

推荐系统

一、简介

推荐系统是解决互联网信息爆炸所带来的信息过载的一种方案，对用户而言可以找到感兴趣的东西，帮助决策，发现新鲜事物；对商家而言则可以提供个性化服务，提高信任度和粘性，增加营收。推荐系统就是解决这两方面问题的重要工具，使得用户和商家达到“双赢”。

一般的推荐系统是由后台的日志系统，核心推荐算法以及前台的展示页面三部分组成，而学术上主要特指其中的推荐算法，本文的推荐系统概述将着重围绕其中的推荐算法部分。

常见的推荐算法可以分为基于内容的推荐、协同过滤推荐、基于关联规则的推荐、基于知识的推荐、组合（混合）推荐及其他推荐这几类，而前两种算法是最为主要的、用得最多的推荐算法。

推荐系统领域主要面临的问题包含了以下几个方面：冷启动问题，包括用户冷启动和物品冷启动，指的是刚进入一个系统的新用户或者新产品由于没有历史记录，系统无法给其提供精确的推荐，可以通过基于内容的方法解决。用户的长期兴趣和短期兴趣问题，指的是推荐时不仅要考虑用户的长期兴趣，而且需要考虑用户的短期兴趣，可以通过类似 LSTM 模型来解决。跨领域推荐问题，指的是利用一个系统中的行为记录来预测另一系统中用户预期行为，可以通过跨域的知识迁移来解决。实时性问题，指的是在具有极大用户和物品空间中，如何在可接受时间内给出推荐。此外，还面临着时效性问题、多样性和新颖性问题、用户隐私问题等一系列问题。

二、推荐引擎算法

近几年，随着数据量的不断增加，仅靠传统的推荐算法已经难以支撑推荐系统，识别精度低、推荐响应时间长等劣势逼迫推荐引擎算法的改进。因此，深度学习为推荐系统提供了另一个支撑点，利用神经网络强大的数据处理和学习能力，结合传统的推荐算法，使得推荐系统在海量数据集下，也能取得非常高的准确性。因此，本章根据所使用的神经网络的类型，对推荐引擎算法进行分类[1]。

2.1 基于 CNN 的推荐算法

CNN（Convolution Neural Network）在图像、文字、语音信号处理和特征提取等方面具有优越的性能，因此广泛应用在多媒体等的特征提取领域。

DeepCoNN [2]利用了两个并行的 CNN 用于处理用户信息和物品信息，模型如图 1 所示。Look-up 层对文本信息进行 word embeddings，对每一个词形成一个 c 维的词向量，同时保持词序（向量序列），最后对所有词向量进行串联形成一个矩阵。CNN 层包含了卷积层、max-pooling 层以及全连接层。卷积层利用一个大小为 t 的滑动窗口执行卷积操作，激活函数选择性能较好的 ReLUs。Max-pooling 层使得原本长度不一的卷积结果可以收缩到一个固定大小的向量（最大值向量）。由于 CNN 的卷积层使用了多个神经元并行计算，因此 max-pooling 获得多个向量，最后在全连接层将这些向量通过一个加权矩阵，得到 CNN 层的输出结果。由于用户网络输出结果和物品网络的输出结果可能处于不同的特征空间，因此文章将两个网络的结果进行串联，然后利用因子分解机模型（FM）建立模型最终的损失函数。模型的优化训练通过 RMSprop 方法。

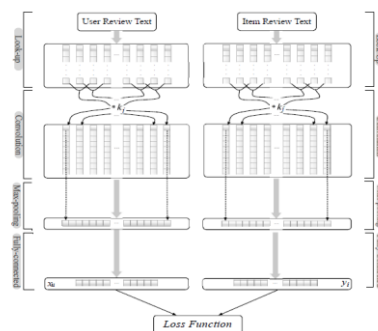


图 1 DeepCoNN

ConvMF [3]结合了 CNN 和 PMF 两种模型，实现内容感知的推荐，图模型如图 2 所示。ConvMF 在传统的 PMF 模型上，增加了物品 V 的影响因素，利用 CNN 模型来描述物品信息。ConvMF 中使用的 CNN 模型如图 3 所示，其架构与 DeepCoNN 中所用模型类似，包含 4 个层，即 embedding layer, convolution layer, pooling layer, output layer。embedding layer 对文档的每一个词生成一个 p 维向量，然后对文档中所有词向量进行串联，构成一个 $p \times 1$ 的矩阵。convolution layer 和 pooling layer 与 DeepCoNN 相同，最后的 output layer 使用一个两层的传统非线性投影层，将 pooling 之后固定大小的特征投影到一个 k 维空间。激活函数为 tanh 函数。模型的优化方法选择 maximum a posteriori (MAP)方法。

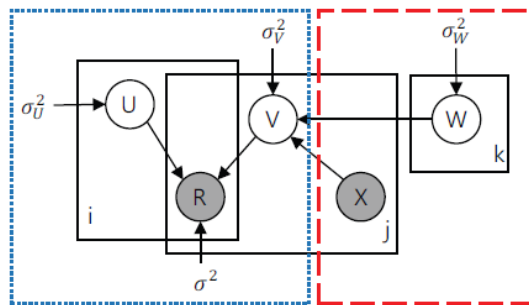


图 2 ConvMF 图模型

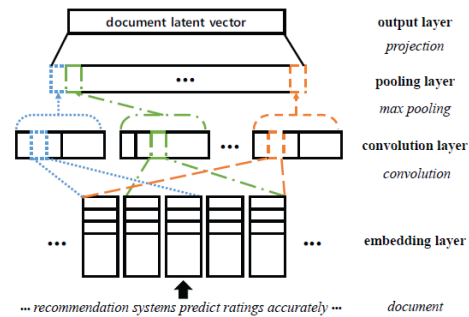


图 3 ConvMF 中的 CNN 模型

VPOI [4]利用 CNN 网络来提取图片特征，实现 POI 推荐，图 4 为 VPOI 的图模型。其中，用户的 POI 推荐的基础模型为 PMF 模型，用户和地点的特征可以通过 CNN 网络利用图片来提取，CNN 网络为 VGG-16 模型，如图 5 所示。模型的优化方法为梯度下降。

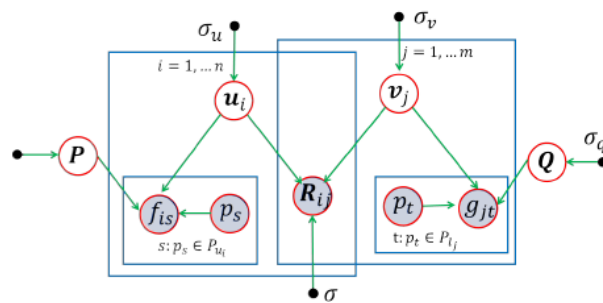


图 4 VPOI 图模型

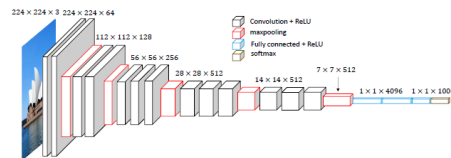


图 5 VGG-16 模型

Xiaoxuan Shen 等 [5]利用 CNN 网络来实现自动学习资源的推荐，设计了如图 6 所示的架构。在训练阶段，图 7 所示的 CNN 网络的输入为语言模型，输出为隐因子模型，语言模型论文选择了主题模型 LDA，将每一个词表示为属于该主题的概率。在推荐阶段，利用训练好的 CNN 网络，同样以语言模型的输出结果为神经网络输入数据，获取学习资源的特征，最后结合学生（用户）喜好特征进行推荐预测。图 7 的 CNN 网络包含了卷积层、mean-overtime region pooling 层、卷积层和全连接层。模型的训练上，利用 LFM 隐因子模型的输出结果和 CNN 的输出结果，最小化两者的误差 MSE。

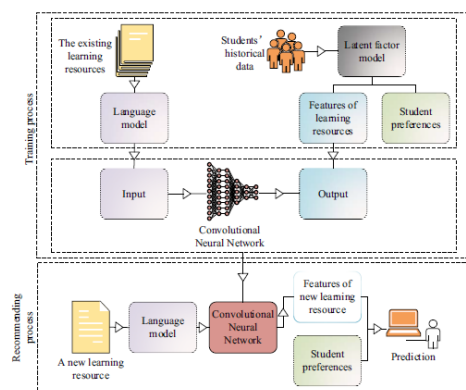


图 6 架构图

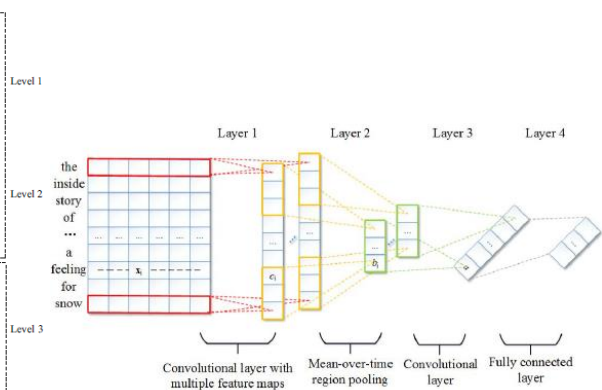


图 7 CNN 架构

2.2 基于 RNN 的推荐算法

RNN (Recurrent Neural Network) 适合对有时间性、时序性的特征进行提取，或者是对有上下文关联信息的提取。RNN 应用于推荐系统领域比较常见的网络模型主要为 LSTM 和 GRU 模型。

由于用户对电影的喜好程度是随时间的迁移而改变，RRN [6] 为了提取动态的时间性的特征，利用 LSTM 模型对用户和电影信息进行提取。LSTM 分别对不同时刻的用户和电影的特征进行提取，如图 8 所示。作者认为，用户对电影的打分除了随时间的变化的因素之外，还存在一些静态因素，因此定义了一个结合静态特征和动态特征的评分函数。在打分预测时，选择最近一次的观察作为神经网络的输入，获得更新后的用户状态和电影状态，从而计算出当前用户对电影的打分预测。

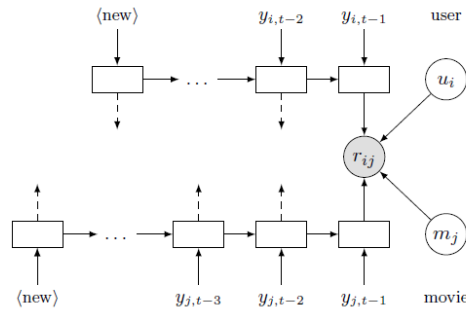


图 8 RRN 模型

Trapit Bansal 等 [7]则利用另一种 RNN 网络——GRU 来对文本进行编码，映射到一个隐因子空间，同时通过多任务学习，进一步提高了文本推荐的精度。如图 9 所示，GRU 网络将文本中的每一个词进行有序编码，利用 mean pooling 获得这段文本的表示，最后加上一个与这段文本相关的 embedding，获得最终的表达方式。最后的多任务学习，利用了两方面的信息，一方面是用户与文本之间利用隐因子模型的 cost，另一部分是标签与文本之间的加权二进制 log 似然 cost，最后通过一个调节因子，将两部分 cost 进行线性叠加，获得最终的损失函数。

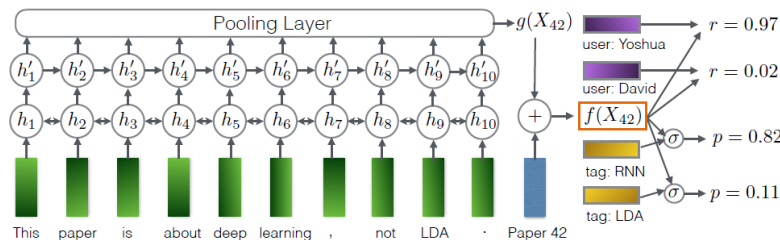


图 9 架构图

2.3 基于 MLP 的推荐算法

MLP (Multilayer Perceptron) 是一种简单而又有效的神经网络模型，也是许多其他类型的神经网络模型的基础。

NCF [8]利用一个 MLP 模型来学习一个非线性的用户-物品交互函数，其架构图如图 10 所示。输入层将用户和物品以 one-hot encoding 的方式进行编码，形成稀疏的向量，然后一个全连接的 embedding 层将稀疏向量投影成一个密集向量（隐式向量）。接着，一个多层的前向神经网络 neural collaborative filtering 层将

用户向量和物品向量进行 CF，得到映射后的用户对物品的预测评分。最后，定义了 pointwise 的平方误差损失函数对模型进行训练。图 10 右图的 NeuMF 模型是在左图的基础上，利用了一个 GMF 和 MLP 两个模型对用户和物品的隐式向量分别进行线性和非线性的表示，最后对两种结果进行混合。

