重磅整理！推荐系统之深度召回模型综述（PART III）

深度传送门 1周前

以下文章来源于NewBeeNLP ，作者一块小蛋糕

NewBeeNLP

永远有料，永远有趣

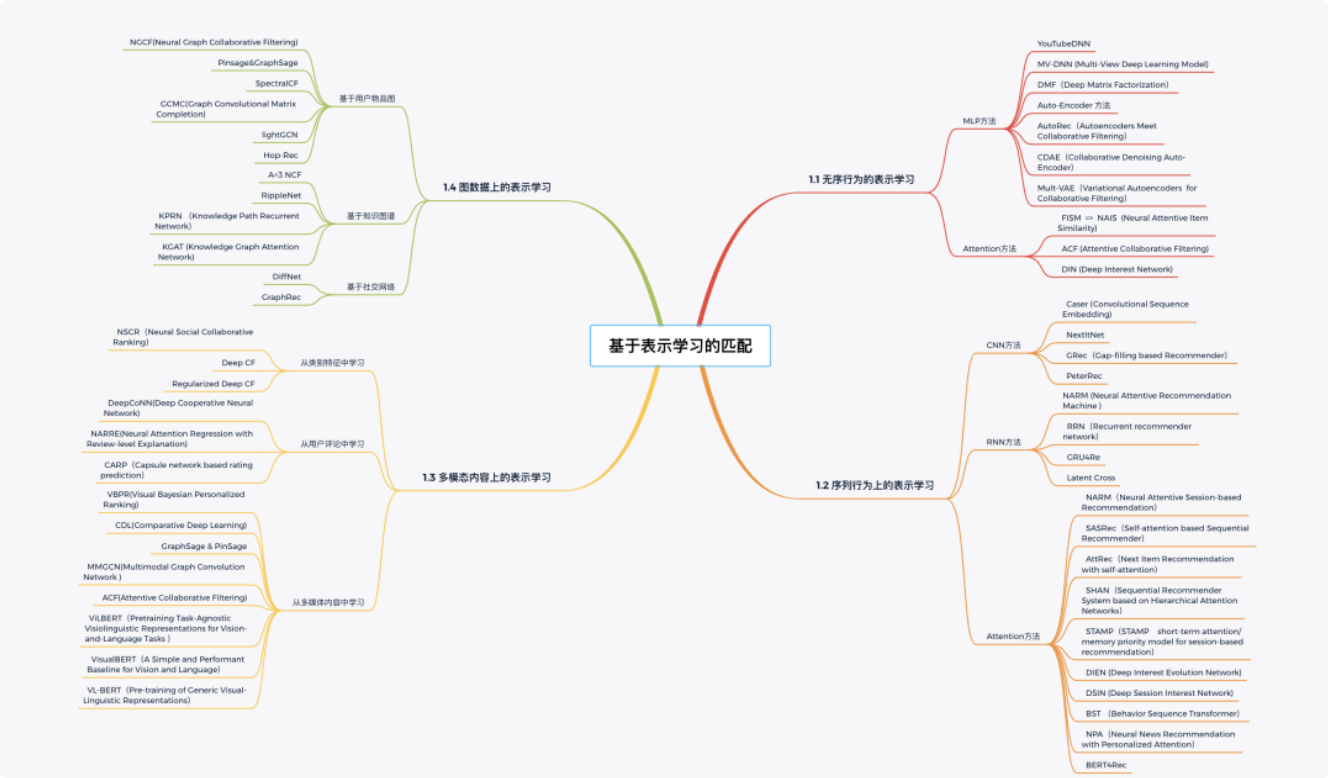
前段时间读完了李航、何向南的《Deep learning for matching in search and Recommendation》，文章思路清晰，总结详实到位，值得一再翻阅，就想借这篇文章结合自己最近一年多的推荐召回工作内容，总结一下推荐系统中的深度召回模型，论文因篇幅限制，很多模型并未详细介绍，因此本文补充了一些内容。

> 这篇综述文章现在好像不好下载，很多同学私信我，一个个发邮箱不太方便，现在大家可以直接在公众号后台回复『**DLM**』下载。

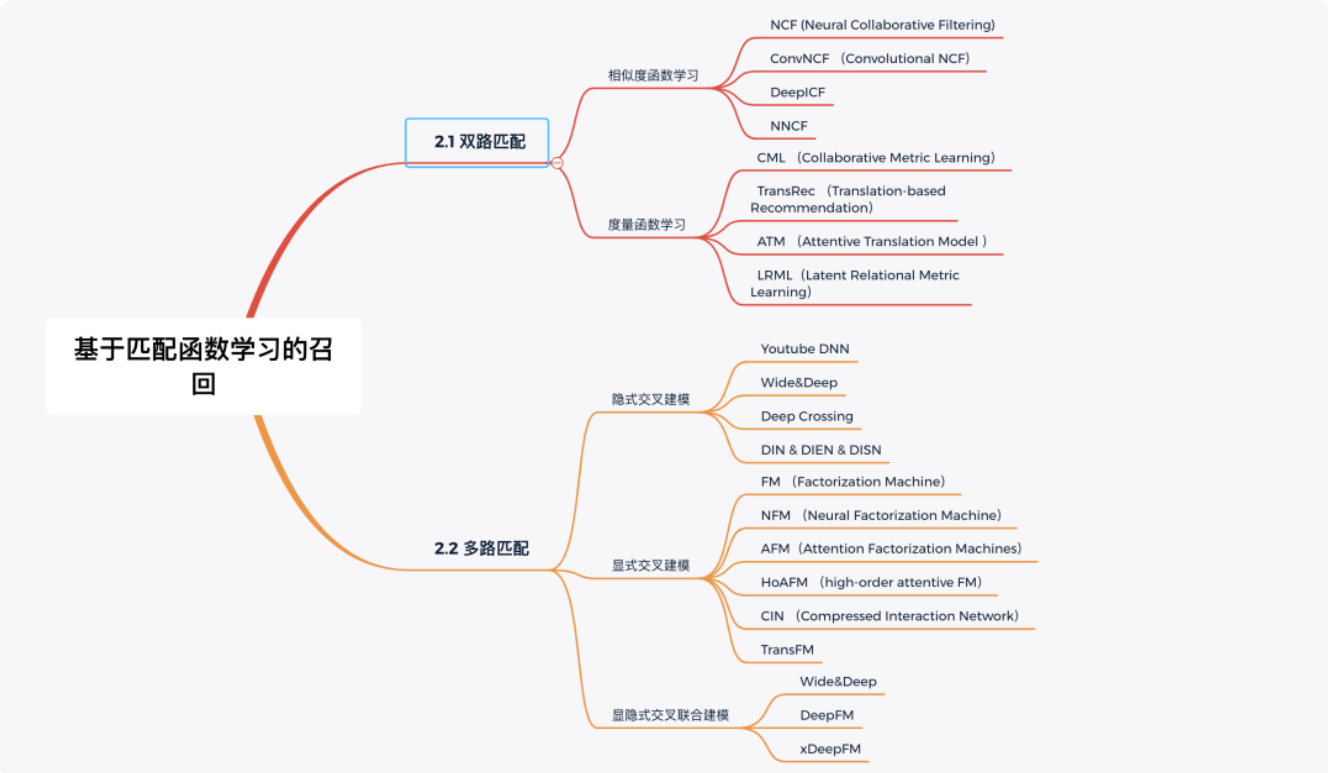
匹配（matching）是衡量用户对物品的兴趣的过程，也是推荐召回中的工作内容。机器学习中是以learning to match的方式根据输入表示和标记数据学习一个匹配函数。而深度学习在其发展过程中以强大的表示学习和泛化能力加上算力提升、数据规模暴涨都使得深度模型在推荐召回中大放异彩。

本系列文章的总结思路是将推荐中的深度召回模型根据学习内容分为两大类：**「表示学习类」**和**「匹配函数学习类」**。

- 表示学习类召回模型中根据输入数据的形式和数据属性又可以分为无序交互行为类、序列化交互行为类、多模态内容类和连接图类。



- 匹配函数学习类模型则包括双路匹配函数和多路匹配函数的学习。



本文是推荐系统总结之深度召回模型的第二篇，配合以下文章食用效果更佳：

- 重磅整理！推荐系统之深度召回模型综述（PART I）

- 重磅整理！推荐系统之深度召回模型综述（PART II）

1.3 多模态内容上的表示学习

除了用户物品的交互行为，用户和物品通常还有一些描述性特征如类别属性（年龄，性别，产品品类等）和文本特征。尤其一些多模态推荐系统中，图片、视频、音频及对应的描述性特征都可以帮助更好的学习用户/物品的表示。

多模态下表示学习可以抽象成：

$$\phi_u(u) = COMBINE(p_u, f(F_u))$$
$$\phi_i(i) = COMBINE(q_i, g(G_i))$$

其中 p_u 代表从用户的历史交互信息中学习到的Embedding，F_u 表示用户的side information，$f(.)$ 表示side information的表示学习函数。COMBINE(.，.)用于联结两边的信息，且g， f，COMBINE都可以是深度神经网络。

从类别特征中学习

NSCR (Neural Social Collaborative Ranking)

2017年新加坡国立大学何向南实验室提出 Neural Social Collaborative Ranking 方法，利用信息域即电商场景中的user-item交互和社交域如推特中的user-user连接进行跨域社交推荐，其中的假设是信息域的用户集和社交域的用户集存在交集，即桥梁用户，也是论文题目中的“丝绸之路”。

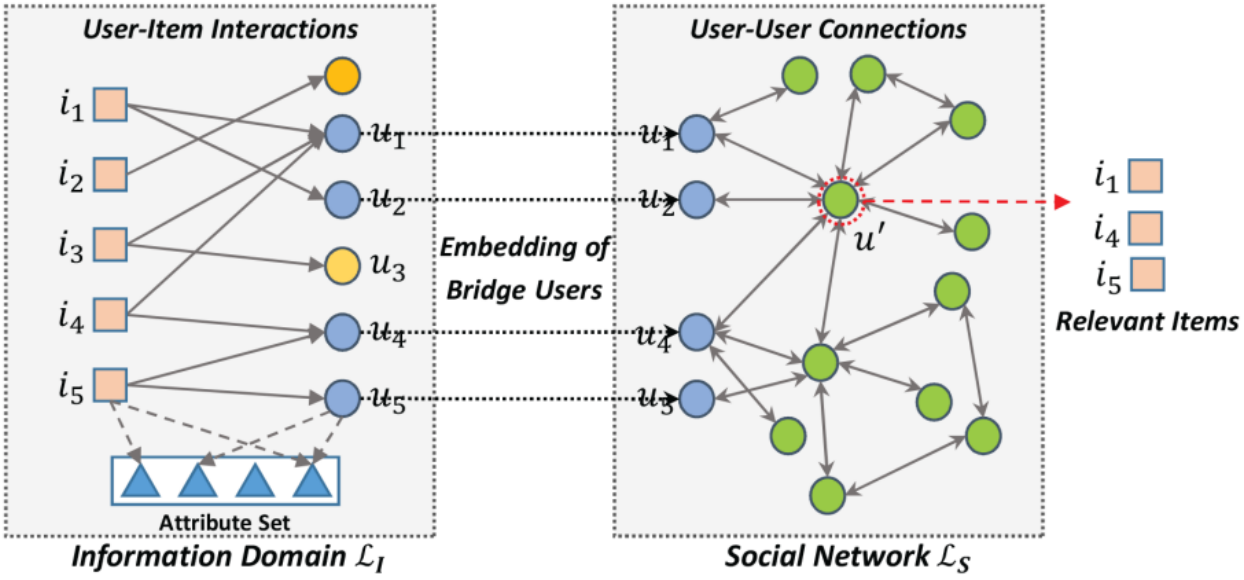


Figure 1: Illustration of the cross-domain social recommendation task.

