训练 BM3 ，我们设计了一个MMCL函数，该函数能够同时优化用户-项目交互图重建损失，以及减少不同增强方法之间的学习特征的不一致性。我们还在3个不同规模的数据集上验证了 BM3 的有效性和效率。结果显示， BM3 在性能上优于现有的多模态推荐方法，并且在训练速度上比基准方法快2-9倍。

Bootstrapped Multi-modal Model

知乎

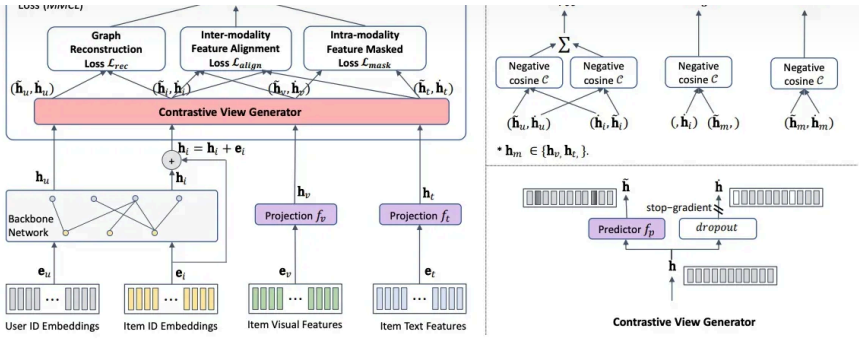


Figure 1: The structure overview of the proposed BM3. Projections f_v and f_t, as well as predictor f_p, are all one-layer MLPs. The parameters of predictor f_p are shared in the Contrastive View Generator (*bottom left*) for ID embeddings and multi-modal latent representations.

本节详述Bootstrap的多模态模型，模型包含3个组件（a) 多模态潜在空间转换器；(b) 对比视图生成器✦；(c) 多模态对比损失。

Multi-modal Latent Space Convertor

假设 $\mathbf{e}_u, \mathbf{e}_i \in \mathbb{R}^d$ 是用户 u 和物品 i 的输入 ID 嵌入，其中 d 是嵌入维度。 $\mathcal{U}$ 和 $\mathcal{I}$ 分别表示用户集合和物品集合。它们的基数分别为 $|\mathcal{U}|$ 和 $|\mathcal{I}|$。

我们通过模态特定特征从预训练模型✦中获取到 $\mathbf{e}_m \in \mathbb{R}^{d_m}$，其中 $m \in \mathcal{M}$ 表示一种特定的模态，取自完整的模态集合 $\mathcal{M}$ 中的一个，并且 d_m 表示特征的维度。 $\mathcal{M}$ 的基数记作 $|\mathcal{M}|$。 在本论文中，我们将考虑两种模态。

Multi-modal Features

给定多模态特征向量 $\mathbf{e}_m$，使用 MLP 投影函数 f_m 将其投影到低维的潜在空间，得到最终结果。

$$\mathbf{h}_m = \mathbf{e}_m \mathbf{W}_m + \mathbf{b}_m,$$

MLP中的线性变换矩阵✦$\mathbf{W}_m$和偏置向量$\mathbf{b}_m$构成了一个空间模型。

ID Embeddings

$$\mathbf{H}^{l+1} = \sigma\left(\hat{\mathbf{A}}\mathbf{H}^l\mathbf{W}^l\right),$$

$$\mathbf{H}^{l+1} = (\mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2})\mathbf{H}^l,$$

在第$(l+1)$层隐藏层的节点嵌入是通过线性聚合第l层嵌入得到的。该过渡矩阵为加权邻接矩阵。我们使用读取函数汇总所有隐藏层的表示以获得用户和项目的最终表示。但是，GCNs可能会遇到过平滑问题。为了克服这个问题，我们可以按照LATTICE的方法，在项目初始嵌入$\mathbf{H}_i^0$上添加残差连接✦以获取其最终表示。即：

$$\mathbf{H}_i^k = \mathbf{H}_i^{k-1} + \tilde{\mathbf{H}}_i$$

$$\mathbf{H_u} = \mathrm{R}\{\scriptsize\mathbf{EADOUT}\}(\mathbf{H}_u^0, \mathbf{H}_u^1, \mathbf{H}_u^2, \ldots, \mathbf{H}_u^L);$$
$$\mathbf{H_i} = \mathrm{R}\{\scriptsize\mathbf{EADOUT}\}(\mathbf{H}_i^0, \mathbf{H}_i^1, \mathbf{H}_i^2, \ldots, \mathbf{H}_i^L) + \mathbf{H}_i^0,$$