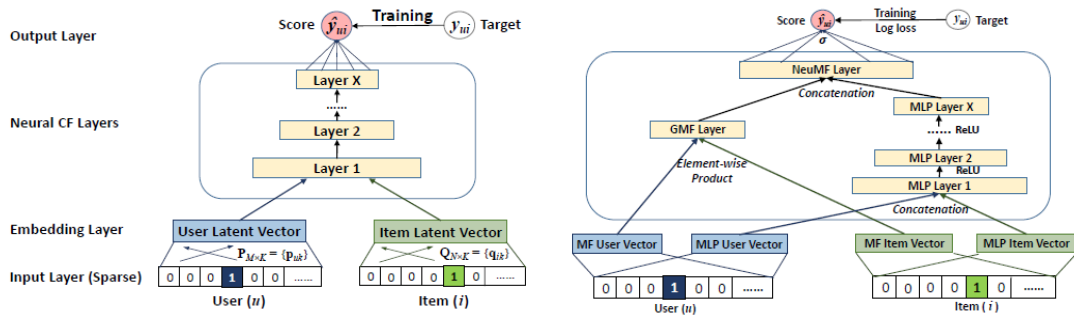


图 10 NCF 和 NeuMF 架构图

NSCR[9] 提出在一些信息导向的数据集（如亚马逊）和社交导向的数据集（如 facebook）上利用一些两个数据集中的公共用户，提出一个 NSCR 模型（如图 11 左图）来实现跨域推荐。信息域的表达上，提出了一个如图 11 右图的多层前馈神经网络（与 NCF 类似，差别在于增加一个 pooling 层），输入层将用户、物品以及它们对应的属性利用 one-hot encoding 编码成稀疏向量，接着 Embedding 层将稀疏向量映射成密集向量。为了将长短不一的向量转变成长度一致的向量，增加了一个 pooling 层，仿照因子分解机提出一种 pairwise pooling，从而可以对用户-属性或者物品-属性这种 pairwise correlation 进行建模。全连接的多层隐层利用矩阵分解思想，将用户表达与物品表达进行 element-wise product，利用 MLP 构建多层隐层，激活函数为 ReLU。网络训练的目标为最小化基于回归的 pair-wise 损失函数。

社交域的表达上，利用类似信息域中顶层的 MLP 隐层结构，对社交用户与物品进行预测，网络的训练的目标为最小化图归一化中由 smoothness 和 fitting 两个约束条件组成的损失函数。最终，两个域的信息融合简单地将两者的目标函数进行相加得到。

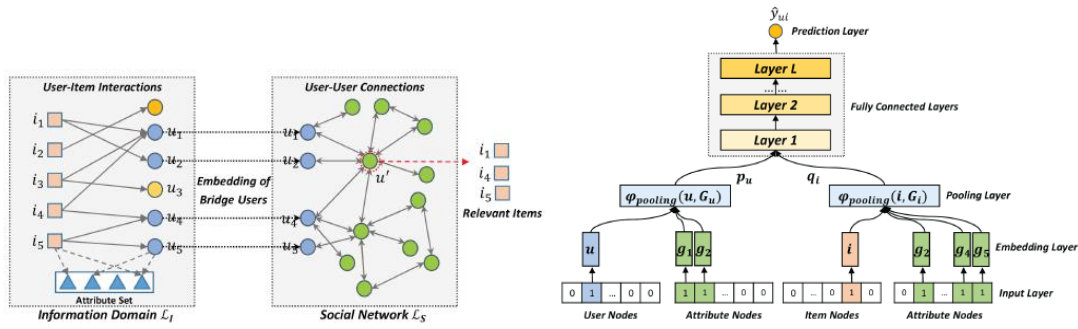


图 11 NSCR 架构图和信息域的 Deep CF 模型

2.4 基于 AE 的推荐算法

AE (Autoencoder) 是一种无监督的学习网络，在推荐系统领域，一般可以利用 AE 在 bottleneck 层来学习低维的特征表达 (如 AutoSVD++、CDL)，或者在 reconstruction 层填补评分矩阵中的缺失值 (如 AutoRec、CDAE)。

AutoRec [10] 是一种 CF 的自编码器框架，利用 AE 网络，将观察向量映射到一个低维的隐式空间 (隐层)，然后再重构到输出空间，从而对评分进行预测。图 12 为基于物品的 AutoRec 架构图，可以发现 AE 的网络非常简单，先对输入的观察向量映射到一个低维空间，然后再进行重构得到恢复后的向量。网络训练的目标函数是最小化 AE 网络输入和输出之间的误差。

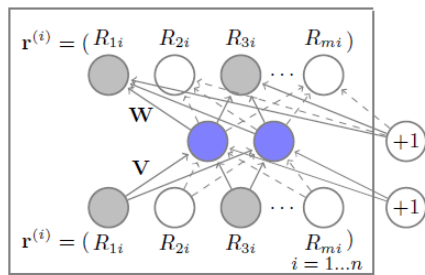


图 12 基于物品的 AutoRec 架构图

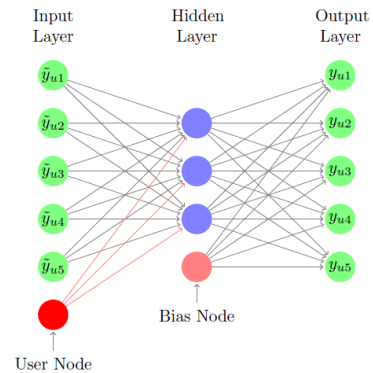


图 13 CDAE 架构图

CDAE [11] 利用去噪的自编码器 (DAE) 来实现 top-n 推荐，与 AutoRec 一样也是单隐层结构，差别在于输入层增加了用户节点，如图 13 所示。输入层包含了一个物品的 I 个节点以及一个用户节点，然后中间的隐层将输入层的 $I+1$ 个

节点映射成低维的隐式表达形式，最后的输出层将低维的隐式向量进行重构，生成与原始输入向量相同维度的输出向量。网络训练的目标为最小化平均重构误差，运用 SGD 对神经网络的参数进行学习。

AutoSVD++ [12] 利用收缩的自编码器（CAE）来构建一种混合协同过滤框架，从而实现推荐，其架构图如图 14 所示。朴素的 AutoSVD 结构如下：利用 CAE 来提取物品的低维特征表达，然后将提取后的物品低维特征添加到 Biased SVD 中，替换其中的物品表达向量。同样的，AutoSVD++则是将 CAE 提取后的物品低维特征替换 SVD++模型的物品表达向量。模型的训练通过最小化正则化的平方误差，参数通过 SGD 算法进行学习。

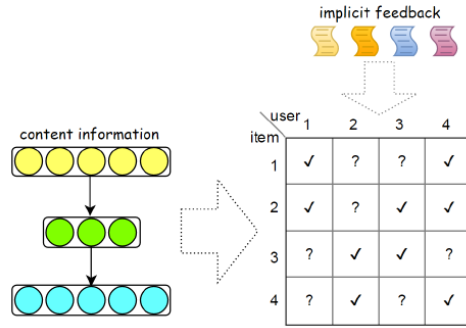


图 14 AutoSVD++架构图

CDL [13] 利用一种贝叶斯版本的堆叠去噪的自编码器（SDAE）来提取物品的内容信息深度表达，从而实现基于协同过滤的推荐，CDL 的图模型如图 15 左图所示。图 16 为一个两层的 SDAE 架构，其中输入层 X_0 为带噪声的物品数据， X_2 为 bottleneck 层，而最后的输出层 X_c 为干净的物品数据， X_0 - X_2 充当编码器的作用，而 X_2 - X_c 充当解码器的作用。SDAE 训练的目标为最小化 X_c 与 X_2 的平方误差（即重构误差）。本文提出的贝叶斯版本的 SDAE 在每一层中都引入高斯变量。最后，将贝叶斯版本的 SDAE 中的 bottleneck 层 X_2 的输出结果（编码器编码结果）作为物品的隐式表达，结合用户的隐式向量进行评分预测。

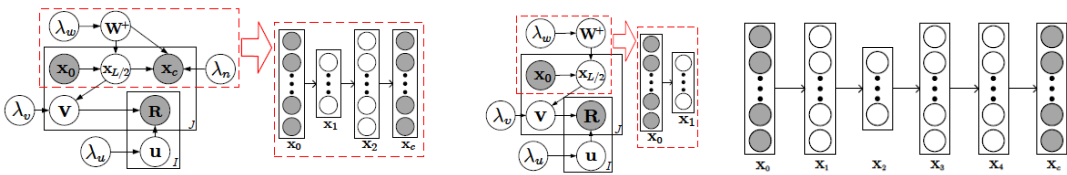


图 15 CDL 图模型

图 16 两层的 SDAE 架构

2.5 基于 DSSM 的推荐算法

DSSM (Deep Semantic Similarity Model) 通过将不同实体映射到同一个低维隐式空间中, 用余弦相似度度量不同实体 (用户与物品) 之间的距离, 从而实现推荐。

DSPR [14] 基于标签将用户和物品映射到一个公共的隐式空间中, 然后通过最大化用户与相关物品的相似度 (最小化用户与不相关物品的相似度), 实现 top-n 推荐, DSPR 架构图如图 17 所示。对于用户和物品, 利用两个共享参数的多层前馈网络, 映射到一个抽象的低维空间, 计算用户和物品间的余弦相似度, 最后对某一用户, 利用 softmax 函数计算用户与物品间的得分, 推荐得分较高的 n 个物品。模型的训练目标是最大化用户与相关物品的 log 函数同时最小化用户与不相关物品的 log 函数, 其中不相关物品集是通过负采样的方式, 随机选择部分物品构造的子集, 从而减低计算复杂度。

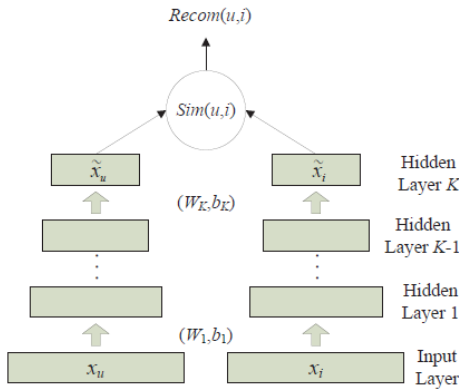


图 17 DSPR 架构图

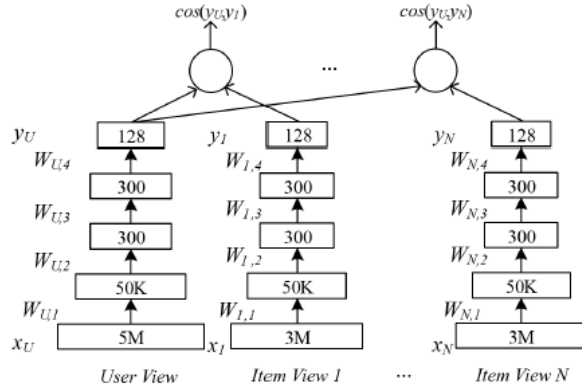


图 18 MV-DNN 架构图

MV-DNN [15] 同样也是利用 DSSM 模型, 使用不同域的数据 (新闻、电影等), 构造多视图的 DSSM 模型, 进行推荐, 其架构图如图 18 所示。与 DSPR 的不同点之处主要是利用了多个领域的数据集 (跨域), 并构造了多个视图的 DSSM 模型协同推荐。

2.6 基于混合模型的推荐算法

由于不同的神经网络具有不同的优势，因此，在推荐系统领域除了使用单一神经网络模型进行推荐之外，融合多种不同类型的神经网络也成为一种新的研究趋势，更能发挥不同模型的长处，进一步提高推荐的准确性。

Chenyi Lei 等 [16]结合 CNN 和 MLP 两种神经网络，实现图片推荐。论文提出了一种 CDL 模型（如图 19），用于将图片信息和用户信息映射到同一个隐式空间中。其中图片的映射网络借鉴 AlexNet 的 CNN 模型，包含了 5 个卷积层、3 个 max-pooling 层以及 3 个全连接层；用户信息的映射网络，通过将用户的浏览记录等文本信息进行向量化，然后进行 k-means 聚类，将缩小标签空间，同时避免同义词的干扰，最后利用词袋模型向量化，作为 MLP 网络的输入，最终获得和 CNN 模型相同维度的输出。

