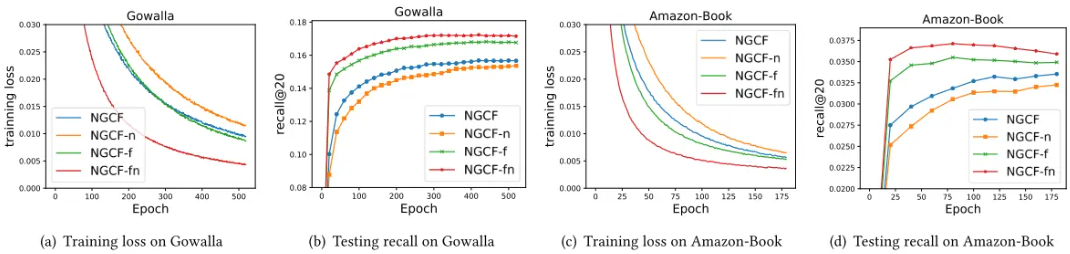
1. **LightGCN**
2. **基本认知**

在推荐系统领域，图卷积网络（GCN）已然成为结合协同过滤算法的关键技术之一。然而，提出LightGCN模型的原文作者认为：**GCN中关于特征转换和非线性激活的设计，没有显著提升协同过滤性能的同时，反而徒增训练难度[[1]](#footnote-0)**。由此，作者在提出的LightGCN模型中只保留领域聚合和多层传播等核心部分，通过在用户-项目交互图上线性传播的方式学习用户和项目不同层次上的嵌入表征，然后将所有层上学习到的用户和项目嵌入加权求和，从而完成用户和项目的最终表示。LightGCN的轻量特性体现为：模型的可训练参数，仅有用户和项目的初始嵌入表示及超参。

重要的是，在只由用户和项目ID来生成初始嵌入表示的情形下，原文作者**针对NGCF展开消融研究**，探索了特征转换和非线性激活设计对于协同过滤过程的有效性。具体而言，原文作者比较了NGCF-f（删除了和的特征转换）、NGCF-n（删除非线性激活函数）、NGCF-fn（同时删除了特征转换矩阵和非线性激活函数）和NGCF，在度量指标Recall和NDCG下的推荐性能。实验结果，如下图所示。其中，相较于基准NGCF，去除特征转换的NGCF-f和NGCF-fn，在推荐性能上得到了显著提升；单独去除非线性激活函数，将对推荐性能产生不良影响；同时去除特征转换和非线性激活设计，将获得最显著的推荐性能表现。至此，综合考量不同模型关于训练损失和推荐性能的变化趋势，可得出**结论**：NGCF模型的恶化不是过度拟合问题，而是由于特征转换和非线性激活设计造成的负面影响。

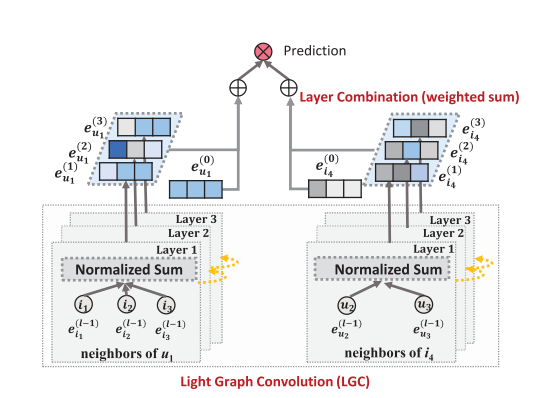


* 1. **网络架构**

（1）邻域聚合

LightGCN的网络架构，如下图所示。不同于NGCF，LightGCN在每一个图卷积层中，通过使用放弃自连接、特征转换和非线性激活等设计的加权求和聚合器，完成了用户和项目节点表示在不同抽象层次上的邻域聚合工作。具体公式，如下所示。其中，表示用户评过分的项目列表，表示用户评过分项目的数量，表示项目的第层嵌入表示。





（2）组合嵌入

此后，LightGCN还需要将用户和项目的原始输入，和不同图卷积层生成的嵌入表示做一个加权和，才能获得用户和项目的最终嵌入表示。具体公式，如下所示。其中，超参表示第层用户或项目嵌入的参考程度，推荐取值。