由于桥梁用户数量通常不多，如果信息域和社交域以端到端的方式同时学习用户物品表示，会发生样本数严重不足的情况，因此NSCR将信息域和社交域的学习过程分隔开，各自优化不同的损失函数，如上图所示，先在利用丰富的用户物品特征信息及二者交互行为数据学习信息域中用户和物品的embedding表示；再将桥梁用户在信息域中的embedding通过社交网络传播出去以学习社交域中的非桥梁用户的表示。

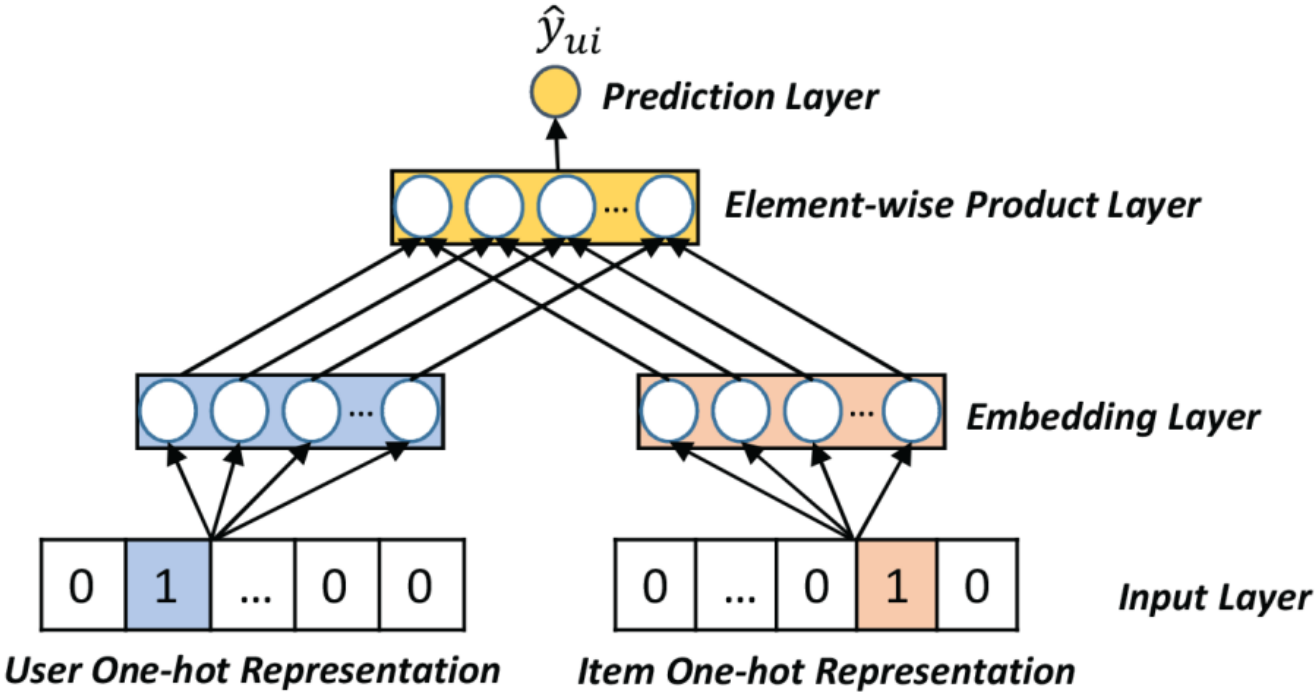


Figure 2: MF as a shallow neural network model.

首先信息域的表示学习过程可以应用任何协同过滤式方法，这里作者延续之前NCF中的观点，如上图所示，认为协同过滤的矩阵分解模型中用户和物品表示的内积交叉表达力不

足，而且SVD类模型对用户物品属性特征只是简单embedding相加，不能捕获高阶交叉信息。因此采用深度协同过滤NCF为基础架构，提出Attributed-aware DeepCF模型。

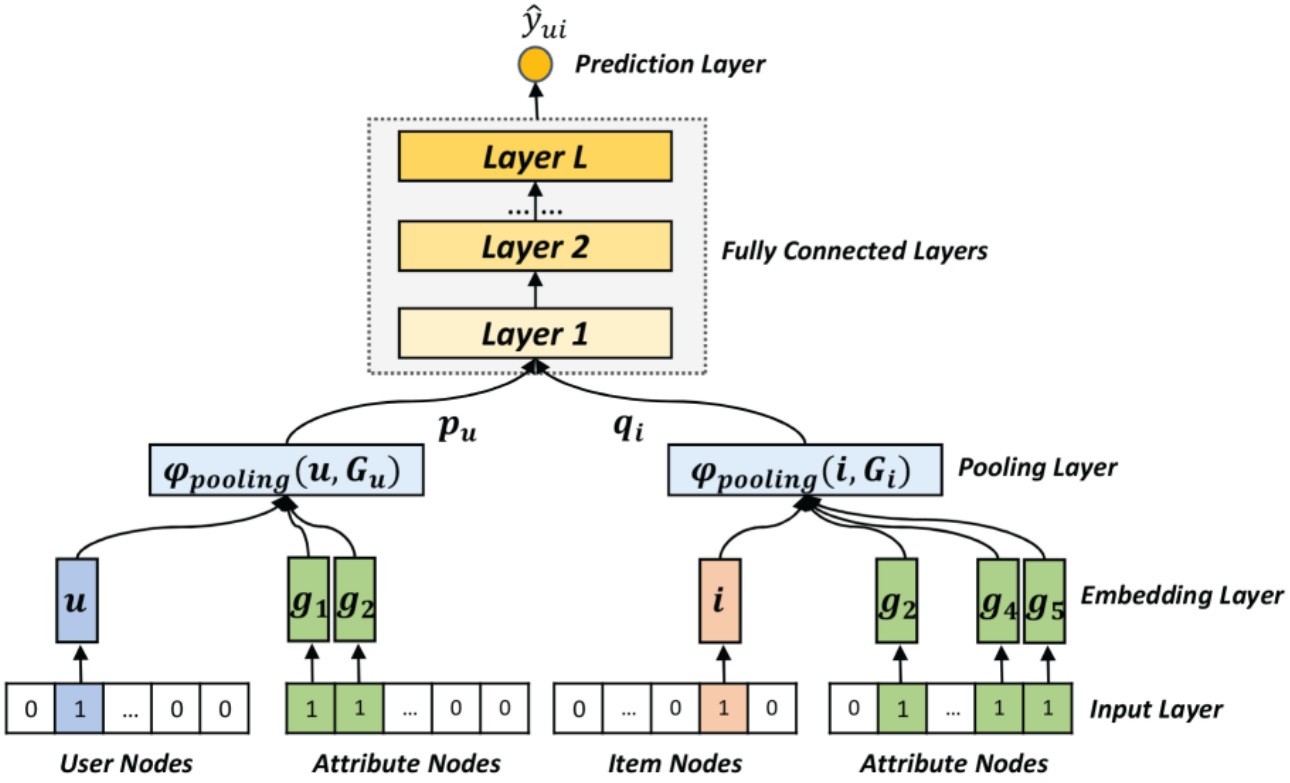


Figure 3: Illustration of our Attributed-aware Deep CF model for estimating an user-item interaction.

- **「输入层&Embedding层」**：输入用户和物品的id及各自属性特征的one-hot编码，经过Embedding层转换成低维稠密表示，即上图中的u, i, g_t^u, g_t^i
- **「Pooling层」**：由于属性特征数量不定，Embedding层之后的向量集维度也各不相同，因此需要Pooling操作，而常规的average和max Pooling因过于简单都不能捕获用户/物品和其属性特征之间的交叉信息，因此引入FM和NFM（也由该团队提出）模型中的pairwise Pooling操作：
  $p_u = \phi_{pairwise}(u, g_t^u) = \sum_{t=1}^{V_u} u \odot g_t^u + \sum_{t=1}^{V_u} \sum_{t'=t+1}^{V_u} g_t^u \odot g_{t'}^u$，其中 $\odot$ 表示两个向量的元素积，物品侧也是同样的操作，这里需要强调的是pairwise Pooling操作和max/average操作具有相同的复杂度，都是线性复杂度；
- **「hidden层」**：将上面用户物品两边的结果 $p_u \odot q_i$ 输入MLP
- **「prediction层」**：将MLP的结果转换成预测的分值：$\hat{y}_{ui} = w^T e_L$
- **「损失函数」**：使用pair-wise的目标排序函数，即学习正负样本的相对顺序而不是绝对分值，$L_I = \sum_{(u,i,j) \in O} (y_{uij} - \hat{y}_{uij}^2 = \sum_{(u,i,j) \in O} (\hat{y}_{ui} - \hat{y}_{uj} - 1)^2$，其中 $y_{uij} = y_{ui} - y_{uj}, \hat{y}_{uij} = \hat{y}_{ui} - \hat{y}_{uj}$

社交域的学习根据假设：若两个用户有很强的社交关联，那他们的偏好也是相似的，即有相似的表示，采用半监督的学习方式优化两个损失之和 $L_s = \theta(U_2) + \mu\theta(U)$：

- **「平滑约束」**：即图上位置相邻的点的表示差别不应太大，因此需要结构一致性损失：$\theta(U_2) = \frac{1}{2}\sum_{u',u''\in U_2} s_{u'u''}||\frac{P_{u'}}{\sqrt{d_{u'}}} - \frac{P_{u''}}{\sqrt{d_{u''}}}||^2$
- **「拟合约束」**：为了使桥梁用户在信息域和社交域中的表示相似，保持两个域的latent space一致，称为拟合约束：$\theta(U) = \frac{1}{2}\sum_{u'\in U}||P_{u'} - P_{u'}^{(0)}||^2$训练完成后将社交域中非桥梁用户的表示 $P_{u'}$ 与信息域中物品的表示 q_i 输入信息域的预测框架中，得到item的排序。

Deep CF

来自论文：A unified framework of representation learning and matching function learning

DeepCF将每个类别特征都映射成Embedding向量，然后和用户/物品的id Embedding做bi-interaction池化操作，池化后的用户和物品向量联合进入一层MLP，获得最终分值：

$$\phi_u(u) = BI - Interaction(p_u, f_t^{u\,V_u}_{t=1}) = \sum_{t=1}^{V_u} p_u \bigodot f_t^u + \sum_{t=1}^{V_u}\sum_{t'=t+1}^{V_u} f_t^u \bigodot f_{t'}^u$$

$$\phi_i(i) = BI - Interaction(q_i, g_{t\,t=1}^{i\,V_i}) = \sum_{t=1}^{V_i} q_i \bigodot g_t^i + \sum_{t=1}^{V_i}\sum_{t'=t+1}^{V_i} g_t^i \bigodot g_{t'}^i$$

$$\hat{y}_{ui} = MLP(\phi_u(u) \bigodot \phi_i(i))$$

其中，f_t^u, g_t^i 代表用户和物品属性的Embedding，V_u, V_i 表示用户和物品的属性数目。bi-interaction池化操作考虑了用户的id Embedding和属性Embedding之间的所有两两交叉。最后将联合得到的用户表示 $\phi_u(u)$ 和物品表示 $\phi_i(i)$ 做元素积交叉后送入MLP做最终的预测。这里MLP也可以替换成内积。