在图片信息和用户信息映射到同一个隐式空间之后，论文又构建了两个距离计算网络，用于计算用户与图片在隐式空间中的距离。最后，利用交叉熵作为损失函数定义目标函数。

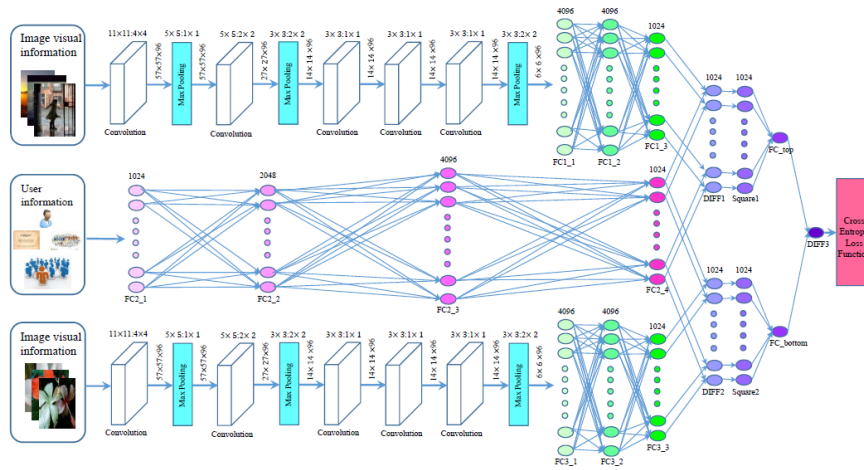


图 19 CDL 架构图

Yuyun Gong 等 [17]结合 CNN 和 RNN 网络，关注于博客推荐领域，结合博客文字及图片，利用 attention 机制结合视觉和语义信息为博客推荐标签，架构图如图 20 所示。图片的特征提取通过将图片分割成 $N \times N$ 块，然后对每一个块，利用 VGGNet 提取 D 维的特征向量；文本的特征提取则是利用了 LSTM 网络进行提取。接着，设计了两种 attention 机制，分别是博客文本导向的视觉 attention

和博客图片导向的语义 attention。最后，结合两种 attention，利用一个单层的 softmax 分类器，计算每一个标签的得分，将得分高的标签作为 top-n 的推荐结果。模型训练的目标为最小化 log 损失函数，参数通过 Adam 进行学习。

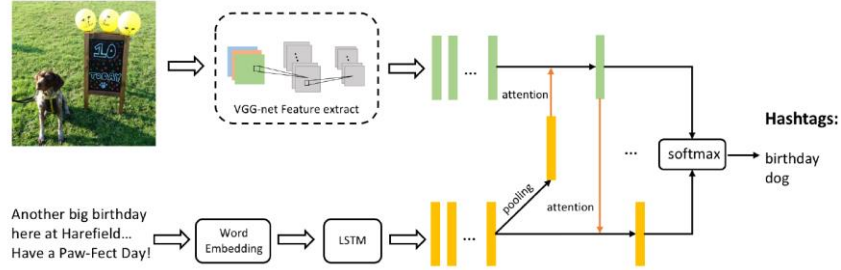


图 20 HashTag 架构图

NCN [18] 结合 CNN 和 RNN 网络，是一种 encoder-decoder 架构的模型，如图 21 所示，用于学术上的相关论文推荐。编码器部分，对一个引用的每一个词，用一个窗口大小为 1 的滑动窗口进行卷积；然后进行 max-pooling 操作，得到一个固定大小的向量，最后用一个全连接层获得不同词组之间特征图的关系，值得注意的是论文还通过重复使用不同大小的滑动窗口，获得不同长度的词组的特征。解码器部分，利用论文的 title 作为 GRU 网络输入，同时前一步的输出结果会反过来影响 attention 机制的权值参数。作者网络额外利用 citation context (query) 和 cited paper's (document) author(s)，同样利用 Encoder 的方法进行处理；最后将三种 Encoder 的结果进行串联，作为 attention 机制共同的输入。模型的训练通过最大化 log 条件概率，参数通过 SGD 算法学习。

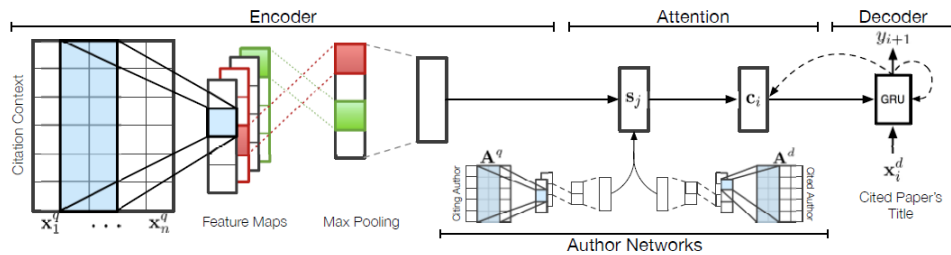


图 21 NCN 架构图

CKE [19] 结合 CNN 和 AE 网络，将结构性知识、文本性知识以及视觉性知识三部分作为知识基，将其作为物品的隐式表达融合到协同过滤框架中，从而提高推荐的质量，CKE 的框架如图 22 所示。对于结构性知识，利用 Bayesian TransR

方法对实体-关系进行处理，将每一个实体从实体空间映射到关系空间，然后计算实体-关系-实体三元组的得分。对于文本性知识，利用 6-layer Bayesian SDAE 网络进行特征提取，将中间的 X3 层网络的输出作为文本的隐式表达。对于视觉性知识，利用 6-layer Bayesian SCAE 网络进行特征提取，同样将中间的 Z3 层网络的输出作为图像的隐式表达。最后，物品的隐式向量表达为上述三种知识基的表达形式外加物品原本的隐式向量，结合用户的隐式向量，作为 CF 框架的输入。模型的训练目标为最大化 log 似然函数，模型参数通过 SGD 算法学习。

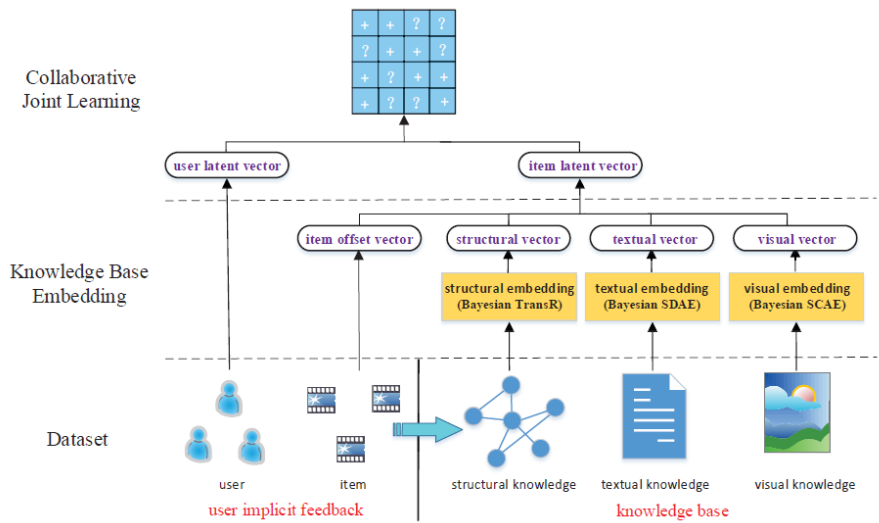


图 22 CKE 架构图

三、推荐结果评估指标

推荐系统中，常见的推荐算法性能评估指标包括了如下几类：

1、Hit Ratio (HR): Top-n 推荐中，推荐物品命中率为 ground truth 物品在推荐列表中的比例。

2、Normalized Discounted Cumulative Gain (NDCG): 表示 Top-n 推荐中，命中物品在推荐列表的位置。

3、Precision@N: 准确率，公式化如下：

$$\text{Precision@N} = \frac{|C_{N,\text{rec}} \cap C_{\text{adopted}}|}{N}$$

4、Recall@N: 召回率，公式化如下：

$$\text{Recall@N} = \frac{|C_{N,\text{rec}} \cap C_{\text{adopted}}|}{|C_{\text{adopted}}|}$$

5、Average Precision (AP): 平均查准率，公式化如下：

$$\text{AP@N} = \frac{\sum_{k=1}^N \text{Precision@k} \times \text{rel}(k)}{\min\{N, |C_{\text{adopted}}|\}}$$

6、Mean Average Precision (MAP): 表示所有用户 AP 得分的均值。

7、Mean Square Errors (MSE): 均方误差，公式化如下：

$$\text{MSE} = \frac{1}{N} \sum_{n=1}^N (r_n - \hat{r}_n)^2$$

8、Root Mean Squared Error (RMSE): 均方根误差，公式化如下：

$$\text{RMSE} = \sqrt{\frac{1}{|T|} \sum_{(u,i) \in T} (r_{ui} - \hat{r}_{ui})^2}$$

上述 8 种推荐算法性能评估指标，在 top-n 问题或者评分预测问题场景下，需要用到不同的评价指标。

参考文献

- [1] Zhang S, Yao L, Sun A. Deep learning based recommender system: A survey and new perspectives[J]. arXiv preprint arXiv:1707.07435, 2017.
- [2] Zheng L, Noroozi V, Yu P S. Joint deep modeling of users and items using reviews for recommendation[C]//Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. ACM, 2017: 425-434.
- [3] Kim D, Park C, Oh J, et al. Convolutional matrix factorization for document context-aware recommendation[C]//Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 233-240.
- [4] Wang S, Wang Y, Tang J, et al. What your images reveal: Exploiting visual contents for point-of-interest recommendation[C]//Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2017: 391-400.
- [5] Shen X, Yi B, Zhang Z, et al. Automatic Recommendation Technology for Learning Resources with Convolutional Neural Network[C]//Educational Technology (ISET), 2016 International Symposium on. IEEE, 2016: 30-34.
- [6] Wu C Y, Ahmed A, Beutel A, et al. Recurrent recommender networks[C]//Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. ACM, 2017: 495-503.
- [7] Bansal T, Belanger D, McCallum A. Ask the gru: Multi-task learning for deep text recommendations[C]//Proceedings of the 10th ACM Conference on Recommender Systems. ACM, 2016: 107-114.
- [8] He X, Liao L, Zhang H, et al. Neural collaborative filtering[C]//Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2017: 173-182.
- [9] Wang X, He X, Nie L, et al. Item silk road: Recommending items from information domains to social users[C]//Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval.

ACM, 2017: 185-194.

- [10] Sedhain S, Menon A K, Sanner S, et al. Autorec: Autoencoders meet collaborative filtering[C]//Proceedings of the 24th International Conference on World Wide Web. ACM, 2015: 111-112.
- [11] Wu Y, DuBois C, Zheng A X, et al. Collaborative denoising auto-encoders for top-n recommender systems[C]//Proceedings of the Ninth ACM International Conference on Web Search and Data Mining. ACM, 2016: 153-162.
- [12] Zhang S, Yao L, Xu X. Autosvd++: an efficient hybrid collaborative filtering model via contractive auto-encoders[J]. arXiv preprint arXiv:1704.00551, 2017.
- [13] Wang H, Wang N, Yeung D Y. Collaborative deep learning for recommender systems[C]//Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2015: 1235-1244.
- [14] Xu Z, Chen C, Lukasiewicz T, et al. Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling[C]//Proceedings of the 25th ACM International on Conference on Information and Knowledge Management. ACM, 2016: 1921-1924.
- [15] Elkahky A M, Song Y, He X. A multi-view deep learning approach for cross domain user modeling in recommendation systems[C]//Proceedings of the 24th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2015: 278-288.
- [16] Lei C, Liu D, Li W, et al. Comparative deep learning of hybrid representations for image recommendations[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2545-2553.
- [17] Zhang Q, Wang J, Huang H, et al. Hashtag recommendation for multimodal microblog using co-attention network[C]//Proceedings of the 26th International Joint Conference on Artificial Intelligence. AAAI Press, 2017: 3420-3426.
- [18] Ebesu T, Fang Y. Neural Citation Network for Context-Aware Citation Recommendation[C]//Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2017: 1093-1096.

- [19]Zhang F, Yuan N J, Lian D, et al. Collaborative knowledge base embedding for recommender systems[C]//Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 2016: 353-362.