其中，$\mathbf{READOUT}$函数是一个可微分的函数，我们将其设定为LightGCN模型的默认均值函数作为最终的ID嵌入更新。 通过多模态潜在空间转换器，我们可以得到三种类型的潜在表示：用户ID嵌入、项ID嵌入和单模态项嵌入。 接下来部分将展示**BM3**设计的损失，以便高效地优化参数，而无需负样例。

Multi-modal Contrastive Loss

前研究使用停梯度策略防止模型得到常数解。同时，他们采用在线与目标网络，使模型参数按师生模式学习。本文通过将数据增强推迟至在线网络编码后简化SSL范例。首先说明数据增强。

知乎

使用图增强技术生成原始图的辅助视图，以供自我监督学习使用。输入特征通过两种图形进行编码，并生成对比视图。为降低计算复杂度和内存成本，采用类似节点dropout的简单潜在嵌入丢弃技术，不需求助图增强。对比潜在嵌入$\dot{\mathbf{h}}$由$\mathbf{h}$计算得出，其概率值为p。

$$\dot{\mathbf{h}} = \mathbf{h} \cdot \text{Bernoulli}(p).$$

对比视角$\dot{\mathbf{h}}$的应用，以及将原始的嵌入向量$\mathbf{h}$输入到MLP预测器中。

$$\tilde{\mathbf{h}} = \mathbf{h}\mathbf{W}_p + \mathbf{b}_p,$$

W_p, b_p分别为预测函数$^+$ f_p的线性变换矩阵和偏置。

Graph Reconstruction Loss

在输入用户-物品对时，**BM3**生成对比视图。损失函数$^+$是两个在线表示之间的负余弦相似度$^+$。

$\mathcal{L}_{rec} = \mathcal{C}(\tilde{\mathbf{h}}_u, \dot{\mathbf{h}}_i) + \mathcal{C}(\dot{\mathbf{h}}_u, \tilde{\mathbf{h}}_i)$. 函数$\mathcal{C}(x, y) = \frac{(x-y)^2}{4}$

$$\mathcal{C}(\mathbf{h}_u, \mathbf{h}_i) = -\frac{\mathbf{h}_u^T \mathbf{h}_i}{||\mathbf{h}_u||_2 ||\mathbf{h}_i||_2},$$

使用$||\cdot||_2$表示ℓ_2范数$^+$。目标是最大化对于给定用户的正向扰动项i的预测，并且反过来也是如此。最小可能的值是-1。最后，强制反向传播损失只在线性网络中进行。我们按照停止梯度（sg）的操作方式更新方程。

$$\mathcal{L}_{rec} = \mathcal{C}(\tilde{\mathbf{h}}_u, sg(\dot{\mathbf{h}}_i)) + \mathcal{C}(sg(\dot{\mathbf{h}}_u), \tilde{\mathbf{h}}_i).$$

在停止梯度操作中，目标网络不利用$(\dot{\mathbf{h}}_u, \dot{\mathbf{h}}_i)$计算梯度。

Inter-modality Feature Alignment Loss

我们将物品的多模态特征与目标ID对齐。对齐使ID在具有相似多模态特征的物品上更接近。对于每个物品的单模态潜在嵌入$\mathbf{h}_m$，我们比较视图生成器输出的对比对$(\tilde{\mathbf{h}}_m^i, \dot{\mathbf{h}}_m^i)$。我们使用负余弦相似度来衡量$\dot{\mathbf{h}}_i$和$\tilde{\mathbf{h}}_m^i$之间的对齐程度。

$$\mathcal{L}_{align} = \mathcal{C}(\tilde{\mathbf{h}}_m^i, \dot{\mathbf{h}}_i).$$

Intra-modality Feature Masked Loss

最后， BM3 的损失函数考虑了内部模式特征被遮蔽的情况。为了使模型学习到稀疏的嵌入表示，我们在大规模变换器中进行了验证。将潜在向量$\mathbf{h}_m$的一部分随机标记为稀疏嵌入$\dot{\mathbf{h}}_m^i$，并对内部模式特征被遮蔽的损失进行了定义：

$$\mathcal{L}_{mask} = \mathcal{C}(\tilde{\mathbf{h}}_m^i, \dot{\mathbf{h}}_m^i).$$

我们添加了正则化惩罚在线性模型$^+$的输入和隐藏层。最终损失函数为...

$$\mathcal{L} = \mathcal{L}_{rec} + \mathcal{L}_{align} + \mathcal{L}_{mask} + \lambda \cdot (||\mathbf{h}_u||_2^2 + ||\mathbf{h}_i||_2^2).$$

Top-K Recommendation

$$s(\mathbf{h}_u, \mathbf{h}_i) = \tilde{\mathbf{h}}_u \cdot \tilde{\mathbf{h}}_i^T.$$