这个结构的优势在于能够很好的实现用户/物品各自属性的交叉和用户与物品之间的属性交叉。

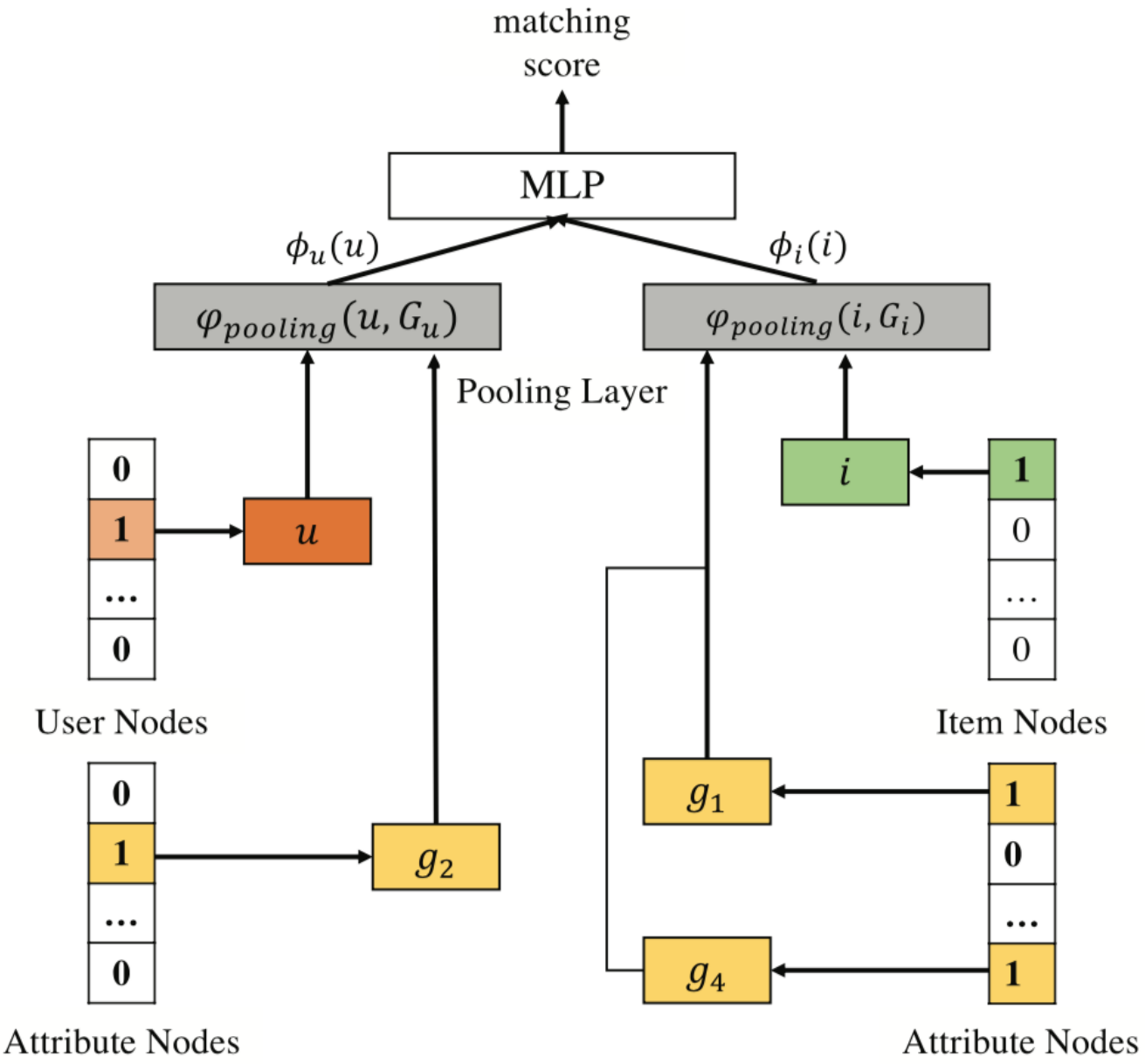


Figure 5.8: Model architecture of attribute-aware deep CF model.

Regularized Deep CF

这个模型的想法是先各自用自动编码机学习用户和物品特征的表示，再在推荐任务中联合训练用户/物品的表示。将自动编码机的损失视作推荐的正则项。左边的自动编码机用隐层U以 $L(X,U)$ 为损失从用户特征中学习到用户表示；右边的自动编码机用隐层V以 $L(Y,V)$ 为损失从物品特征中学习到物品表示；U和V以 (R,U,V) 为损失重建用户物品的打分矩阵。整个模型由这三个损失联合优化。

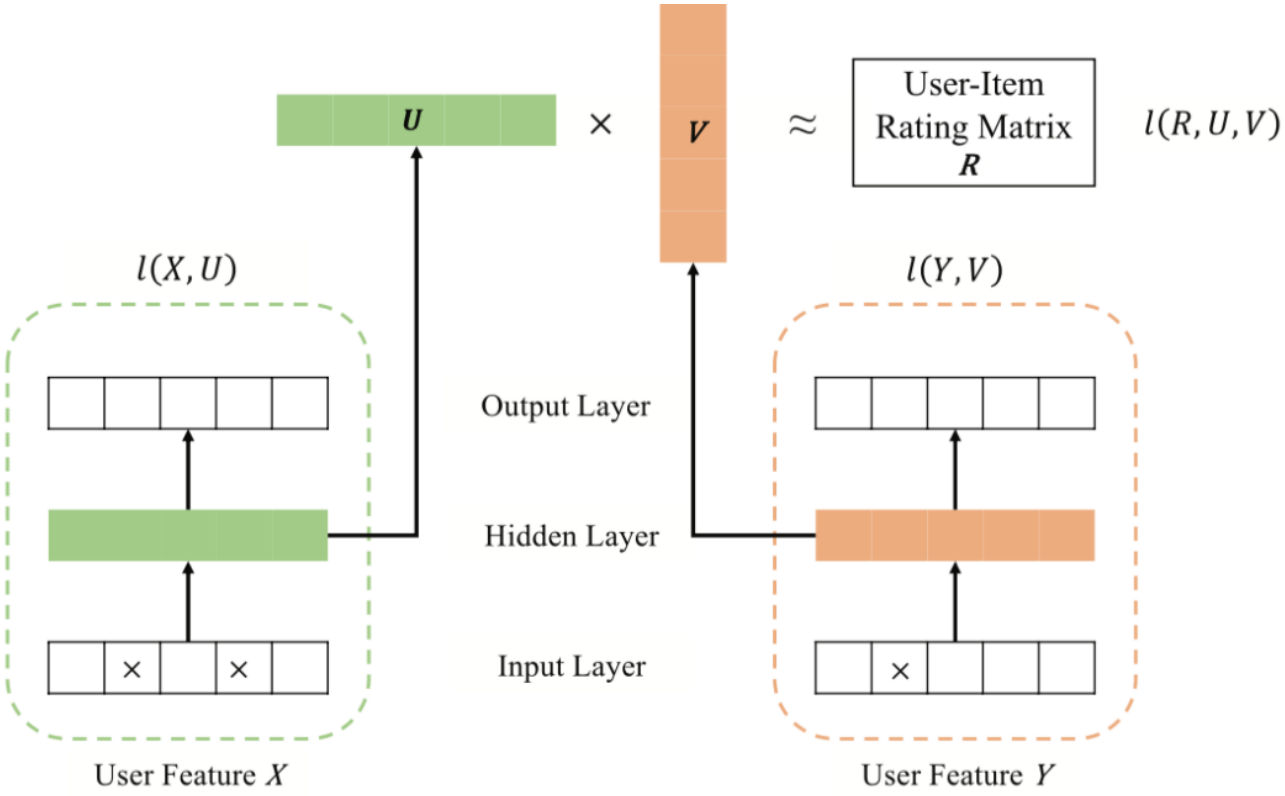


Figure 5.9: Model architecture of attribute-aware deep CF model.

从用户评论中学习

推荐系统中，其他用户的评论会显著影响用户在线购买的决策。利用评论中的信息不仅能帮助提高准确率还能提升推荐的可解释性。

DeepCoNN(Deep Cooperative Neural Network)

Deep Cooperative Neural Networks是从用户评论中联合学习物品属性和用户观点的模型。

如下图所示，包含两个并行的网络，一个从评论中学习用户观点；一个从评论中学习物品属性。这两个网络在最后一层合并实现联合学习。先将用户u写的所有评论整理成一个单独的具有n个词的文档 $d^u_{1:n}$，转换成词向量矩阵 $V^u_{1:n} = [\phi(d^u_1, \phi(d^u_2), \ldots, \phi(d^u_n))]$，再用一维CNN将词向量矩阵转换成表示向量：

$$x_u = Net_u(d^u_{1:n}) = CNN(V^u_{1:n})$$

对物品i也是同样的过程，首先将物品i的所有评论合并到一个m个词的文档中，转换成词向量矩阵后应用一维CNN，$x_i = Net_i(d^i_{1:m}) = CNN(V^i_{1:m})$。最后计算用户u和物品i的匹配分值，将 x_u 和 x_i 合并成一个向量 $z = [x^T_u, x^T_i]$，然后用FM计算：

$$y_{ui} = w_0 + \sum_{k=1}^{|z|} w_k z_k + \sum_{k=1}^{|z|} \sum_{l=k+1}^{|z|} w_{kl} z_k z_l$$

其中的 w_0, w_k, w_{kl} 都是FM的参数。

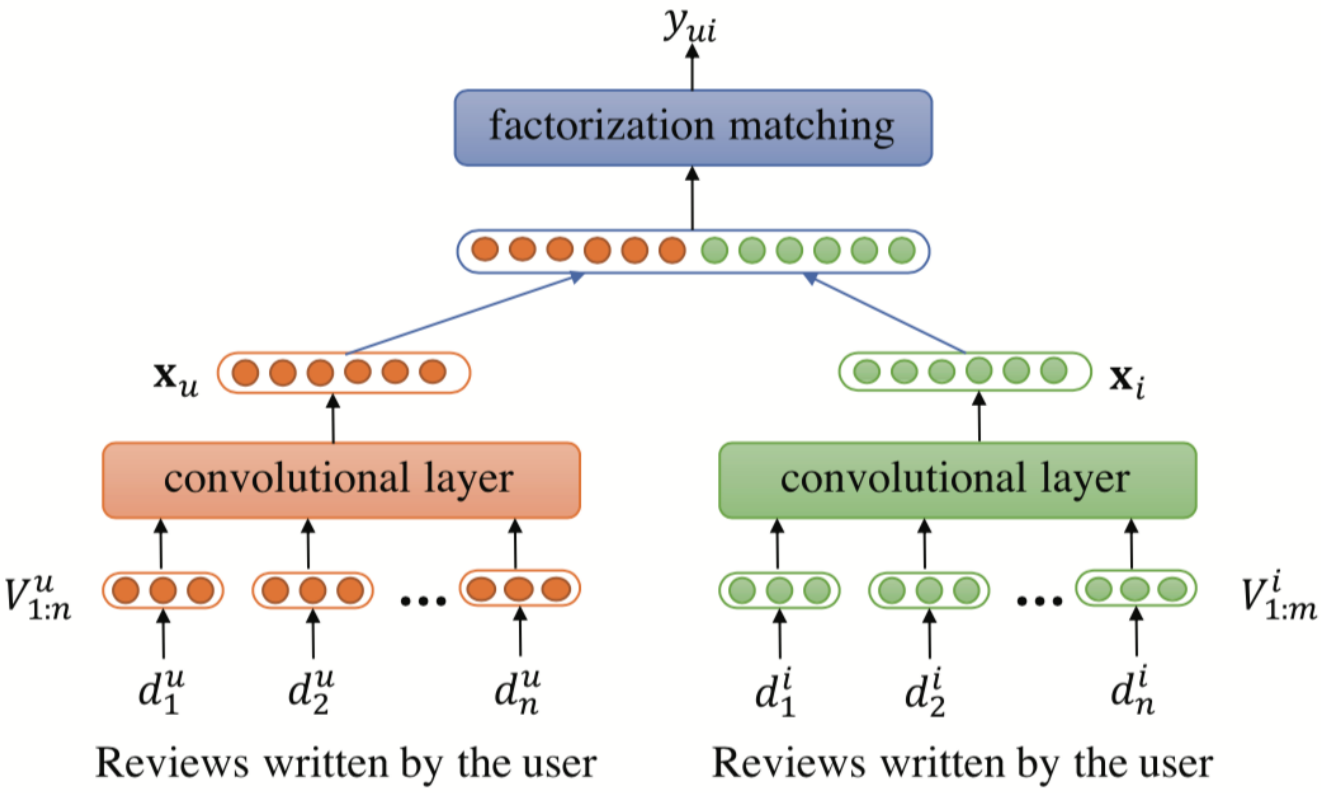


Figure 5.10: Model architecture of DeepCoNN.