论文学习

基于CNN的推荐算法

CNN（Convolution Neural Network）在图像、文字、语音信号处理和特征提取等方面具有优越的性能，因此广泛应用在多媒体等的特征提取领域。



论文学习

Joint deep modeling of users and items using reviews for recommendation

Word Representation

$$V_{1:n}^u = \phi(d_1^u) \oplus \phi(d_2^u) \oplus \phi(d_3^u) \oplus \dots \oplus \phi(d_n^u), \quad (1)$$

CNN Layers

1.卷积层

$$z_j = f(V_{1:n}^u * K_j + b_j) \quad (2)$$

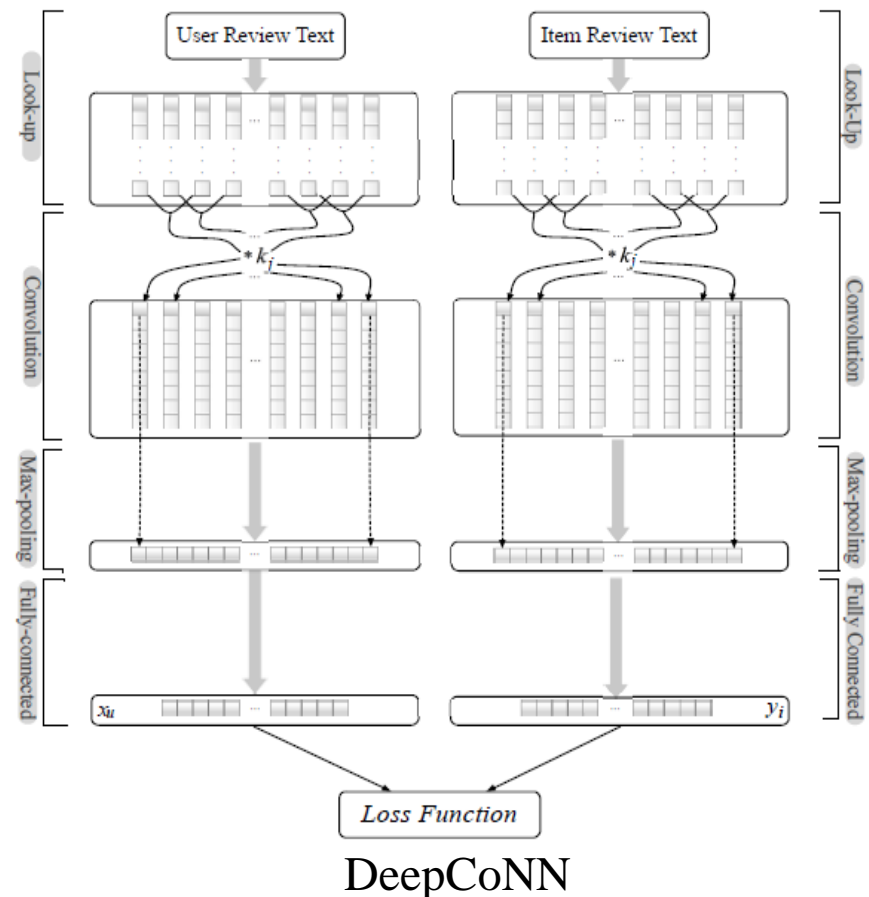
激活函数：ReLU $f(x) = \max\{0, x\}$

2.max-pooling层

$$o_j = \max\{z_1, z_2, \dots, z_{n-t+1}\}$$

$$O = \{o_1, o_2, o_3, \dots, o_{n_1}\}, \quad (5)$$

3.全连接层





论文学习

Joint deep modeling of users and items using reviews for recommendation

- Factorization Machine (FM)

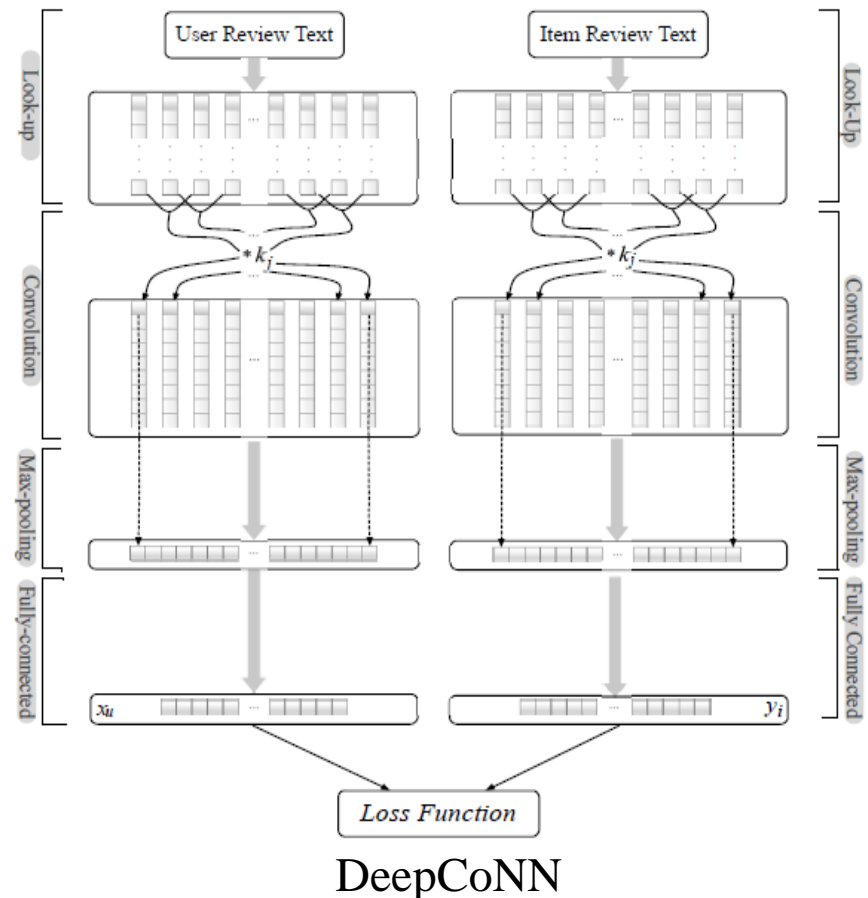
1. 对两个网络的结果进行连接

$$\hat{\mathbf{z}} = (\mathbf{x}_u, \mathbf{y}_i)$$

2. FM建立损失函数

$$J = \hat{w}_0 + \sum_{i=1}^{|\hat{\mathbf{z}}|} \hat{w}_i \hat{z}_i + \sum_{i=1}^{|\hat{\mathbf{z}}|} \sum_{j=i+1}^{|\hat{\mathbf{z}}|} \langle \hat{\mathbf{v}}_i, \hat{\mathbf{v}}_j \rangle \hat{z}_i \hat{z}_j, \quad (7)$$

where \hat{w}_0 is the global bias, \hat{w}_i models the strength of the i_{th} variable in $\hat{\mathbf{z}}$ and $\langle \hat{\mathbf{v}}_i, \hat{\mathbf{v}}_j \rangle = \sum_{f=1}^{|\hat{\mathbf{z}}|} \hat{\mathbf{v}}_{i,f} \hat{\mathbf{v}}_{j,f}$. $\langle \hat{\mathbf{v}}_i, \hat{\mathbf{v}}_j \rangle$ models the second order interactions.





论文学习

Convolutional matrix factorization for document context-aware recommendation

- PMF+CNN

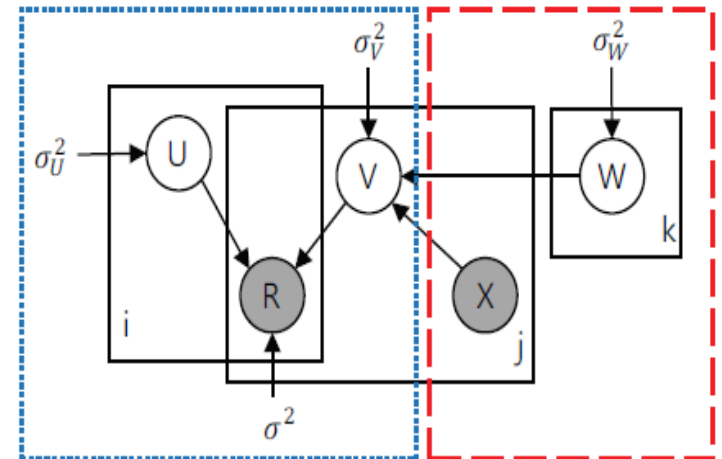
$$p(R|U, V, \sigma^2) = \prod_i^N \prod_j^M N(r_{ij} | u_i^T v_j, \sigma^2)^{I_{ij}}$$
$$p(U|\sigma_U^2) = \prod_i^N N(u_i | 0, \sigma_U^2 I)$$

利用CNN来提取物品表示：

$$v_j = \text{cnn}(W, X_j) + \epsilon_j$$

$$\epsilon_j \sim N(0, \sigma_V^2 I)$$

$$p(V|W, X, \sigma_V^2) = \prod_j^M N(v_j | \text{cnn}(W, X_j), \sigma_V^2 I)$$



ConvMF



论文学习

Convolutional matrix factorization for document context-aware recommendation

■ CNN架构

embedding layer

$$D = \begin{bmatrix} \cdots & w_{i-1} & w_i & w_{i+1} & \cdots \end{bmatrix}$$

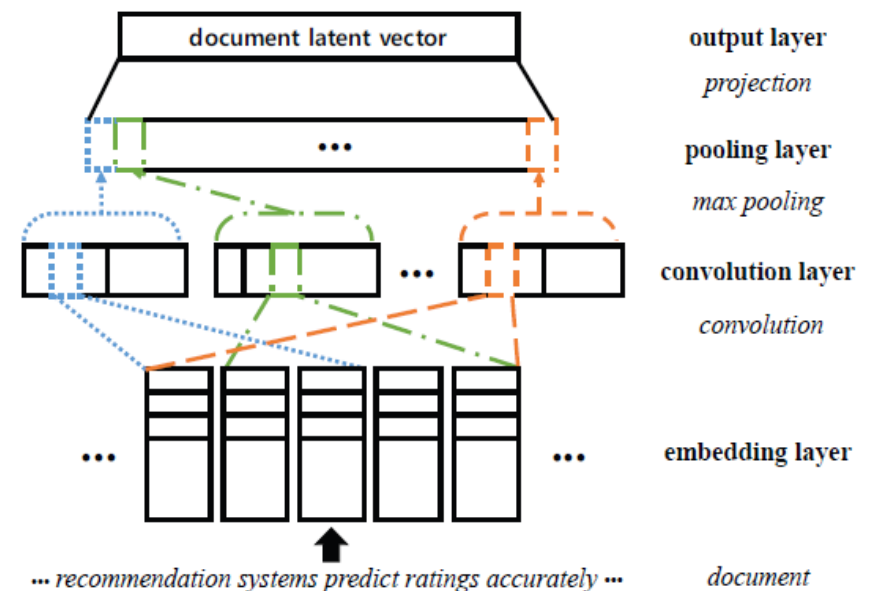
convolution layer (同DeepCoNN)

pooling layer (同DeepCoNN)

output layer

conventional nonlinear projection

$$s = \tanh(W_{f_2} \{ \tanh(W_{f_1} d_f + b_{f_1}) \} + b_{f_2}) \quad (3)$$





论文学习

Automatic Recommendation Technology for Learning Resources with Convolutional Neural Network

- CNN input computing by language model

主题模型LDA，将每一个词表示为属于该主题的概率

- CNN output solved by latent factor model

$$J(U, V) = \sum_{ij} (U_i \cdot V_j - r_{ij})^2 + \lambda_1 \|U\|_1 + \lambda_2 \|V\|_1 \quad (5)$$

	Book 1	Book 2	Book 3	Book 4	...	Book n
Tom	8			10	...	2
Mary		3		7	...	
Joan			5		...	8
Bob	9	9			...	
...
Richard				9	...	2
Carol		1	9		...	