高分度量了用户对物品的偏好程度。

Computational Complexity

知乎

$\mathcal{O}(\sum_{m\in\mathcal{M}}|\mathcal{I}|d_m d)$

和对比损失的成本

$\mathcal{O}((2+2|\mathcal{M}|)dB)$。

因此， BM3 的总计算复杂度为

$\mathcal{O}(2L|\mathcal{E}|d/B+\sum_{m\in\mathcal{M}}|\mathcal{I}|d_m d+(2+2|\mathcal{M}|)dB)$。

相比于MMGCN和DualGNN，LATTICE通过构建项项图来进行多模态特征投影，但其时间复杂度较高：构建项项之间的相似矩阵需要$\mathcal{O}(|\mathcal{I}|^2 d_m)$的时间，规范化矩阵需要$\mathcal{O}(|\mathcal{I}|^3)$的时间，获取每个项的前$k$个最相似的项则需要

$\mathcal{O}(k|\mathcal{I}|\log(|\mathcal{I}|))$

的时间。

Experiments

Experimental Datasets

根据先前研究，我们使用公开的亚马逊评论数据集进行实验评估。该数据集提供了产品描述和图片，大小因产品类别而异。为进行大规模评估，我们选择了三种品类：婴儿、运动和户外（以Sports表示）以及电子产品。

Table 2: Statistics of the experimental datasets.

Datasets	# Users	# Items	# Interactions	Sparsity
Baby	19,445	7,050	160,792	99.88%
Sports	35,598	18,357	296,337	99.95%
Electronics	192,403	63,001	1,689,188	99.99%

每评价评分被视为一次用户与产品的正向交互记录。这种设定已广泛应用于以往的研究中。原始数据经过五核处理，包括每个商品和每个用户的交互数，过滤后的结果已在表中呈现。数据稀疏性由互动数✦除以用户数乘以商品数衡量。

预处理后的数据集包含视觉和文本两个模态。对于视觉模态，我们使用了已在GitHub上公开发布✦的包含4096维视觉特征的数据集。对于文本模态，我们通过将每个项目的标题、描述、类别和品牌组合在一起，并使用sentence-transformers获取每个项目384维句子嵌入来提取文本嵌入。

Baseline Methods

对比了，包括通用的CF模型和多模态模型✦，以验证其有效性。

- BPR: 这是一种基于贝叶斯的排序损失优化的矩阵分解模型。
- "LightGCN" 这是一个简化的图卷积网络✦，只执行线性传播和聚合 邻居之间的隐藏层嵌入被平均以计算最终的用户和项目的嵌
- VBPR 是 Variational Bayesian PRing 的缩写，这是一种机器学习方法。
- MMGCN：一种构建模态特定图的方法，使用GCN学习各模态用户的偏好，最后通过组合各模态表示生成用户和项目的表示。
- GRCN 是一种改进的GCN方法，它通过删除假正向边来提高性能。该方法使用优化的二元图来表示用户和物品，并通过信息传播和聚合进行学习。
- "双GNN"方法：该方法从用户-物品二分图✦构建额外的用户用户相关联图，并使用它来融合用户在相关联图中邻近用户的表示。

将前三个基线作为通用模型，这些模型仅依赖用户-项目交互作为推荐依据。其他多模态模型则结合了用户-项目交互和多模态特征。BUIR被归类为自我监督模型和剩余的模型被归类为监督模型，因为前者通过负样本进行学习，后者需要标注数据。提出的 BM3 模型属于自我监督多模态领域。

Setup and Evaluation Metrics

为了公平比较，我们采用8:1的评估标准。

Implementation Details

本文采用相同的用户/项目嵌入大小为64，并使用Xavier初始化嵌入参数。优化器设为Adam，学习率为0.001。参数的调节基于已发表论文，以保持公平的比较。 BM3 模型通过PyTorch实现。在所有数据集上进行网格搜索以寻找最优参数设置。GCN层数量在1和2之间可调。用于嵌入扰动的Dropout率在0.3和0.5之间可选，正则化系数在0.1和0.01之间。 考虑到收敛性，早期停止和总epochs都被固定为20和1000，分别。训练停止指标由验证数据上的R\@20决定。 我们的模型和所有基线已集成到统一的多模态推荐平台MMRec中。

Effectiveness of BM3 (RQ1)

本研究比较了不同推荐方法在三个数据集上的性能。提出的模型在每个数据集上均优于通用的推荐方法和最先进的多模态推荐方法。模型相较于最佳基线分别提高了3.68%，6.15%和20.39%的召回率⁺\@10。这证明了模型的有效性和优势，并且表明模型在大型图上的表现优于基线。第二，MMGCN等多模态推荐模型并非总能超越非模态模型。即使VBPR在所有数据集上优于其竞争对手（如BPR），在使用LightGCN作为下游CF模型的GRCN和DualGNN时并未显著提升性能。LATTICE使用了间接的方法，通过构建项-项关系图并执行图卷积操作，以利用多模态特性。可能的原因有两个：MMGCN、GRCN和DualGNN性能较差。

Table 3: Overall performance achieved by different recommendation methods in terms of Recall and NDCG. We mark the global best results on each dataset under each metric in boldface and the second best is underlined.