NARRE(Neural Attention Regression with Review-level Explanation)

所有评论简单拼接意味着有价值和无价值的评论权重都相同，而事实往往并不是如此，因此NARRE将不同评论赋予不同的权重。

模型结构如下图所示。物品i的m条评论首先被转换成m个矩阵 $V_{i,1}, V_{i,2}, \ldots, V_{i,m}$，这些矩阵经过卷积层得到特征向量 $O_{i,1}, O_{i,2}, \ldots, O_{i,m}$。之后经过一个注意力池化层来聚合能描述物品i的有用评论。物品i的第l条评论的权重定义：

$$a_{i,l} = \frac{exp(a^*_{il})}{\sum_{k=1}^{m} exp(a^*_{ik})}$$

其中注意力权重：

$$a^*_{il} = h^T ReLU(W_o O_{i,l} + W_u u_{il} + b_1) + b2$$

u_{il} 是写下这第l条评论的用户u的embedding。最终物品i的表示为：

$$x_i = W_0 \sum_{l=1}^{m} a_{i,l} O_{i,l} + b_0$$

对于每个用户u写下的m条评论，其用户表示的计算过程也类似。NARRE中，计算最终用户物品匹配分值的预测层是一个扩展隐因子模型：

$$y_{ui} = W_1^T((q_u + x_u) \odot (p_i + x_i)) + b_u + b_i + \mu$$

其中 $\odot$ 是元素积，q_u, p_i 分别表示用户偏好和物品特征。

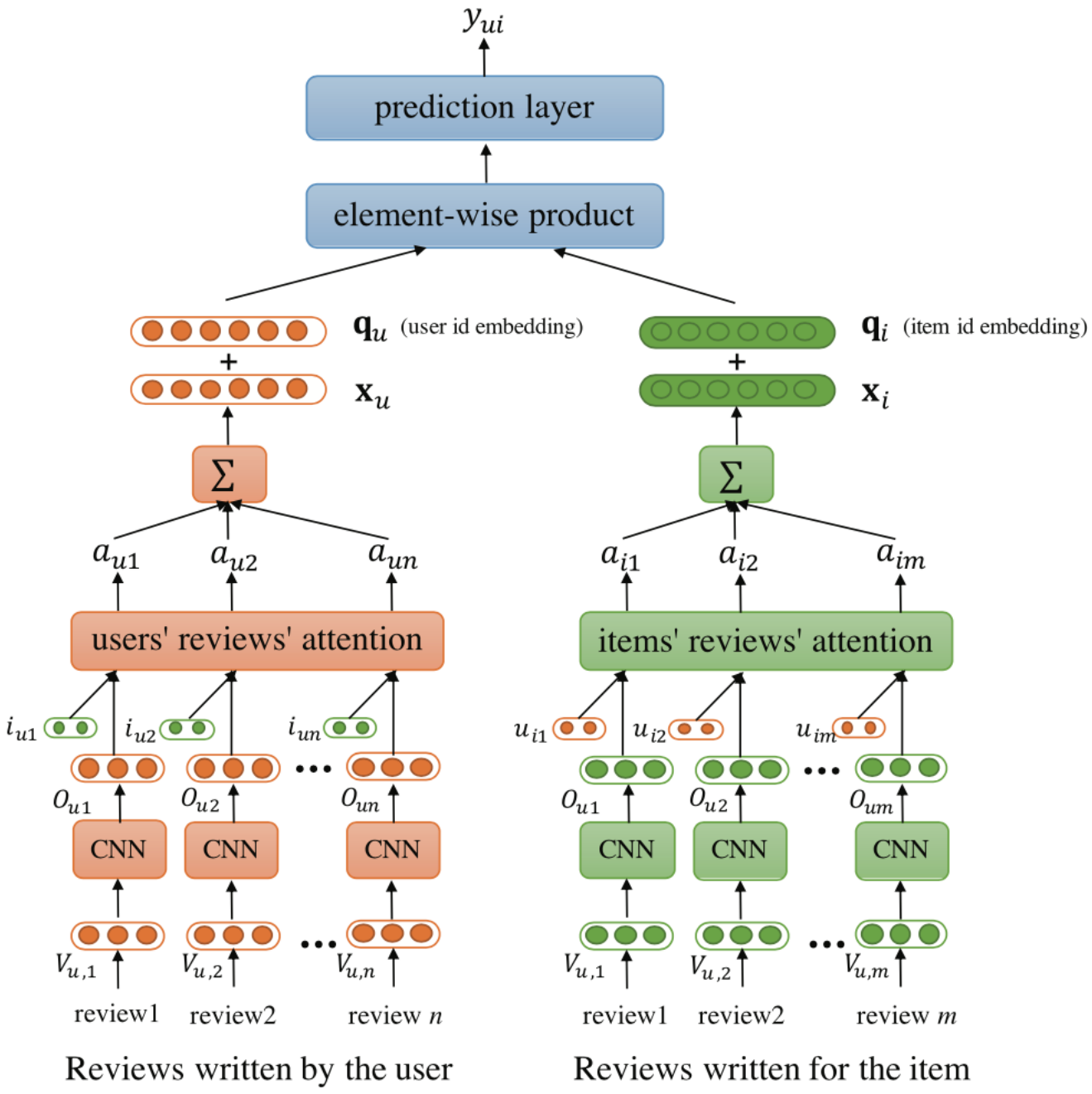


Figure 5.11: Model architecture of NARRE.

CARP (Capsule network based model for rating prediction)

2019年的CARP模型的作者认为单纯基于注意力机制在理解用户对某个item的喜好程度上还是不够精确，因此提出基于胶囊网络的用户评论分值预测模型。

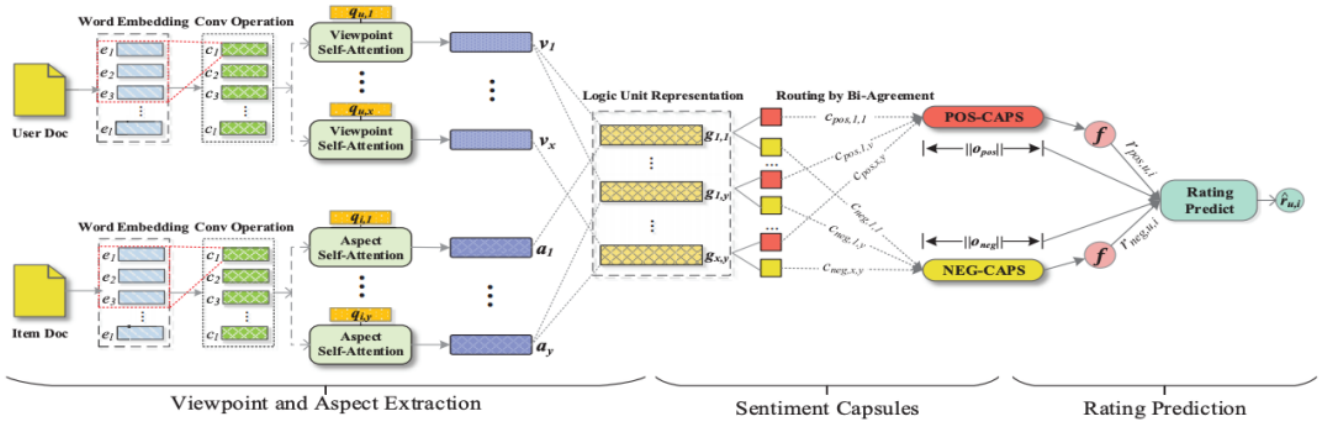


Figure 2: The network architecture of CARP.

模型结构如上图所示，分为三个部分：

- 「**viewpoint and aspect extraction**」：分别从用户和物品的评论内容中提取观点，首先文档经过Embedding层得到单词Embedding矩阵，再对每个单词和其两边 $\frac{c-1}{2}$ 窗口内的单词Embedding经过一个带ReLU的卷积操作得到其上下文表示c，由于并不是每个词都与主题观点相关，所以用一个self-attention对c重新分配权重：$s_{u,x,j} = c_j \bigodot \sigma(W_{x,1}c_j + W_{x,2}q_{u,x} + b_x)$，$W_{x,1}, W_{x,2}, b_x$ 是第x个观点的转换矩阵和偏置向量，$q_{u,x}$ 是所有用户共享的第x个观点的Embedding，是一个可学习向量。再通过一个观点共享的转换矩阵将 $s_{u,x,j}$ 映射提取为上下文观点的表示 $p_{u,x,j}$，$p_{u,x,j} = W_p s_{u,x,j}$。再以用户文档中各观点的上下文embedding均值作为观点的初级表示，$v_{u,x} = \frac{1}{l}\sum_j p_{u,x,j}$，然后计算观点间注意力权重：$attn_{u,x,j} = softmax(p_{u,x,j}^T v_{u,x})$，最终将用户观点表示 $v_{u,x}$ 重写为权重和 $v_{u,x} = \sum_j attn_{u,x,j} p_{u,x,j}$。以同样的方式从物品评论文档中提取M个aspect表示，当然参数是另一套。

- 「**sentiment Capsule**」：正常情况下，所有的用户观点表示和物品aspect表示都需要两两计算以得到用户评论决策背后的动机，如：$g_{x,y} = [(v_{u,x} - a_{i,y}) \bigoplus (v_{u,x} \bigodot a_{i,y})]$，文中将这样两两计算的设计叫逻辑单元，由一个用户观点和一个物品aspect表示和一个计算函数组成。但实际上全部两两组合带来的计算量非常高，也不全是有意义的，因此需要sentiment capsule部分筛选出有意义的组合。文中设计了两个胶囊网络，表示积极情绪和消极情绪，针对每个 $g_{x,y}$ 计算：$t_{s,x,y} = W_{s,x,y} g_{x,y}, s_{s,u,i} = \sum_{x,y} c_{s,x,y} t_{s,x,y}, o_{s,u,i} = \frac{||s_{s,u,i}||^2}{1+||s_{s,u,i}||^2} \frac{s_{s,u,i}}{||s_{s,u,i}||}$。作者还设计了一种双协议路由机制计算 $c_{s,x,y}$，以更快的迭代到两个胶囊的输出向量编码用户喜好/不喜欢物品的程度的状态。

- 「**Rating prediction**」：预测用户评分，$r_{s,u,i} = w_s^T h_{s,u,i} + b_{s,3}$CARP的优化过程中引入了多任务以加快优化速度，根据评分阈值给评论打上正面/负面评论的标签，增加一个二分类任务。

19年阿里提出的MIND模型也是将胶囊网络应用于推荐中多兴趣的学习，只是计算复杂度较高，只能在精排阶段使用，并不能用于召回环节，感兴趣的同学可以参考《深度推荐系统总结系列一》[1]

从多媒体内容中学习

CNN作为图片视频的有效特征提取器，被广泛应用于多媒体推荐中。

VBPR(Visual Bayesian Personalized Ranking)

使用DeepCNN从产品的图片中提取4096维特征向量 g_i 后，经过特征转换矩阵映射到协同过滤的Embedding空间中，$\theta_i = Eg_i$，θ_i 与物品id的Embedding q_i 拼接后得到物品的最终表示。