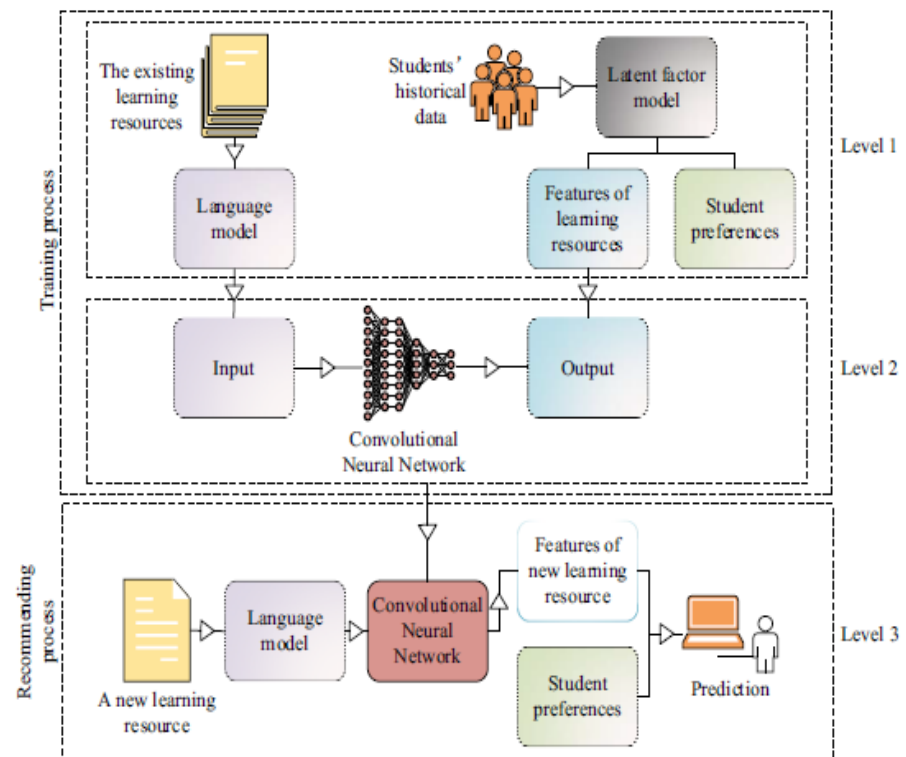
Historical Ratings Matrix (R)

	Literature	Computer	...	Science Fiction
Tom	1.85	0.21	...	0.87
Mary	0.32	-0.47	...	1.11
Joan	0.21	-0.32	...	0.89
Bob	1.33	0.99	...	0.54
...
Richard	1.77	0.15	...	0.84
Carol	0.16	-0.77	...	0.63

Student's Preferences (U)

	Book 1	Book 2	Book 3	Book 4	...	Book n
Literature	1.54	0.25	-0.16	0.88	...	1.88
Computer	0.05	1.66	0.86	-0.21	...	0.04
...
Science Fiction	0.36	0.88	1.05	1.69	...	-0.31

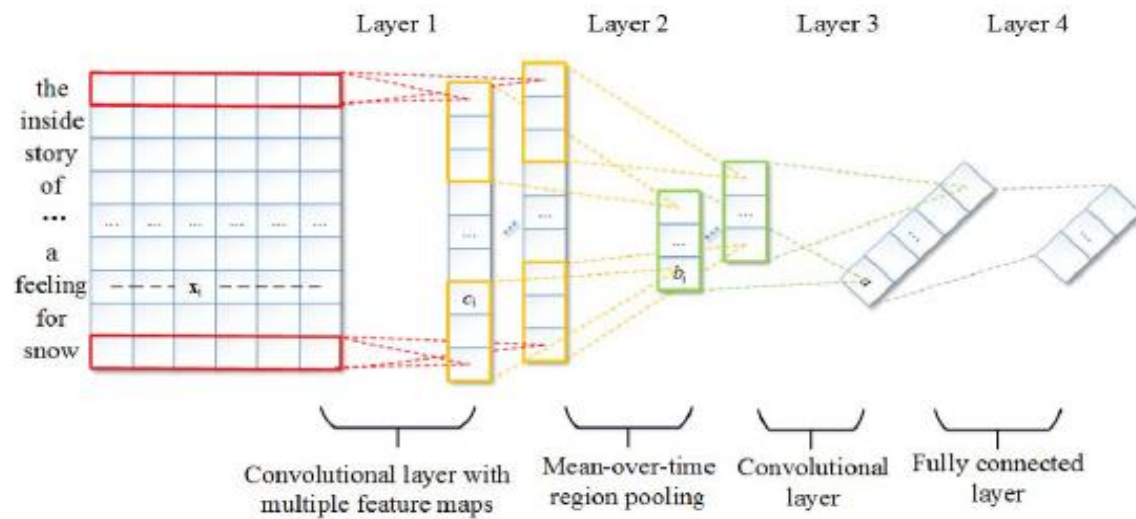
Features of Learning Resources (V)





论文学习

Automatic Recommendation Technology for Learning Resources with Convolutional Neural Network



- Construction of the CNN model
- 网络训练

$$\arg \min_{w, b} \sum_i \|y_i' - y_i\|^2 \quad (4)$$



论文学习

基于RNN的推荐算法

RNN (Recurrent Neural Network) 适合对有时间性、时序性的特征进行提取，或者是对有上下文关联信息的提取。RNN应用于推荐系统领域比较常见的网络模型主要为LSTM和GRU模型。



论文学习

Recurrent recommender networks

- 利用LSTM建立用户状态

$$[f_t, i_t, o_t] = \sigma [W [h_{t-1}, z_t] + b] \quad (1)$$

$$l_t = \tanh [V [h_{t-1}, z_t] + d] \quad (2)$$

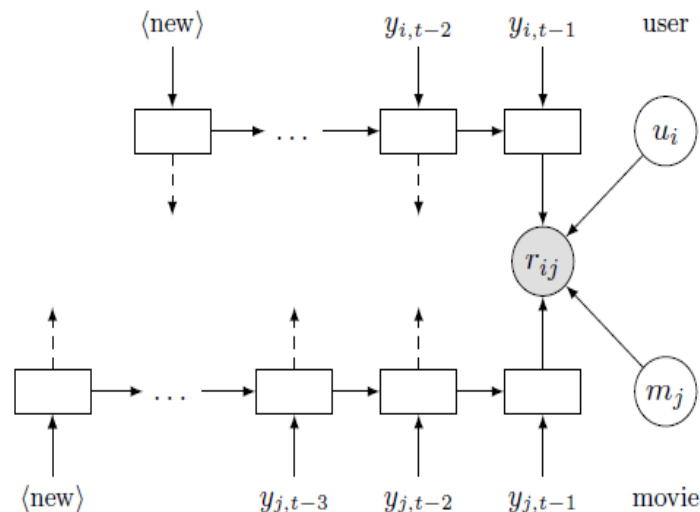
$$c_t = f_t \cdot c_{t-1} + i_t \cdot l_t \quad (3)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (4)$$

where f_t , i_t and o_t denote forget gate, input gate, and output gate respectively.

$$u_t := \text{LSTM}(u_{t-1}, y_t)$$

- 同理，建立电影状态表示





论文学习

Recurrent recommender networks

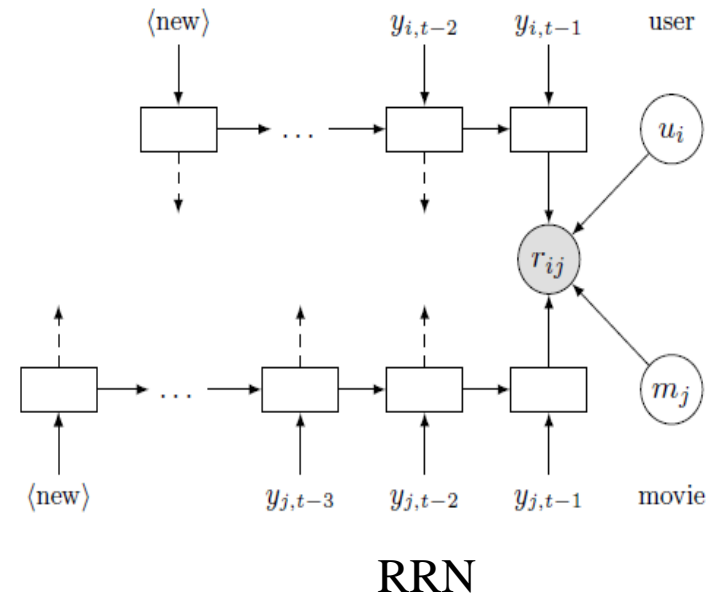
■ 评分预测

(静态部分+动态部分)

$$\hat{r}_{ij|t} = f(u_{it}, m_{jt}, u_i, m_j) := \langle \tilde{u}_{it}, \tilde{m}_{jt} \rangle + \langle u_i, m_j \rangle \quad (7)$$

where \tilde{u}_{it} and \tilde{m}_{jt} are affine functions of u_{it} and m_{jt} respectively. That is, we have

$$\tilde{u}_{it} = W_{\text{user}} u_{it} + b_{\text{user}} \text{ and } \tilde{m}_{jt} = W_{\text{movie}} m_{jt} + b_{\text{movie}}$$





论文学习

Ask the gru: Multi-task learning for deep text recommendations

GRUs提取文本特征

与其他word embedding的区别在于，可以获得Order-Sensitive的文本特征

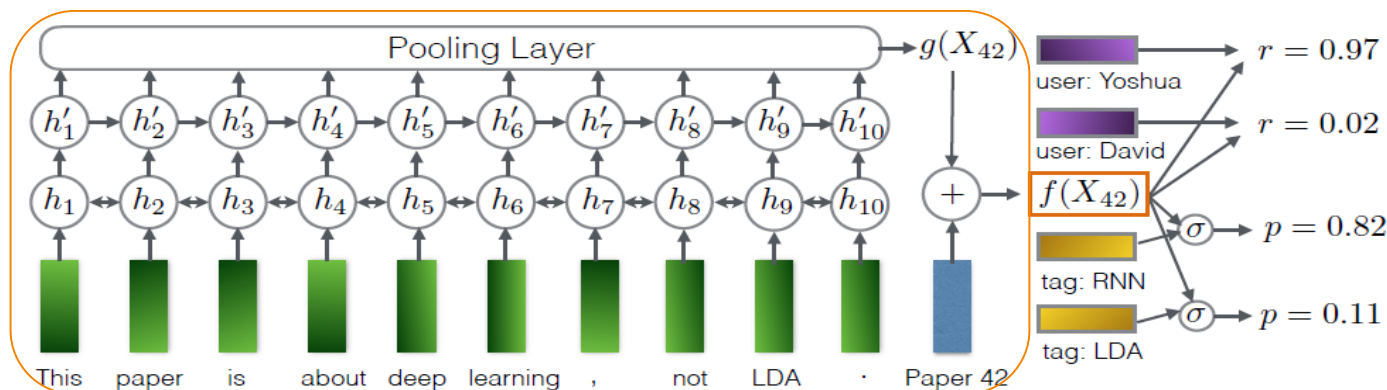
The GRU hidden vector output at step t , h_t , for the input sequence $X_j = (w_1, \dots, w_t, \dots, w_{n_j})$ is given by:

$$\begin{bmatrix} f_t \\ o_t \end{bmatrix} = \sigma \left(\theta^1 \begin{bmatrix} \tilde{e}_{w_t} \\ h_{t-1} \end{bmatrix} + b \right) \quad (6)$$

$$c_t = \tanh(\theta_w^2 \tilde{e}_{w_t} + f_t \odot \theta_h^2 h_{t-1} + b_c) \quad (7)$$

$$h_t = (1 - o_t) \odot h_{t-1} + o_t \odot c_t \quad (8)$$

$$f(X_j) = g(X_j) + \tilde{v}_j \quad (4)$$





论文学习

Ask the gru: Multi-task learning for deep text recommendations

Multi-Task Learning

1. 用户与文本 (LFM)

$$\hat{r}_{ij} = b_i + b_j + \tilde{u}_i^T \tilde{v}_j, \quad (1)$$

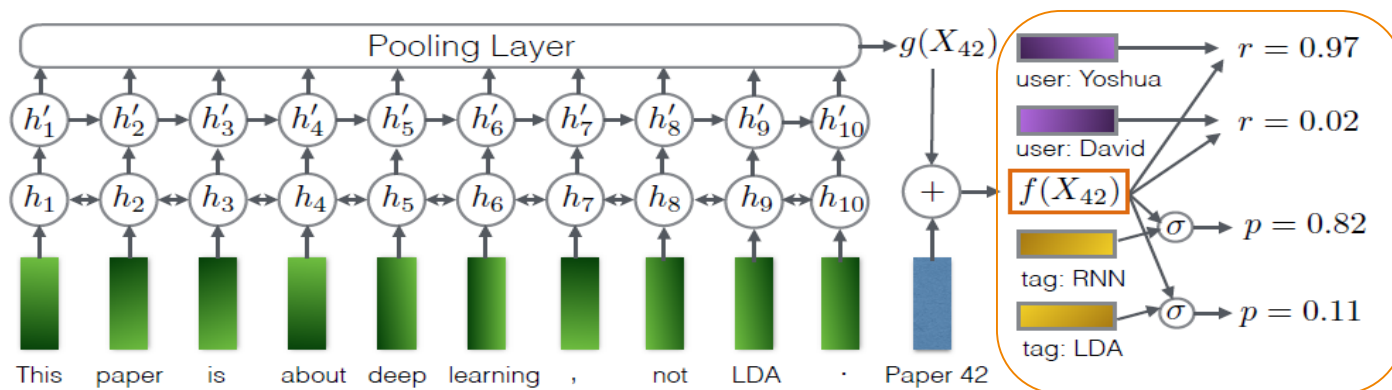
$$C_R(\theta) = \frac{1}{|R|} \sum_{(i,j) \in R} c_{ij} (\hat{r}_{ij} - r_{ij})^2 + \Omega(\theta)$$