值得注意的是，通过加权和各层嵌入结果的方式来获得用户和项目的最终表征，是**因为**随着图卷积层数的增加，邻域聚合可能会导致嵌入表征取值愈加平滑，使用最后一个图卷积层生成的嵌入结果是有问题的。同时，通过加权和的方式结合不同图卷积层的嵌入结果，还可以模拟自连接的图卷积效果。此外，不同图卷积层的嵌入结果本身捕获了不同层次的语义，对于它们的组合能够更好地完成表征学习，比如：第一层生成的某用户表征，是根据其全部交互项目的平滑结果；第二层生成的用户（项目）表征，是根据具有交互项目（用户）重叠的全部用户（项目）的平滑结果。

值得补充的是，可凭借如下公式**理解二阶嵌入平滑**，即：第二层生成的用户表征，如何使用具有交互项目重叠的全部用户的表示，通过归一化求和的方式获得平滑结果。



分析上述公式，讨论二阶近邻对用户嵌入表示的影响程度，可得出如下结论：近邻用户和目标用户间的共同交互项目越多，则近邻的影响越大；共同交互项目的流行度越低，则存在交互关联的所有近邻用户的影响越大；近邻用户的活跃度越低，则其对于用户的表示影响越大。显而易见的是，LightGCN至少在第二个图卷积层上的聚合操作，与协同过滤算法关于用户相似性的假设不谋而合。

（3）预测输出

经验是，在只由用户和项目ID来生成初始嵌入表示的情形下，考虑使用内积完成交互模型的构建更为简洁高效。由此，若需使用LightGCN预测用户和项目间的相关性分数，直接将用户和项目的最终嵌入向量内积即可，公式为：。

* 1. **矩阵逻辑[[2]](#footnote-1)**

设用户-项目交互矩阵为（表示用户数，表示项目数），如若用户和项目间存在交互，则矩阵对应位置取值为1，否则为0。由此，为了更自然地表示用户和项目间的二分关系，可考虑使用构造的邻接矩阵A表示二分图。此外，还可额外引入度矩阵D对构造的邻接矩阵A进行归一化处理，帮助信息在图中更平滑地流动，从而缓解矩阵乘法中的数值爆炸或消失问题。具体表示，如下所示。其中，D是一个对角矩阵，对角元素的取值为节点的度，即：邻接矩阵A第行向量中的非零项数。



至此，轻量图卷积（Light Graph Convolution, LGC）的等价矩阵形式，如下所示。



更进一步，用于表示所有用户和项目的最终嵌入矩阵，可进行如下分解。其中，表示用户和项目的嵌入表示维数。值得注意的是，可以通过设置的取值，实现在可控的过平滑情形下使用较大的K进行远距建模。



* 1. **训练过程**

如前所述，LightGCN的可训练参数仅有用户和项目的初始嵌入表示，模型的复杂度和矩阵分解（MF）相当。LightGCN采用贝叶斯个性化排名（BPR）作为损失函数，选用Adam优化器作为优化函数，并以mini-batch的方式完成模型的迭代训练。

**BPR损失函数**，公式如下所示。具体而言，在迭代学习节点嵌入的过程中，BPR通过最大化每个用户（共M个用户）的每个交互项目（用户有个历史交互项目）和其它所有未知项目（）间的预测得分差距，从而实现个性化推荐。其中，超参控制L2正则化的程度，sigmoid激活函数用于将评分差取值映射到（0, 1）之间，使得损失函数的梯度更新稳定高效。值得注意的是，在LightGCN中不需要引入dropout，针对嵌入的L2正则化处理就足以应对过拟合问题。



此外，另一个值得关注的参数是，它在组合嵌入时衡量第层用户或项目的嵌入，在最终嵌入表征中的参考程度。一种常见的想法是，可利用注意力机制学习参数。**更进一步**，也可将可学习的注意力权重矩阵同最终嵌入相乘，从而利用注意力机制考虑所有用户和项目的最终嵌入表征。此外，也可以将最终嵌入同时赋值给Q、K、V，并将计算而来的注意力分数和值矩阵相乘，代替LightGCN模型直接通过点积来计算相似性的过程。当然，也可以通过引入线性层，将Q、K、V的嵌入维数投影为特定的注意力维数，从而完成考虑注意力机制的缩放点积计算过程。具体实现过程，详见后续实验报告。

1. He X, Deng K, Wang X, et al. Lightgcn: Simplifying and powering graph convolution network for recommendation[C]//Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval. 2020: 639-648. [↑](#footnote-ref-0)
2. https://www.jianshu.com/p/dca660b0c827 [↑](#footnote-ref-1)