最后物品表示和用户表示通过内积交叉得到预测分值：$\hat{y_{ui}} = \phi_u(u)^T[q_i, Eg_i]$。这里省略了偏置项。模型使用pairwise的BPR损失优化。

VBPR中的Deep CNN是预训练好的特征提取器，在推荐任务中并不会更新。鉴于Deep CNN一般都是基于通用图片库如ImageNet训练而来，可能并不适合如服饰推荐的场景，因此下面的模型就是为了解决这个问题。

CDL(Comparative Deep Learning)

基于内容的图片推荐模型，端到端训练时同时更新Deep CNN和其他模型参数，使用基于用户交互的pairwise 的BPR变种损失优化模型。之后也有使用对抗学习更新该模型参数的方法，但由于用户物品的交互数据量级超出图片语料库太多，该方案的训练时长是个大问题。

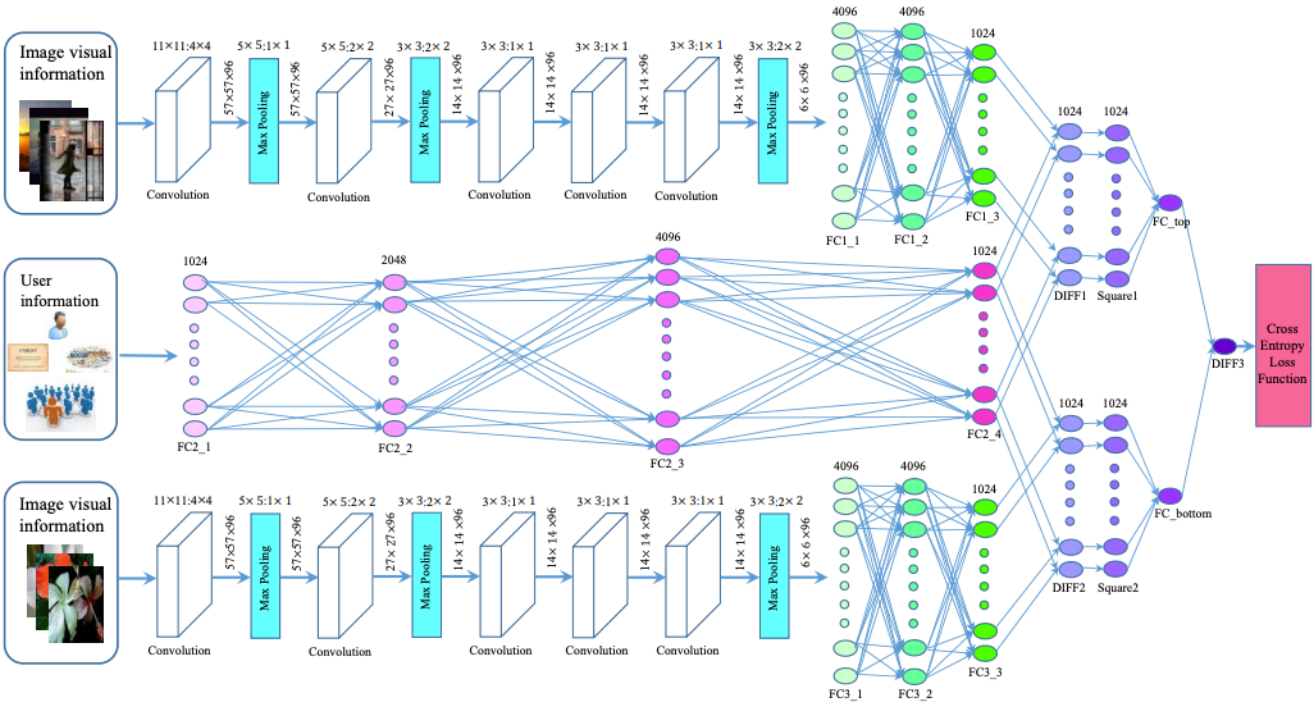


Figure 2. This figure depicts the deep network used for comparative deep learning (CDL). There are three sub-networks that all output 1024-dim vectors as representations of images and users, respectively. The top and bottom sub-networks processing images are identical. The middle sub-network is processing users. Following these sub-networks are two distance calculating nets. The difference between distances is fed into the final cross-entropy loss function for comparison with label. The numbers shown above each arrow give the size of the corresponding output. The numbers shown above each box indicate the size of kernel and size of stride for the corresponding layer.

Deep Image CTR model(DICM)

这篇文章并不是关于图像建模的，与之前将图片特征用于物料侧不同，阿里的这篇文章是将图片用于用户特征建模，基于用户历史点击过的图片建模用户的视觉偏好，即deep Image CTR model（DICM）。

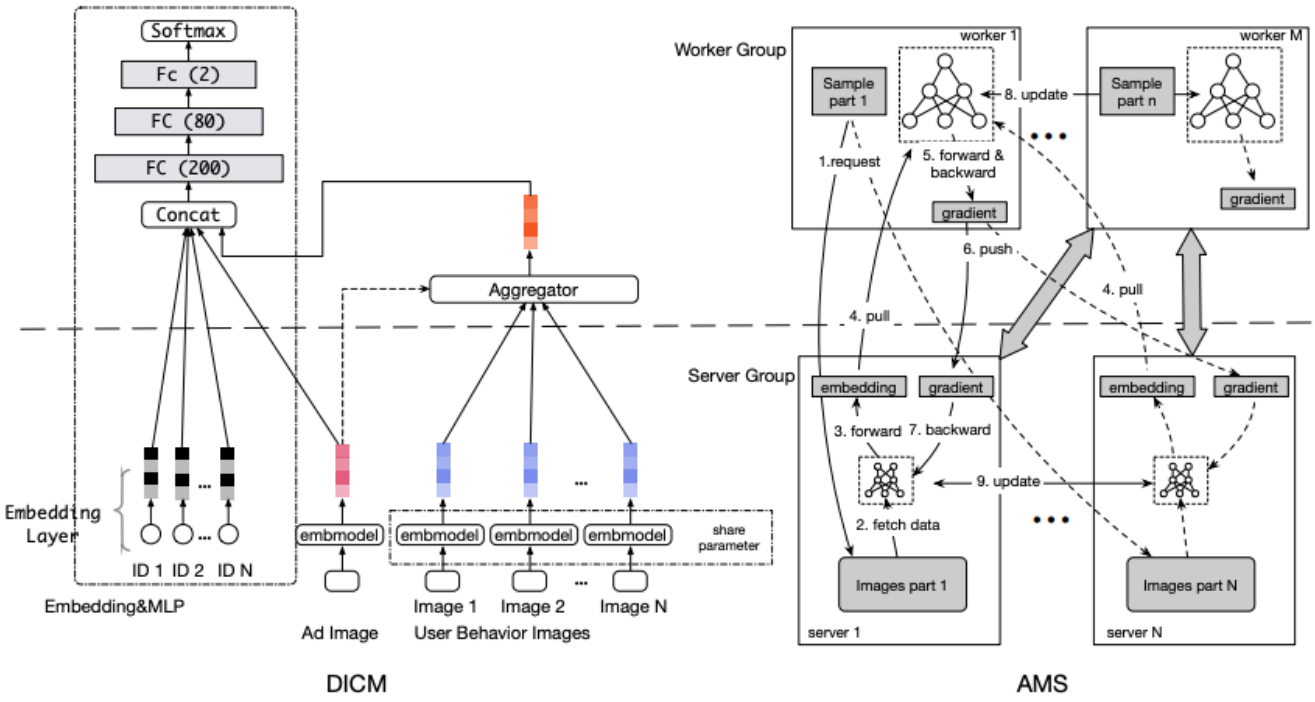


Figure 2: DICM network architecture implemented by Advance Model Server

DICM模型结构如上图左侧所示：

- 左边是embedding+MLP模式
- embedding层的粉色块是待预测的商品图片embedding，蓝色块是用户互动过的历史图片embedding，历史图片embedding和商品图片embedding计算attention分值后pooling成一个向量表示用户的视觉偏好
- 将id特征的embedding结果、商品图片embedding与用户兴趣embedding拼接起来，传入MLP，进行充分的交互，而且使用用户互动的历史图片表示用户兴趣可以解决商品的冷启动问题

但这篇文章还有一个更大的创新点：将图片的embedding和id类embedding一样保存到PS的server节点上，以图片index为索引，但一般的图像模型如VGG16提取出来的embedding长度有4096维，在PS模式下训练时会导致网络传输量暴涨，因此在server上增加一个可学习的压缩模型，即一个4096-256-64-12的金字塔型的MLP。

当worker想server请求图片embedding时，server上的压缩模型将原始的4096维压缩成12维后返回。该压缩模型的参数由每个server根据存在本地的图片数据学习得到，并且在一轮迭代结束时，各server上的压缩模型需要同步。

在交互图上用图卷积操作传播多媒体特征信息。代表模型有GraphSage&PinSage、MMGCN。

GraphSage & PinSage

2018年Pinterest公司和斯坦福合作的用于图片推荐的模型，基于2017年斯坦福的GraphSage落地而来，在用户物品交互图上使用图卷积网络提取图片表示。Deep CNN提取的图片表示作为物品节点的初始特征，在交互图上通过卷积操作传播。由于用户物品交互图包含用户对物品的偏好尤其是协同过滤信息，这个方案能使视觉特征更适用于个性化推荐。

关于该模型的详细解读参考下一节：图数据的表示学习。

MMGCN(Multimodal Graph Convolution Network)

19年的用于微视频推荐的多模态图卷积网络，和PinSage的主体思路相似。只是构建的是用户和小视频的二部图，用于给用户推荐小视频，卷积操作传播的特征有视觉、文本和声学特征，最后融合所有模态的输出得到小视频的最终表示。Deep CNN用来提取小视频特征。

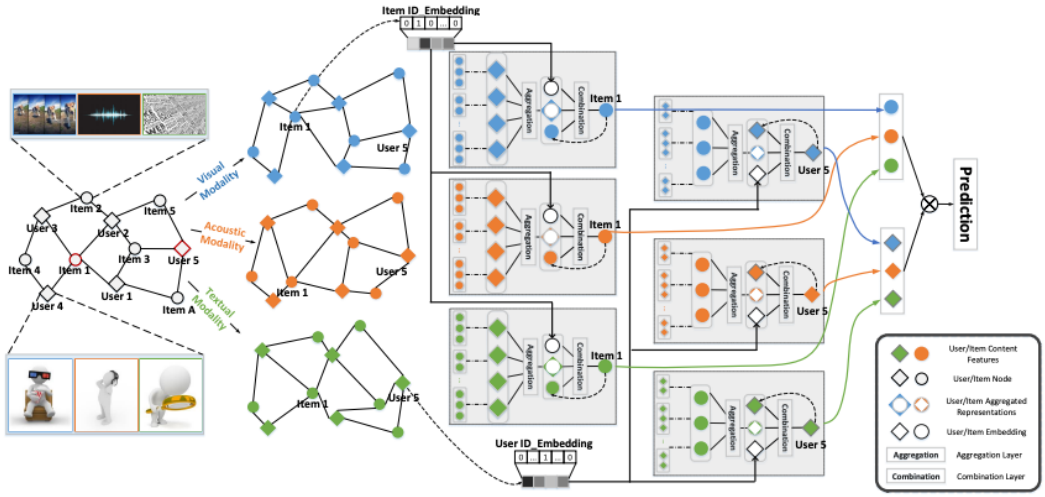


Figure 2: Schematic illustration of our proposed MMGCN model. It constructs the user-microvideo bipartite graph for each modality to capture the modal-specific user preference for the personalized recommendation of micro-video.