2. 标签与文本

$$P(t_{jl} = 1) = p_{jl} = \sigma(f(X_j)^T \tilde{t}_l),$$

$$C_T(\theta) = \frac{1}{|T|} \sum_j \sum_l \{t_{jl} \log p_{jl} + c'_{jl} (1 - t_{jl}) \log(1 - p_{jl})\}$$

$$C(\theta) = \lambda C_R(\theta) + (1 - \lambda) C_T(\theta)$$





论文学习

基于MLP的推荐算法

MLP (Multilayer Perceptron) 是一种简单而又有效的神经网络模型，也是许多其他类型的神经网络模型的基础。



论文学习

Neural collaborative filtering

■ NCF框架

Input层：

用户和物品分别以one-hot encoding的方式进行编码，形成稀疏向量

Embedding层：

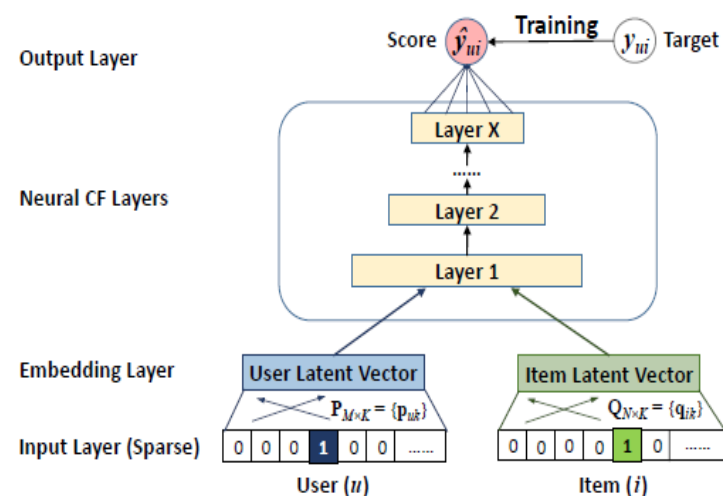
全连接层将稀疏向量投影成一个密集向量（隐式向量）

Neural CF层：多层前馈CF网络

$$f(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I) = \phi_{out}(\phi_X(\dots \phi_2(\phi_1(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I))\dots)),$$

最终，得到NCF预测模型：

$$\hat{y}_{ui} = f(\mathbf{P}^T \mathbf{v}_u^U, \mathbf{Q}^T \mathbf{v}_i^I | \mathbf{P}, \mathbf{Q}, \Theta_f),$$



NCF



论文学习

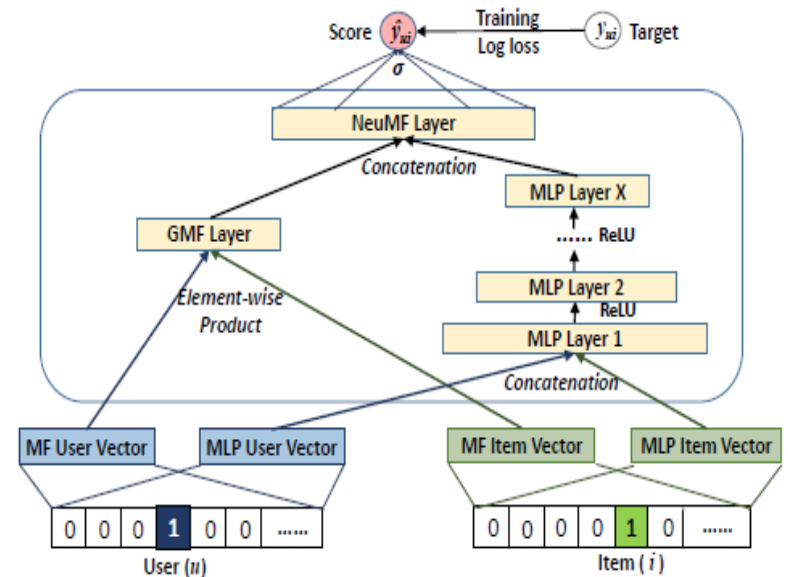
Neural collaborative filtering

- GMF+MLP升级版

$$\phi^{GMF} = \mathbf{p}_u^G \odot \mathbf{q}_i^G,$$

$$\phi^{MLP} = a_L(\mathbf{W}_L^T(a_{L-1}(\dots a_2(\mathbf{W}_2^T \begin{bmatrix} \mathbf{p}_u^M \\ \mathbf{q}_i^M \end{bmatrix} + \mathbf{b}_2)\dots)) + \mathbf{b}_L),$$

$$\hat{y}_{ui} = \sigma(\mathbf{h}^T \begin{bmatrix} \phi^{GMF} \\ \phi^{MLP} \end{bmatrix}),$$



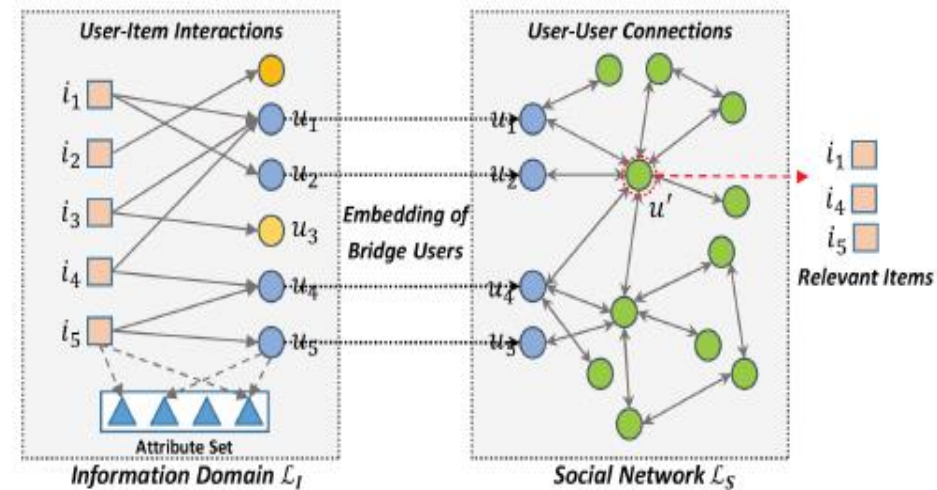
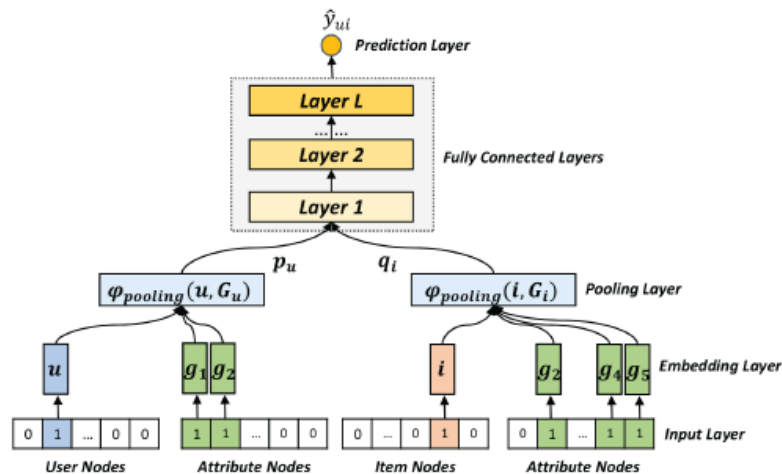


论文学习

Item silk road: Recommending items from information domains to social users

信息导向的数据集（如亚马逊）和社交导向的数据集（如facebook）上利用一些两个数据集中的公共用户，实现跨域推荐。

信息域



NSCR



论文学习

Item silk road: Recommending items from information domains to social users

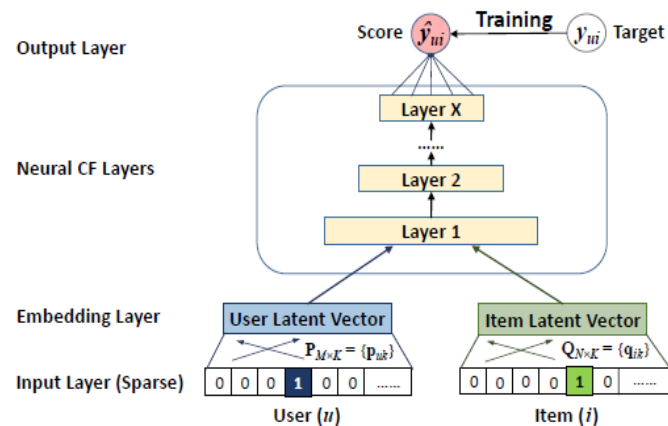
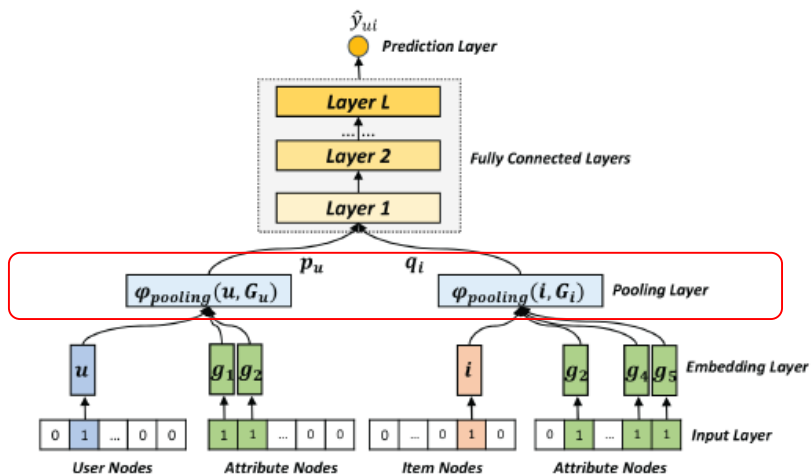
■ 信息域

与NCF模型的差异在于增加pooling层

(pairwise pooling)

$$p_u = \varphi_{pairwise}(u, \{g_t^u\}) = \sum_{t=1}^{V_u} u \odot g_t^u + \sum_{t=1}^{V_u} \sum_{t'=t+1}^{V_u} g_t^u \odot g_{t'}^u,$$

$$q_i = \varphi_{pairwise}(i, \{g_t^i\}) = \sum_{t=1}^{V_i} i \odot g_t^i + \sum_{t=1}^{V_i} \sum_{t'=t+1}^{V_i} g_t^i \odot g_{t'}^i.$$



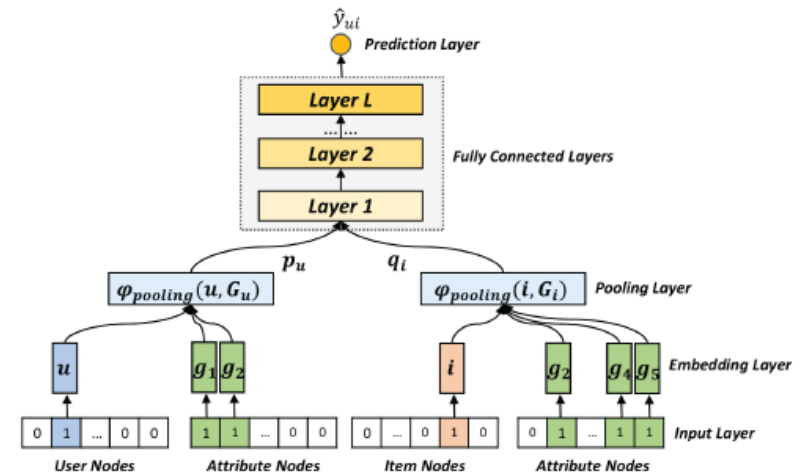


论文学习

Item silk road: Recommending items from information domains to social users

■ 社交域

$$\begin{cases} e_1 = \sigma_1(W_1(p_{u'} \odot q_i) + b_1) \\ \dots\dots\dots \\ e_L = \sigma_L(W_L e_{L-1} + b_L) \\ \hat{y}_{u'i} = w^\top e_L \end{cases}$$



■ 两域融合

$$\mathcal{L} = \mathcal{L}_I(\Theta_I) + \mathcal{L}_S(\Theta_S), \quad (3)$$

where \mathcal{L}_I (or \mathcal{L}_S) denotes the objective function of the information domain (or social domain) learning with parameters Θ_I (or Θ_S),



论文学习

基于AE的推荐算法

AE (Autoencoder) 是一种无监督的学习网络，在推荐系统领域，一般可以利用AE在bottleneck层来学习低维的特征表达（如AutoSVD++、CDL），或者在reconstruction层填补评分矩阵中的缺失值（如AutoRec、CDAE）。