Datasets	Metrics	General models			Multi-modal models					
		BPR	LightGCN	BUIR	VBPR	MMGCN	GRCN	DualGNN	LATTICE	BM3
Baby	R@10	0.0357	0.0479	0.0506	0.0423	0.0378	0.0532	0.0448	0.0544	**0.0564**
	R@20	0.0575	0.0754	0.0788	0.0663	0.0615	0.0824	0.0716	0.0848	**0.0883**
	N@10	0.0192	0.0257	0.0269	0.0223	0.0200	0.0282	0.0240	0.0291	**0.0301**
	N@20	0.0249	0.0328	0.0342	0.0284	0.0261	0.0358	0.0309	0.0369	**0.0383**
Sports	R@10	0.0432	0.0569	0.0467	0.0558	0.0370	0.0559	0.0568	0.0618	**0.0656**
	R@20	0.0653	0.0864	0.0733	0.0856	0.0605	0.0877	0.0859	0.0947	**0.0980**
	N@10	0.0241	0.0311	0.0260	0.0307	0.0193	0.0306	0.0310	0.0337	**0.0355**
	N@20	0.0298	0.0387	0.0329	0.0384	0.0254	0.0389	0.0385	0.0422	**0.0438**
Electronics	R@10	0.0235	0.0363	0.0332	0.0293	0.0207	0.0349	0.0363	-	**0.0437**
	R@20	0.0367	0.0540	0.0514	0.0458	0.0331	0.0529	0.0551	-	**0.0648**
	N@10	0.0127	0.0204	0.0185	0.0159	0.0109	0.0195	0.0202	-	**0.0247**
	N@20	0.0161	0.0250	0.0232	0.0202	0.0141	0.0241	0.0248	-	**0.0302**

Efficiency of BM3 (RQ2)

我们在提供准确性比较的基础上报告了 BM3 对于基准的效率，以充分利用内存和训练时间。需要注意的是，所有模型首先在配备有12GB内存的GeForce RTX 2080 Ti上进行评估，如果无法装入12GB内存，则将模型升级到配备有32GB内存的Tesla V100 GPU。不同方法的效率摘要在表中。从表中，我们可以看到以下两个观察。 1. 从通用模型和多模态模型的角度来看，图模型⁺通常比经典CF模型消耗更多的内存。具体来说，经典CF模型需要用户和物品的一致表示学习所需的最小GPU内存成本。而图模型通常需要保留一个额外的用户-项目交互图用于信息传播和聚合。此外，图模型多模态推荐模型需要更多的内存，因为它使用用户-项目图和一般多模态特征。 2. 在图模型多模态推荐模型中， BM3 消费的内存较少或与其他基线相当。然而，它减少了每个轮次的训练时间的2到9倍。 与最好的基线相比， BM3 的训练时间需求仅为LATTICE的一半，且消耗的内存也为LATTICE的一半。尽管 BM3 使用LightGCN作为其骨干模型，但它并未引入LightGCN外部太多的额外成本，原因是 BM3 删除了负采样时间⁺并使用了更少的GCN层。

Table 4: Efficiency comparison of BM3 against the baselines.

Datasets	Metrics	General models			Multi-modal models					
		BPR	LightGCN	BUIR	VBPR	MMGCN	GRCN	DualGNN*	LATTICE	BM3
Baby	Memory (GB)	1.59	1.69	2.29	1.89	2.69	2.95	2.05	4.53	2.11
	Time (s/epoch)	0.47	0.99	0.77	0.57	3.48	2.36	7.81	1.61	0.85
Sports	Memory (GB)	2.00	2.24	3.75	2.71	3.91	4.49	2.81	19.93	3.58
	Time (s/epoch)	0.95	2.86	2.19	1.28	16.60	6.74	12.60	10.71	3.03
Electronics	Memory (GB)	3.69	4.92	10.13	6.20	14.54	17.38	8.85	11.21	6.23
	Time (s/epoch)	6.75	67.49	63.77	14.20	470.15	152.68	341.02	-	73.31

知乎

本文提出了一种新颖的自我监督学习框架，称为 BM3 ，用于多模态推荐。 BM3 消除了在模型中用户和物品之间互动所需的随机采样负例的要求。 为了生成自我监督学习中的对比性视图， BM3 利用一种简单且高效的潜在嵌入丢弃机制来扰动原始用户的和项目的嵌入。此外，还设计了一个基于多模态对比损失的学习范式。

发布于 2023-12-18 20:21 · IP 属地北京

监督学习　阿里巴巴集团　深度学习（Deep Learning）