ACF(Attentive Collaborative Filtering)

不同于上面的模型都是用Deep CNN提取全图表示，ACF假设不同用户对图片中的不同区域感兴趣，因此将图片切分成49（7*7）个小区域，使用Deep CNN提取每个区域的特征，然后用一个注意力网络学习每个区域的权重，而且除了component级的注意力，还学习了item级的注意力。最后经过池化层后得到图片的表示。由于注意力网络基于用户物品交互信息训练而来，最终的图片表示可直接用于推荐任务。

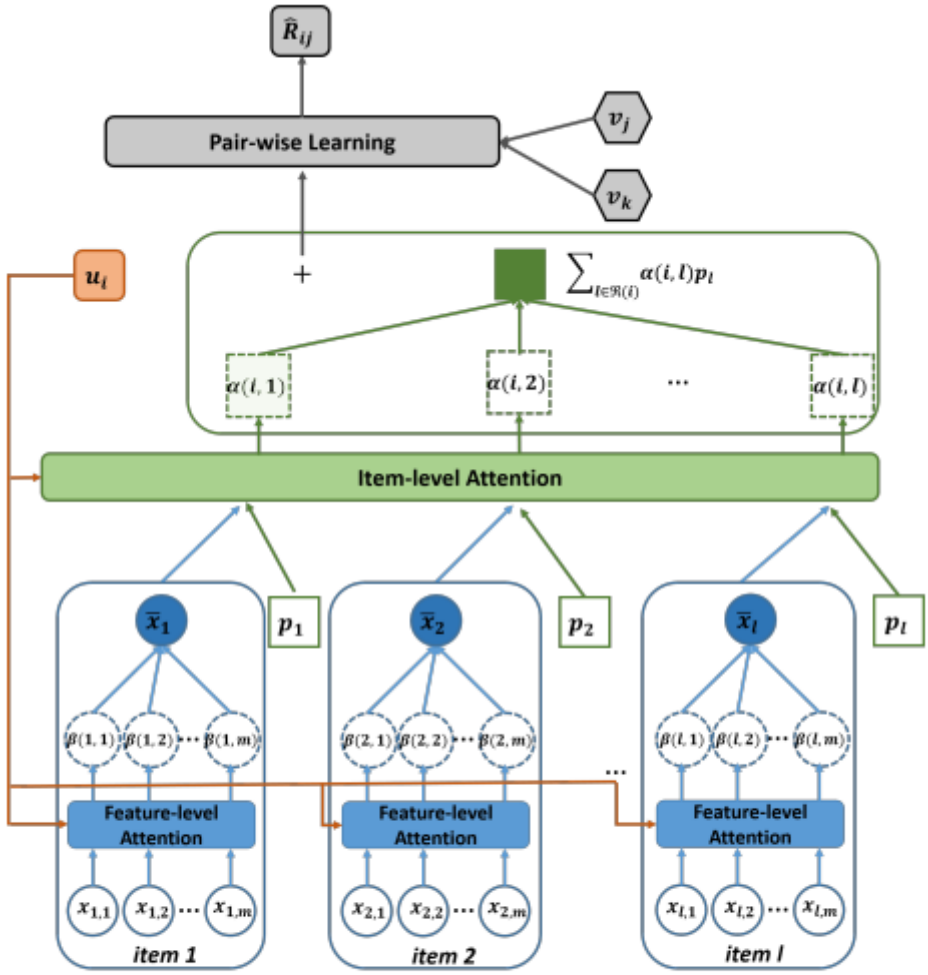
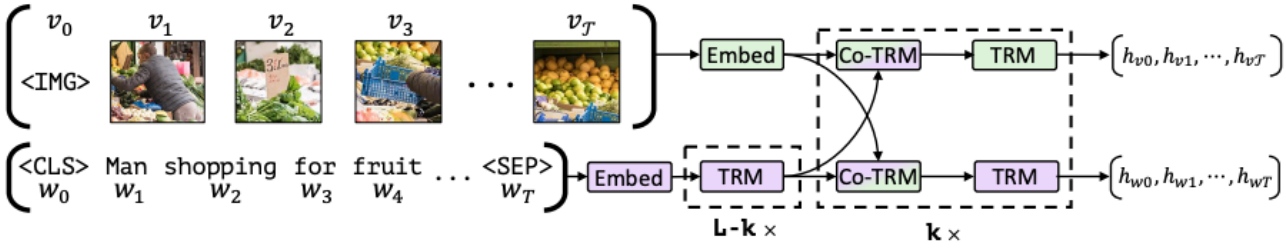


Figure 1: The architecture of our proposed Attentive Collaborative Filtering framework. Our attention model contains two level modules: component-level attention and item-level attention (cf. Section 4.1).

ACF的框架也被用于视频推荐，只是region划分改成了视频中的帧采样。

ViLBERT

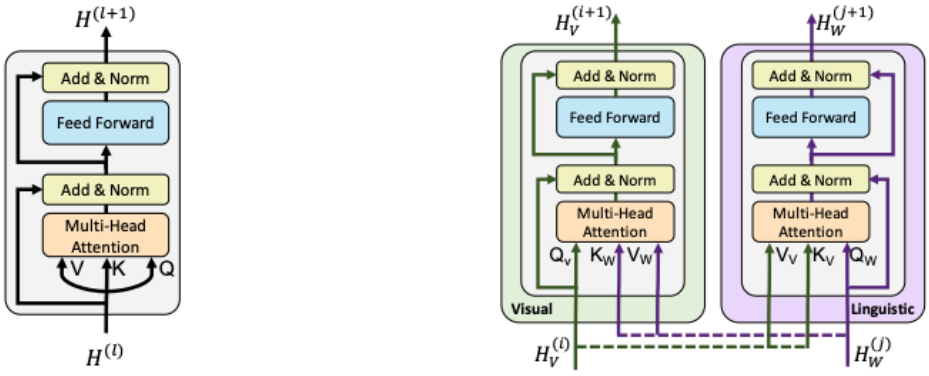
来自论文 Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks



如上图所示，ViLBERT是在bert的基础上提出一个双流架构，分别对每种模态建模，然后通过一组基于注意力的交互将它们融合在一起，即多模态双流模型，两个流分别处理图

像和文本输入，再通过co-attention Transformer层交互，最终学习到图像和文本的无任务偏好的联合表示，预训练后供下游任务使用。

其中co-attention Transformer层如下图所示，给定中间态的图片和文本表示，该层将每个模态中的keys和values输入到其他模态的多头注意力block，为每一种模态的信息产生依赖其他模态的注意力池化特征（attentioned-pooled）。这也是VilBERT和BERT的区别之处。



(a) Standard encoder transformer block (b) Our co-attention transformer layer

Figure 2: We introduce a novel co-attention mechanism based on the transformer architecture. By exchanging key-value pairs in multi-headed attention, this structure enables vision-attended language features to be incorporated into visual representations (and vice versa).

VisualBERT

来自论文：A Simple and Performant Baseline for Vision and Language VisualBERT则是包含一组堆叠的Transformer层，借助self-attention把输入文本中的元素和相关的输入图像中的区域隐式地对齐融合。

1.4 图数据上的表示学习

上面这些表示学习方法都有一个共同的问题：用户和物品的表示是分开学习的，用户和物品之间的关联被忽视了。用户物品交互图谱为用户和物品的关联提供了丰富的信息；物品的知识图谱为物品之间的关联提供了丰富的信息。

因此在图谱上学习用户和物品的表示既能克服上述缺点又能提升推荐的准确率。用户物品交互图可以被构建成二部图、展示用户社交关联的社交网络、体现物品知识的知识图谱。图结构连接用户和物品，有助于探索用户物品间高阶关联、捕获其中有用模式(如协同过滤、社交影响、知识推理)、提升表示学习。

我们将现有工作归为两类：

1. 两阶段学习，先将关系提取三元组或路径，再用关系学习节点表示；

2. 端到端学习， 直接学习节点表示并在节点间进行信息传播。

基于用户物品图

NGCF(Neural Graph Collaborative Filtering)

用户物品交互可以构成一个二部图，NGCF重新定义了协同过滤中的CF信号为图中的高阶连接。直接连接能显式刻画用户和物品的特征：用户交互过的物品显示了用户的偏好，物品的关联用户可以看做是物品的特征。

高阶连接则反映了更复杂的模式，正如下图所示，$u_1 \leftarrow i_1 \leftarrow u_2$ 表示用户u1和u2由于都和物品i1交互而产生了行为相似性；而更长的路径 $u_1 \leftarrow i_1 \leftarrow u_2 \leftarrow i_2$ 则表示用户u1对i2的兴趣，因为他的相似用户u2也交互了i2.

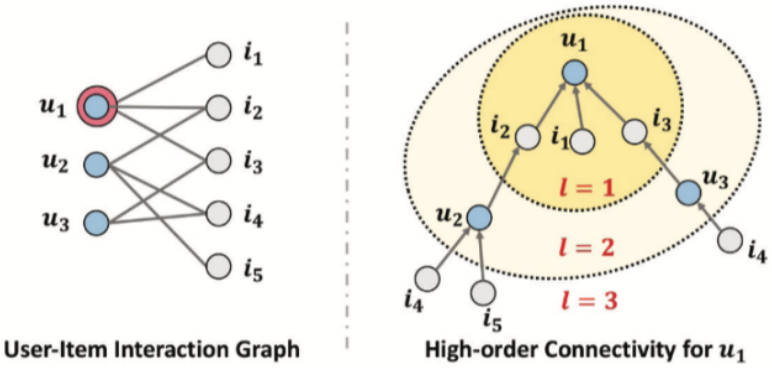


Figure 5.12: An example of high-order connectivity revealed in user-item interaction graph. The figure is taken from Wang *et al.* (2019b).

NGCF的模型结构如下图所示，采用了基于图上信息传播的gnn网络结构，在用户-物品的二部图上做Embedding传播。通常GNN的图卷积层包含两个部分：

1. 信息构建，定义从邻居节点传递到当前节点上的信息含义；
2. 信息聚合，聚合从周围邻居节点上传递来的信息并更新当前节点的表示。

常用实现：

$$p_u^{(l)} = \rho(m_{u\leftarrow u}^{(l)} + \sum_{j\in N_u} m_{u\leftarrow j}^{(l)})\$ ， \quad \$m_{u\leftarrow j}^{(l)} = \alpha_{uj} W^{(l)} q_j^{(l-1)}$$

其中 $p_u^{(l)}$ 表示经过l层传播后用户u的表示，$\rho(.)$ 是非线性激活函数，N_u 是用户u的邻居集合，$m_{u\leftarrow j}^{(l)}$ 是待传递的信息，α_{uj} 是在边(u，j)上传播的衰减因子，通常启发式地设置为 $1/\sqrt{|N_u||N_j|}$，$W^{(l)}$ 表示第l层的转换矩阵。这样一来，L阶连接信息也被编码到更新后的表示中。最后，NGCF将具有各种用户兴趣贡献的层传递过来的表示拼接起来并执行预测：

$$f(u,i) = p^{*T}_u q^*_i, p^*_u = p^{(0)}||\dots||p^{(L)}\$, q^*_i = q^{(0)||\dots||q^{(L)}}$$