论文学习

Collaborative denoising auto-encoders for top-n recommender systems

■ CDAE架构

单隐层神经网络

Input层：

一个物品的I个节点以及一个用户节点

Hidden层：

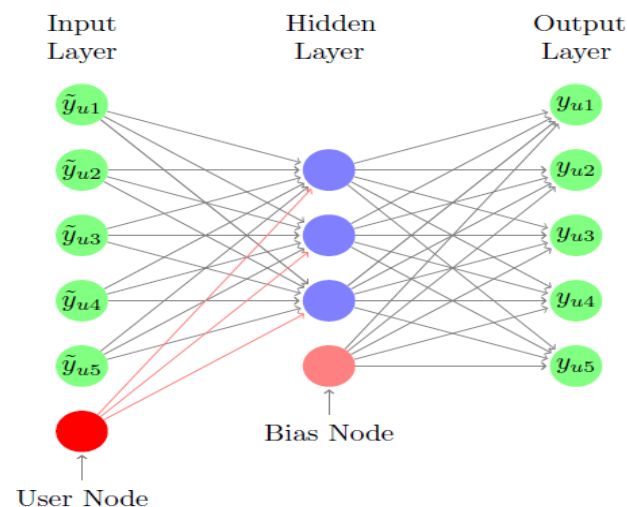
K个节点，全连接

$$z_u = h \left(W^T \tilde{y}_u + V_u + b \right)$$

Output层：

将低维的隐式向量进行重构

$$\hat{y}_{ui} = f \left(W_i'^T z_u + b_i' \right)$$



CDAE



论文学习

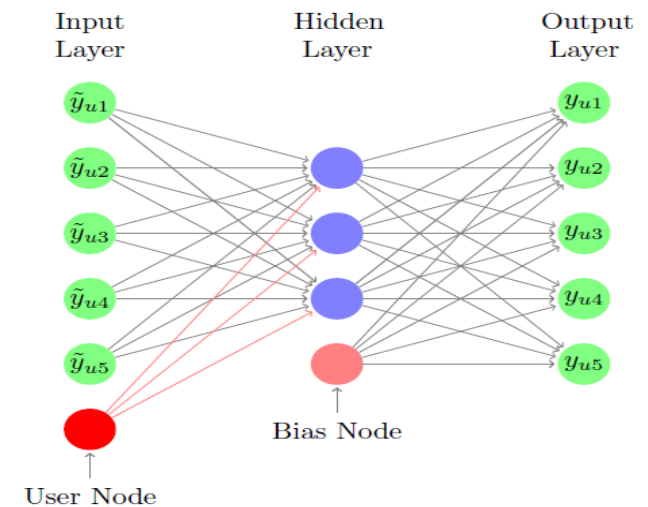
Collaborative denoising auto-encoders for top-n recommender systems

■ 模型学习：

Min average reconstruction error:

$$\arg \min_{W, W', V, b, b'} \frac{1}{U} \sum_{u=1}^U \mathbb{E}_{p(\tilde{y}_u | y_u)} [\ell(\tilde{y}_u, \hat{y}_u)] + \mathcal{R}(W, W', V, b, b'),$$

$$\mathcal{R}(\cdot) = \frac{\lambda}{2} (\|W\|_2^2 + \|W'\|_2^2 + \|V\|_2^2 + \|b\|_2^2 + \|b'\|_2^2)$$





论文学习

Autosvd++: an efficient hybrid collaborative filtering model via contractive auto-encoders

CAE

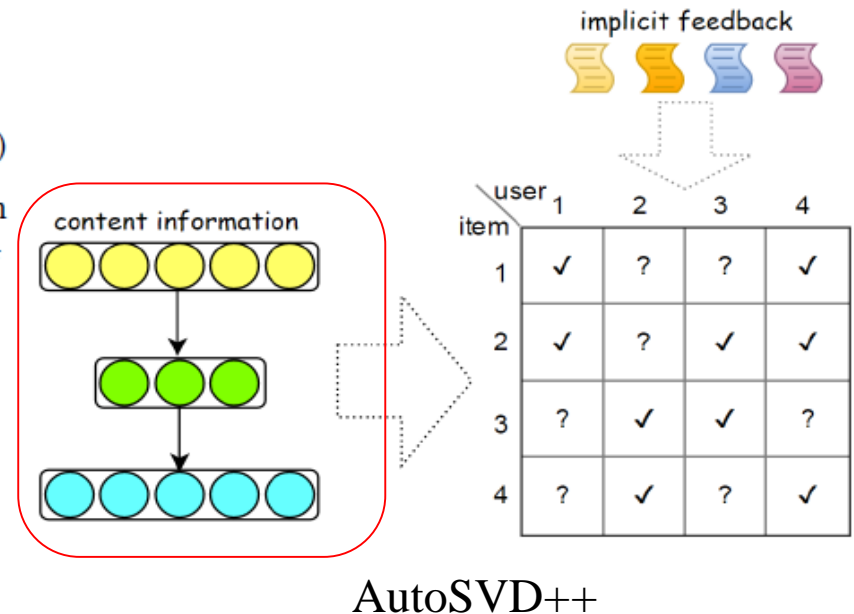
单隐层的无监督神经网络

$$g(f(x)) = s_g(W' s_f(Wx + b_h) + b_y) \quad (4)$$

s_f is a nonlinear activation function, s_g is the decoder's activation function, $b_h \in \mathbb{R}^{d_h}$ and $b_y \in \mathbb{R}^{d_x}$ are bias vectors, $W \in \mathbb{R}^{d_h \times d_x}$

网络训练

$$J_{cae}(\theta) = \sum_{x \in D_n} (L(x, g(f(x))) + \lambda \|J_f(x)\|_F^2)$$





论文学习

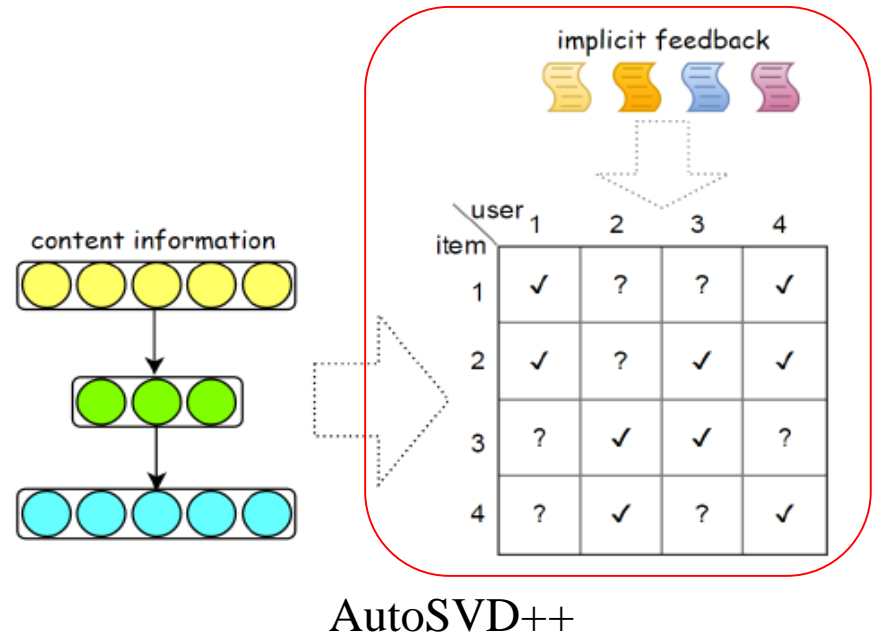
Autosvd++: an efficient hybrid collaborative filtering model via contractive auto-encoders

- AutoSVD : CAE+SVD

$$\hat{r}_{ui} = \mu + b_u + b_i + (\beta \cdot cae(C_i) + \epsilon_i)^T U_u$$

- AutoSVD ++: CAE+SVD++

$$\hat{r}_{ui} = \mu + b_u + b_i + (\beta \cdot cae(C_i) + \epsilon_i)^T (U_u + |N(u)|^{-\frac{1}{2}} \sum_{j \in N(u)} y_j)$$



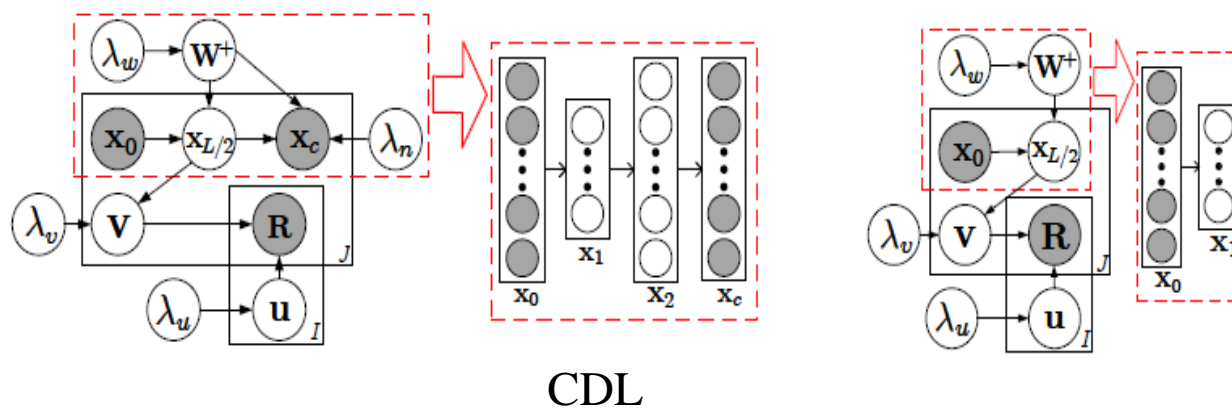


论文学习

Collaborative deep learning for recommender systems

■ CDL

将贝叶斯版本的SDAE中的bottleneck层 x_2 的输出结果（编码器编码结果）作为物品的隐式表达，结合用户的隐式向量进行评分预测



CDL



论文学习

Collaborative deep learning for recommender systems

■ SDAE

输入层 X_0 为带噪声的物品数据

X_2 为bottleneck层，可以输出低维隐式表达

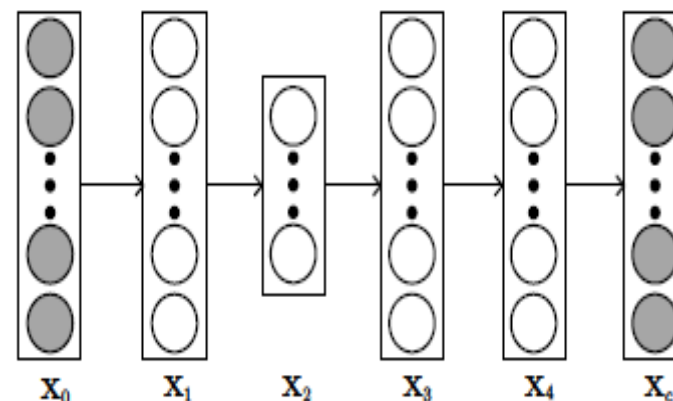
输出层 X_c 为干净的物品数据

X_0 - X_2 充当编码器，而 X_2 - X_c 充当解码器

网络训练，最小化重构误差

$$\min_{\{\mathbf{W}_l\}, \{\mathbf{b}_l\}} \|\mathbf{X}_c - \mathbf{X}_L\|_F^2 + \lambda \sum_l \|\mathbf{W}_l\|_F^2,$$

■ Bayesian SDAE



- For each layer l of the SDAE network,
 - For each column n of the weight matrix \mathbf{W}_l , draw $\mathbf{W}_{l,*n} \sim \mathcal{N}(\mathbf{0}, \lambda_w^{-1} \mathbf{I}_{K_l})$.
 - Draw the bias vector $\mathbf{b}_l \sim \mathcal{N}(\mathbf{0}, \lambda_w^{-1} \mathbf{I}_{K_l})$.
 - For each row j of \mathbf{X}_l , draw $\mathbf{X}_{l,j*} \sim \mathcal{N}(\sigma(\mathbf{X}_{l-1,j*} \mathbf{W}_l + \mathbf{b}_l), \lambda_s^{-1} \mathbf{I}_{K_l})$. (1)
- For each item j , draw a clean input ¹ $\mathbf{X}_{c,j*} \sim \mathcal{N}(\mathbf{X}_{L,j*}, \lambda_n^{-1} \mathbf{I}_J)$.



