多模态表征学习

多模态表征学习，主要包括图像表征、文本表征和属性表征三个方面的学习，是为了提取不同模态的特征信息。提取的过程如图所示。

属性表征：对于item的各个属性，如id、类别、颜色等信息进行编码。对于这些信息，使用One-hot 方法编码的向量会很高维也很稀疏，因此我们使用Embedding Layer的方法，首先获得属性信息的one-hot矩阵作为输入，再通过一个全连接神经网络层，将不同信息映射到致密的低维向量中。Embedding Layer的输出维度dx将作为可选择的参数。

文本表征：使用Word2Vec来训练词向量，然后使用SIF模型，将预训练好的词向量，使用加权平均的方法，对句子中所有词对应的词向量进行计算，得到整个句子的embedding向量，最后使用主成分分析去掉一些special direction，即在完成词加权平均后，移出所有行为向量的公共主成分，只保留反应序列特性部分，来表示文本特征。

图像表征：使用ResNet50的最后一个隐藏层的2018维特征（由Inmagenet训练得到）进行训练获得。

2. 多模态序列图推荐

由于不同模态之间存在语义差异，且不同用户可能对不同模态偏好程度不同，因此为了提高推荐的准确性，为每个模态单独构建一个序列图。

2.1原始图构建

对于某个模态，所有用户的所有序列被建模为一个有向序列图，其中每个用户的每个序列被视为一个子图。对于某个序列子图表示为：Gm=（V, E）。图中每个节点v∈V代表用户交互的某个节点i，其表示为上游通过多模态表征学习获得。每条边（vi-1, vi)∈E意味着用户与物品i-1交互后又与物品i进行交互。由于在一个序列中，用户可能多次与同一个物品进行交互，因此，对每条边分配一个归一化的权重，该权重的计算方式是边的出现次数除以这条边源节点的出度。由于上游学习到的不同节点的维度可能是不一样的，我们需要将所有节点统一到同一个维度空间，每个节点最后统一维度的表示通过GNN网络获得，对应的序列因此可以被表示为由节点张量组成的嵌入张量sm。

2.2序列图中节点嵌入学习

我们通过图神经网络获得节点的隐藏张量表示。对于子图Gm中的节点vi来说，更新函数表示如下：

其中 控制权重，分别是重置和更新门，是序列s中的节点列表，是激活函数，是元素乘法操作。代表节点vi的潜在表示。连接矩阵

决定了图中节点是如何互相联系的，是As中和节点vi联系的两列。

As定义为两个邻接矩阵，分别代表序列中出边和入边的权值。例如，考虑一个序列,其序列图和矩阵As如下图所示。

对于序列图Gm，门图神经网络同时处理所有节点。在矩阵As给出的前提下，公式（1）用来实现不同节点之间的信息传播。它提取邻居的潜在表示，并将信息喂到图神经网络的输入。然后利用更新和重置门，决定哪些被传播的信息被保留，哪些被忽略。然后，按照公式4，根据节点历史表示，重置门以及当前状态构建候选的节点表示。最后的状态是在更新门的控制下，由历史隐藏状态和候选状态结合而成。直到更新完序列图中的所有节点收敛，我们才得到最后的节点潜在表示。

2.3 生成序列嵌入

序列的表示是由序列中的节点直接表示而来的。为了提高推荐的准确性，我们将长期的偏好与序列中最近时间的兴趣结合，使用混合的嵌入作为序列的嵌入。

对于序列s=[v1,...,vn]，其局部嵌入可以简单的为最后一次交互的节点的嵌入，即sl=vn。

其全局嵌入，可以通过序列图中所有节点的表示聚合而成。使用soft-attention机制获取全局嵌入sg的表示

q、W1、W2都是控制item嵌入的权重参数。

最后，将局部嵌入sl与全局嵌入sg的拼接通过线性转换获得最中的序列表示

2.4 模态融合

分别在三个模态序列子图获得序列的最终表示sh1，sh2和sh3之后，我们探索使用两种不同的方法进行模态融合对最后推荐的性能的影响：

串联拼接：使用非线性变换，将三个表征拼接起来：

ii. 元素组合：

2.5 预测推荐与模型训练

· 得到每个模态的嵌入表示之后，将不同的模态的sh嵌入表示进行拼接，得到最后的用户序列嵌入表示。对每一个候选的item，其得分函数计算方式：

再通过一个softmax函数，得到所有候选item被推荐作为下一个交互的item的概率。