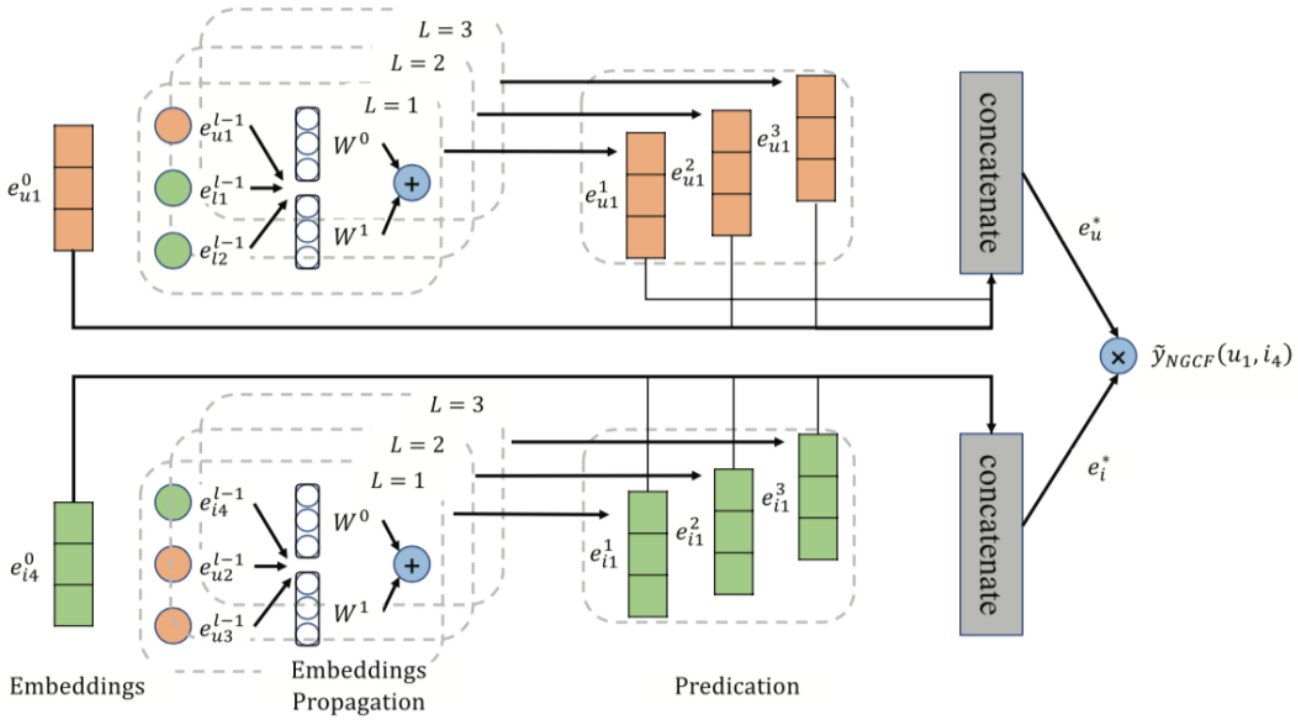


Figure 5.13: Model architecture of NGCF.

巧妙的是，NGCF去掉传播层可以泛化为MF模型；只保留一阶传播层则是SVD++模型。而且图的卷积层有多种实现方式，比如下面的PinSage和GCMC。

Pinsage&GraphSage

参考前文

GCMC(Graph Convolutional Matrix Completion)

将矩阵补全问题看做是关于user-item二分图的链路预测问题，每种链路可以看做是一种label（点击，收藏，喜欢，下载，评分等），GCMC通过在二部图上进行差异化信息传递学习节点的Embedding，再通过一个双线性decoder进行链路预测，输出属于不同评分值的概率，最后使用多分类交叉熵损失。

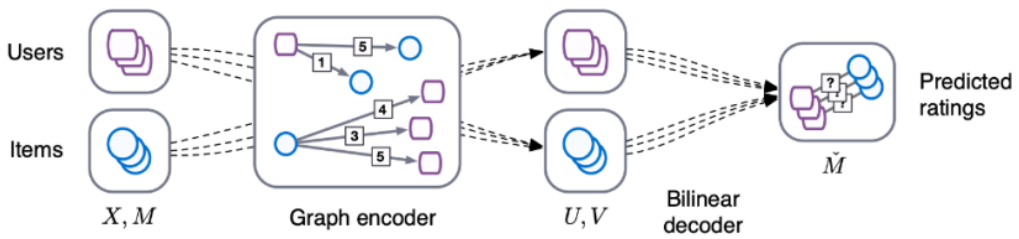


Figure 2: Schematic of a forward-pass through the GC-MC model, which is comprised of a graph convolutional encoder $[U, V] = f(X, M_1, \dots, M_R)$ that passes and transforms messages from user to item nodes, and vice versa, followed by a bilinear decoder model that predicts entries of the (reconstructed) rating matrix $\check{M} = g(U, V)$, based on pairs of user and item embeddings.

针对不同分值的子图分别进行局部图卷积再进行聚合的encoder模型做信息传递，联合MLP和 $p_u^{(l)} = \rho(m_{u\leftarrow u}^{(l)} + \sum_{j\in N_u} m_{u\leftarrow j}^{(l)})$， $m_{u\leftarrow j}^{(l)} = \alpha_{uj} W^{(l)} q_j^{(l-1)}$ 捕获非线性和复杂模式。

lightGCN

尽管NGCF中声明了交互图结构对表示学习的帮助，何向南等提出NGCF中很多设计都是冗余的，尤其是非线性特征转换。主要的争议点在于，用户-物品交互图的节点只有一个除了标识功能没有语义含义的one-hot id特征，这种情况下执行多层非线性转换和神经网络的标准操作没区别。

为了证明这个论点，他们提出了lightGCN模型，在图卷积操作中只保留邻居聚合：

$$p_u^{(l)} = \sum_{i\in N_u} \frac{1}{\sqrt{|N_u|}\sqrt{|N_i|}} q_i^{(l-1)}\$ ， \ \$q_i^{(l)} = \sum_{u\in N_i} \frac{1}{\sqrt{|N_i|}\sqrt{|N_u|}} p_u^{(l-1)}$$

其中 $p_u^{(0)}, q_i^{(0)}$ 是id的Embedding，原来NGCF中的非线性特征转换和自循环都被移除了。经过聚合L层高阶邻居后，LightGCN将所有层的表示加起来作为用户/物品的最终表示：

$$p_u^* = \sum_{l=0}^{L} \alpha_l p_u^{(l)}; q_i^* = \sum_{l=0}^{L} \alpha_l q_i^l$$

其中 α_l 是预先定义好的第l层表示的重要性。同时作者也证明了sum聚合器将自循环归进了图卷积操作中，因此可以移除显式的自循环卷积。相同的数据集和评估方法下，lightGCN比NGCF有15%的提升。

基于知识图谱

KGAT (Knowledge Graph Attention Network)

除了用户物品交互图，最近的很多工作也开始考虑知识图谱中物品间的关联。知识图谱是一种能提供丰富的物品side information(如物品属性和物品关联)的信息来源，节点都是实体，边表示实体之间的关联。通常知识图谱将一些事实整合到一个异质有向图 $G = (h, r, t)|h, t \in \varepsilon, r \in R$，$(h, r, t)$ 表示实体r到实体h之间有连接。知识图谱能增强物品表示的学习，也能建模用户物品关联，直接连接能刻画特征，多阶连接则是更复杂的关联。

比如KGAT模型就是是通过迭代式地从节点的高阶邻居中提取信息来扩展NGCF模型。不同于NGCF中边 (h, t) 上信息传递的衰减因子 α_{ht} 是固定的，KGAT使用关系注意力机制学习边 (h, r, t) 上的关系r。带注意力的embedding传播层定义：

$$p_h^{(l)} = f_1(p_h^{(l-1)}, m_{(h,r,t)}^{(l)} | (h,r,t) \in N_h)$$
$$m_{(h,r,t)}^{(l)} = f_2(q_t^{(l-1)}, \alpha_{(h,r,t)})$$
$$\alpha_{(h,r,t)} = \frac{expg(p_h, e_r, q_t)}{\sum_{(h,r',t')} expg(p_h, e_{r'}, q_{t'})}$$

，其中 $f_1(.)$ 是信息传递函数，用于更新头结点h的表示，$f_2(.)$ 是注意力信息构建函数，产生从头结点h到尾结点t的信息，$\alpha_{(h,r,t)}$ 是注意力网络 $g(.)$ 得到的注意力衰减因子，表明需要传递多少信息以及这些邻居节点对于关系r的重要性。学习到节点表示后，KGAT使用与NGCF一样的预测模型，即将不同层的表示拼接起来，计算：

$$f(u,i) = (p_u^*)^T q_i^* \$ ， \quad \$p_u^* = p^{(0)} || \ldots || p^{(L)} \$ ， \quad \$q_i^* = q^{(0)} || \ldots || q^{(L)}$$

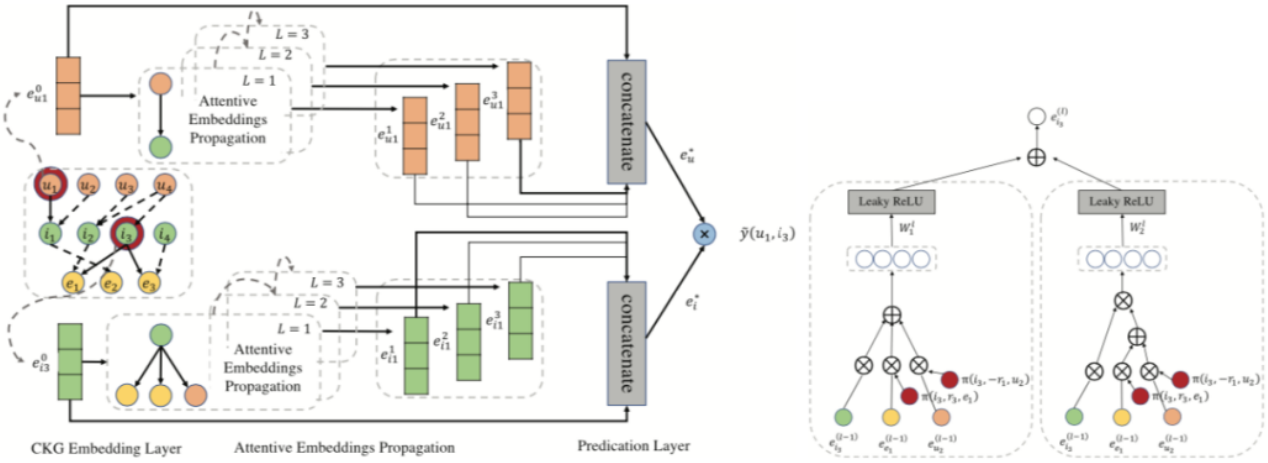


Figure 5.14: Model architecture of KGAT. The left subfigure illustrates the overall model framework, and the right subfigure illustrates the graph convolution operation in KGAT.

尽管端到端建模可以加强高阶连接的表示学习，也有很多工作是基于元路径或路径来直接提炼用户和物品间的相似性。模型会首先定义元路径模式或者提取可信路径喂到有监督学习模型中预测分值。

KPRN （Knowledge Path Recurrent Network）

给定实体间的路径，KPRN用循环网络如LSTM对路径上的元素编码，捕获实体间的语义关系；之后使用一个池化层将多个路径表示转换成单个向量送入MLP获得用户物品对最终的得分：

$$x_k = LSTM([p_{h_1} || e_{r_1}, \ldots, p_{h_L} || e_{r_L}])$$

其中 $p_k = [h_1, r_1, \ldots, h_L, r_L]$ 是第k条路径，(h_l, r_l, h_{l+1}) 则是 p_k 中的第l个三元组。因此KPRN能通过LSTM利用知识图谱上的序列信息并且增强推荐的可解释性。

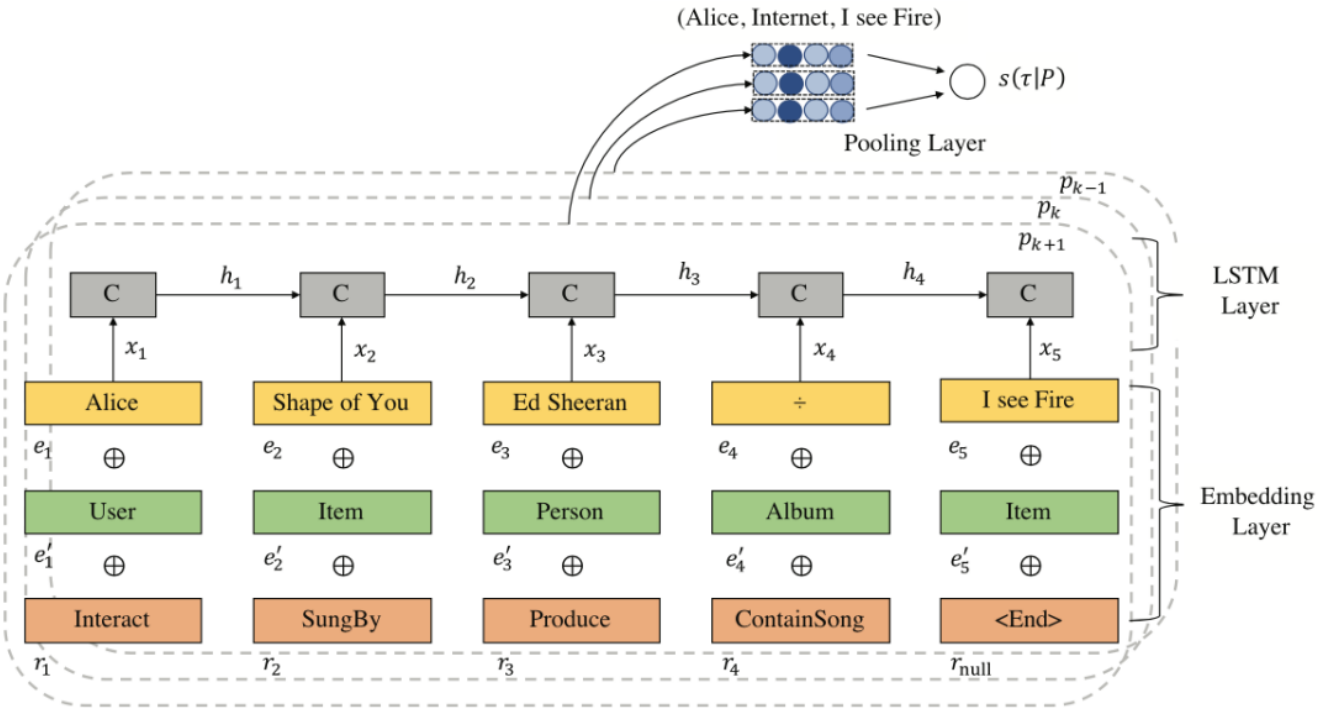


Figure 5.15: Model architecture of KPRN.

基于社交网络

DiffNet

社交推荐模型利用每个用户的局部邻居偏好来缓解数据稀疏性，从而更好地进行用户Embedding建模。DiffNet是一种基于SVD++以及GCN的社交推荐模型。即使用GCN建模用户的社交网络获取用户的Embedding，然后使用SVD++的框架。

对于每个用户，diffusion过程融合了用户相关特征和表示用户行为偏好的隐向量，将用户Embedding随着社交网络的扩散而演化。

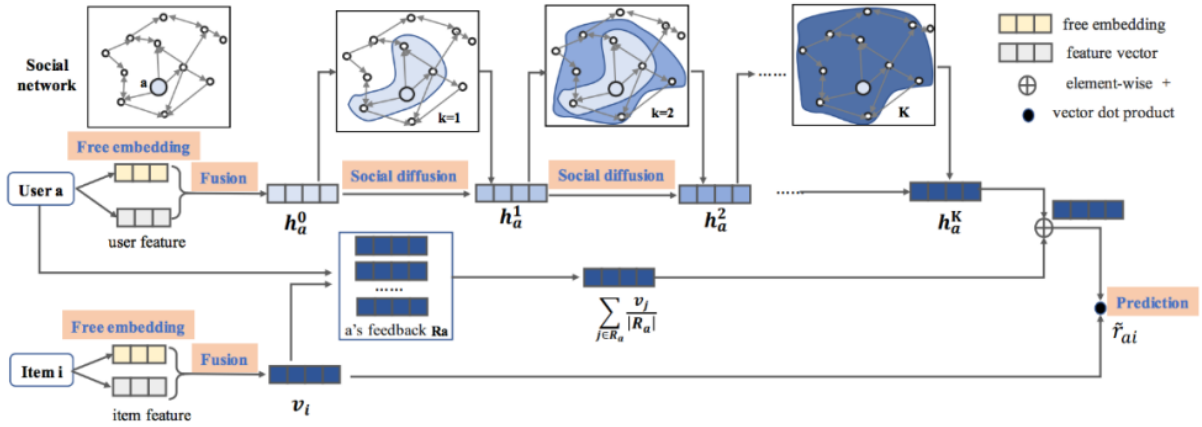


Figure 1: The overall architecture of our proposed model. The four parts of DiffNet are shown with orange background.

GraphRec

整合user-user图和user-item图，使用attention网络学习社交关系的重要程度。

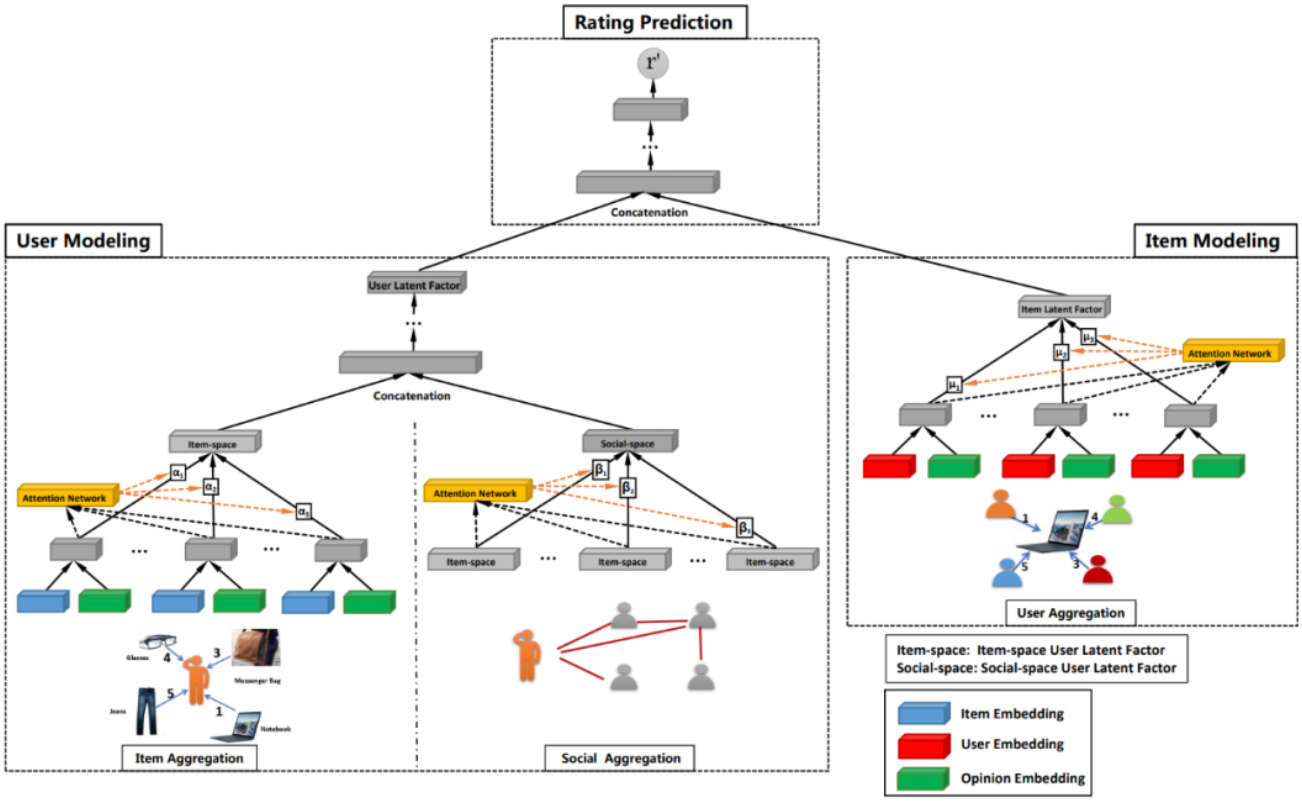


Figure 2: The overall architecture of the proposed model. It contains three major components: user modeling, item modeling, and rating prediction.

整个结构分为三部分

- 用户建模（user Modeling）：分别聚合用户关联的物品和用户生成Embedding后拼接形成用户的Embedding
- 物品建模（item Modeling）：利用用户对物品的连接关系生成item的Embedding
- 打分预测（rating prediction）：将上述用户和物品Embedding拼接后通过一层MLP预估某个用户对某个item的打分。

DiffNet++

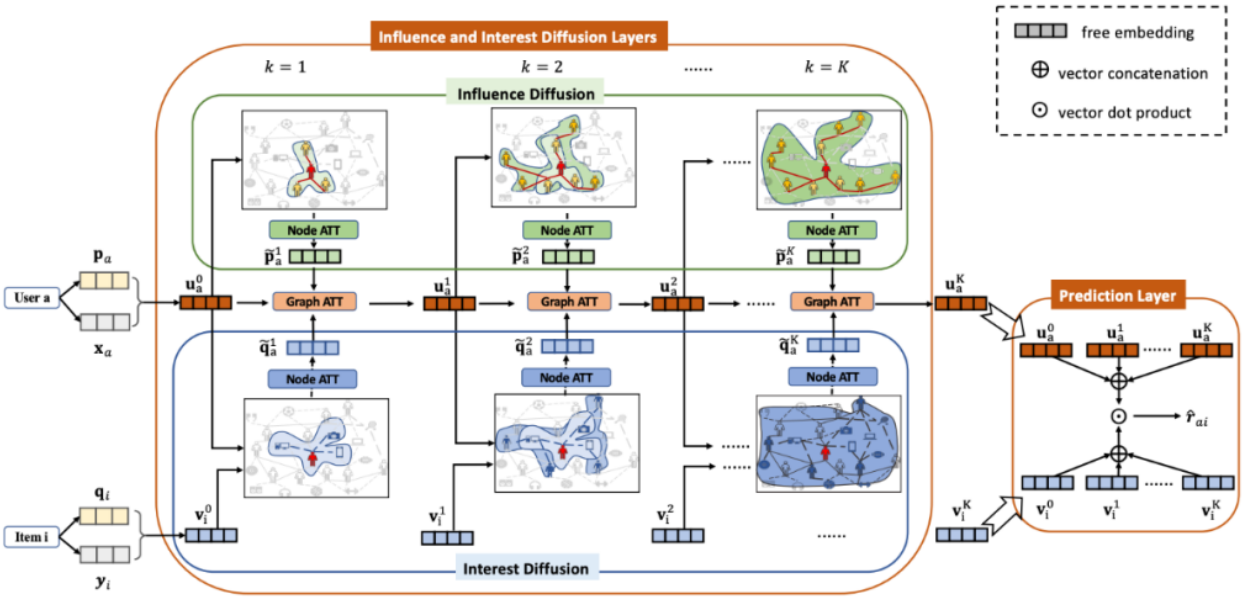


Fig. 2. The overall structure of the DiffNet++ model. As shown in the graph, we use *Node ATT* to denote the node level attention layer in each graph, and *Graph ATT* to denote the graph attention layer when fusing the interest graph representation and social graph representation.

基于多层attention融合用户-用户关注图和用户-物品的交互图的各阶邻居信息，经过全图多次扩散后，预测阶段将用户和物品的全部K层Embedding拼接后计算内积得到偏好分值。

本文参考资料

[1]《深度推荐系统总结系列一》：https://blog.csdn.net/qq_39388410/article/details/106432299