论文学习

基于DSSM的推荐算法

DSSM (Deep Semantic Similarity Model)
通过将不同实体映射到同一个低维隐式空间中，用余弦相似度度量不同实体（用户与物品）之间的距离，从而实现推荐。



论文学习

Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling

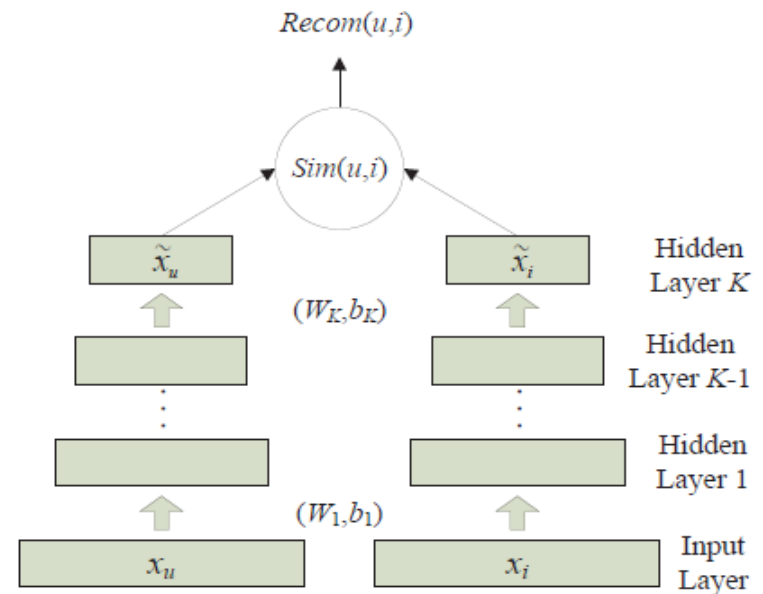
■ DSPR

利用两个共享参数的多层前馈网络，将用户和物品映射到一个抽象的低维空间

$$\begin{aligned}h_j(u) &= \tanh(W_j h_{j-1}(u) + b_j), \\h_j(i) &= \tanh(W_j h_{j-1}(i) + b_j),\end{aligned}$$

计算cos相似度

$$Sim(u, i) = \frac{\tilde{x}_u \cdot \tilde{x}_i}{\|\tilde{x}_u\| \|\tilde{x}_i\|},$$



DSPR



论文学习

Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling

■ 推荐

softmax函数计算用户与物品间的得分，推荐得分较高的n个物品

$$Recom(u, i) = \frac{e^{Sim(u, i)}}{\sum_{i' \in I} e^{Sim(u, i')}}.$$

■ 模型训练

最大化用户与相关物品的log函数同时最小化用户与不相关物品的log函数

$$\begin{aligned} L(\Theta) &= - \sum_{(u, i^*)} \log(Recom(u, i^*)) \\ &= - \sum_{(u, i^*)} [\log(e^{Sim(u, i^*)}) - \log(\sum_{i' \in I} e^{Sim(u, i')})], \end{aligned}$$

■ 负采样

$$L^{NS}(\Theta) = - \sum_{(u, i^*)} [\log(e^{Sim(u, i^*)}) - \log(\sum_{(u, i^-) \in D^-} e^{Sim(u, i^-)})],$$



论文学习

基于混合模型的推荐算法

由于不同的神经网络具有不同的优势，因此，在推荐系统领域除了使用单一神经网络模型进行推荐之外，融合多种不同类型的神经网络也成为一种新的研究趋势，更能发挥不同模型的长处，进一步提高推荐的准确性。



论文学习

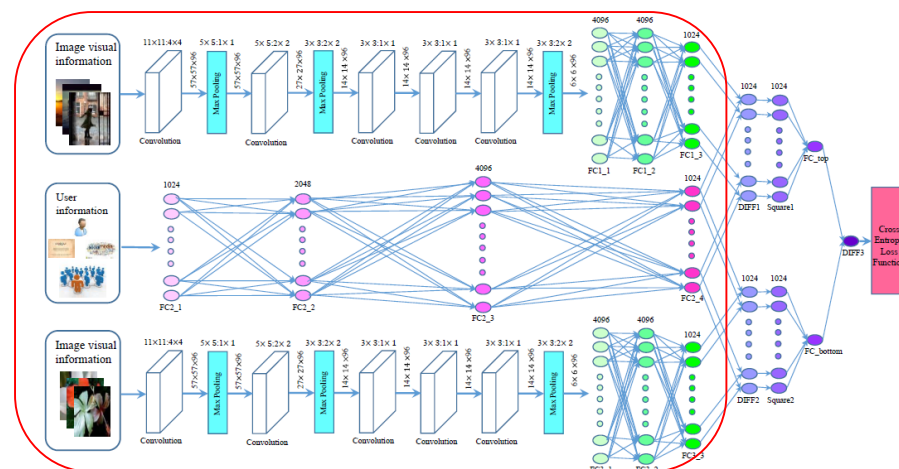
Comparative deep learning of hybrid representations for image recommendations

■ CNN部分

图片的映射网络借鉴AlexNet的CNN模型，包含了5个卷积层、3个max-pooling层以及3个全连接层

■ MLP部分

用户信息的映射网络通过将用户的浏览记录等文本信息进行向量化，然后进行k-means聚类，将缩小标签空间，同时避免同义词的干扰，最后利用词袋模型向量化，作为MLP网络的输入，最终获得和CNN模型相同维度的输出



CDL (CNN+MLP)



论文学习

Comparative deep learning of hybrid representations for image recommendations

距离计算网络

element-wise difference

element-wise square

full-connection

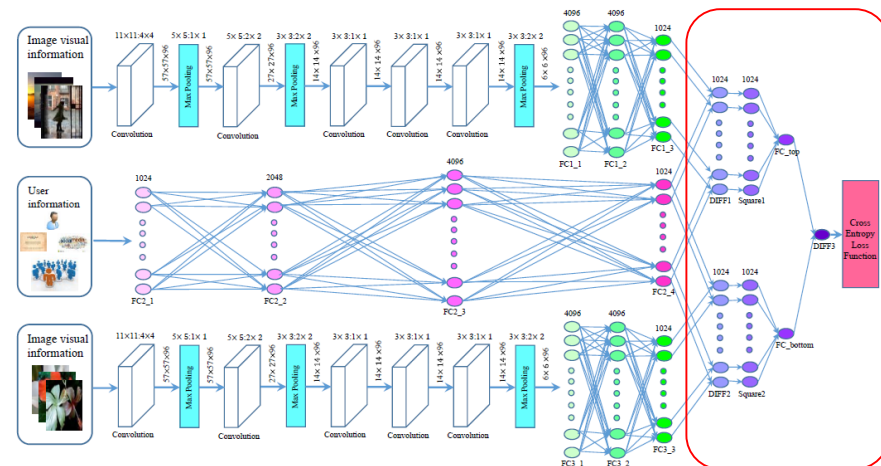
模型训练

三元组的距离

$$o_{ij}^{U_t} = \hat{D}(\pi(U_t), \phi(i)) - D(\pi(U_t), \phi(j)),$$

目标函数 (交叉熵)

$$\min_{\pi, \phi, D} \mathcal{L}(\{\mathcal{T}_t\}) = \sum_t -\bar{P}_{ij}^{U_t} \log(P_{ij}^{U_t}) - (1 - \bar{P}_{ij}^{U_t}) \log(1 - P_{ij}^{U_t}).$$



CDL (CNN+MLP)

$$\bar{P}_{ij}^{U_t} = \begin{cases} 0, & (i = I_t^+, j = I_t^-) \\ 1, & (i = I_t^-, j = I_t^+) \end{cases}$$



论文学习

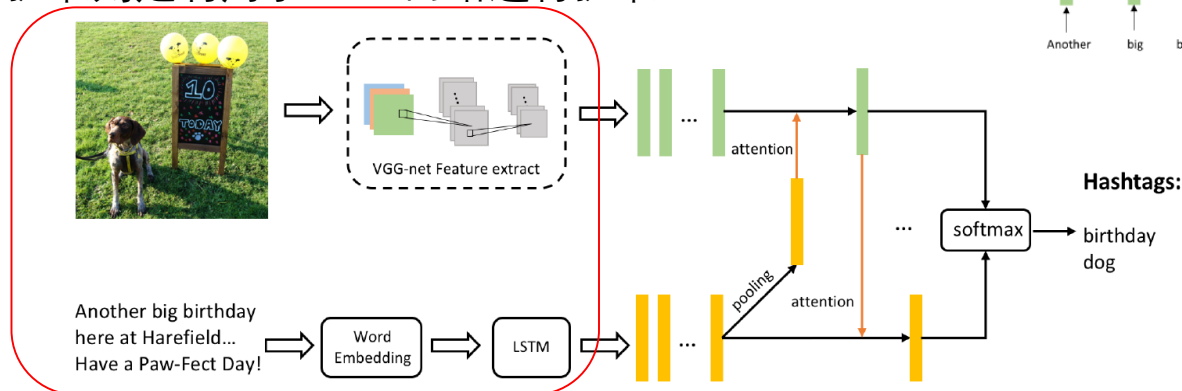
Hashtag recommendation for multimodal microblog using co-attention network

■ CNN部分

图片的特征提取通过将图片分割成 $N \times N$ 块，然后对每一个块，利用VGGNet提取D维的特征向量

■ RNN部分

文本的特征提取则是利用了LSTM网络进行提取



HashTag (CNN+RNN)

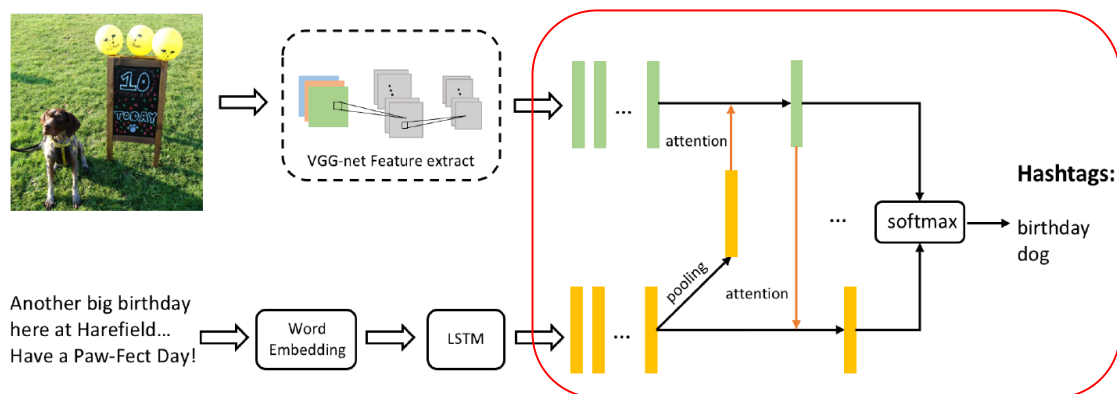


论文学习

Hashtag recommendation for multimodal microblog using co-attention network

- attention机制
- 标签预测：

$$P(y^d = a|h^d, \theta) = \frac{\exp(\theta^{(a)T}(W_f f + b_f))}{\sum_{j \in A} \exp(\theta^{(j)T}(W_f f + b_f))},$$





论文学习

Neural Citation Network for Context-Aware Citation Recommendation

- NCN (encoder-decoder架构)

编码器部分，对引用的每一个词，用一个窗口大小为 l 的滑动窗口进行卷积；然后进行max-pooling操作，最后用一个全连接层获得不同词组之间特征图的关系

解码器部分，利用论文作为GRU网络输入，同时前一步的输出结果会反过来影响attention机制的权值参数</p></div><div data-bbox="152 622 790 897" data-label="Diagram"></div><div data-bbox="46 922 952 975" data-label="Page-Footer"><p>Ebesu T, Fang Y. Neural Citation Network for Context-Aware Citation Recommendation[C]//Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 2017: 1093-1096.</p></div>



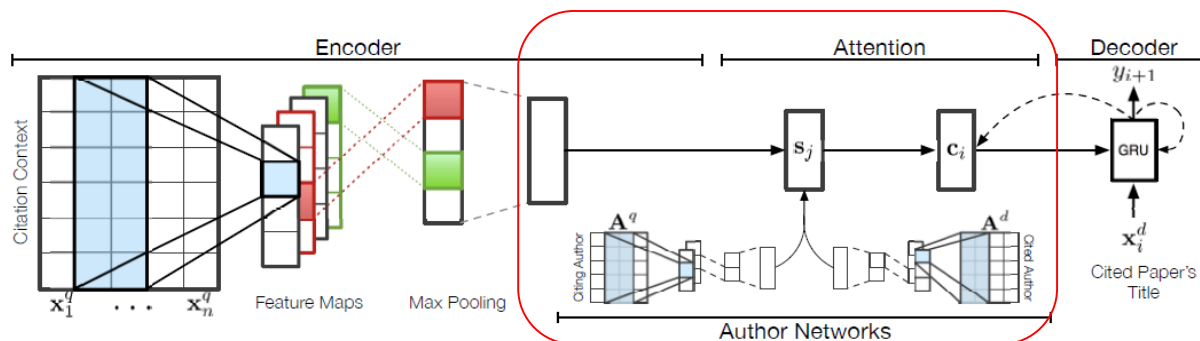
论文学习

Neural Citation Network for Context-Aware Citation Recommendation

- Attention

$$c_i = \sum_j \alpha_{ij} s_j \quad \text{where } \alpha_{ij} = \text{softmax}(v^T \tanh(W_a h_{i-1} + U_a s_j))$$

$$s_j = [f(X^q) \oplus f(A^q) \oplus f(A^d)]_j$$



NCN (CNN+RNN)



论文学习

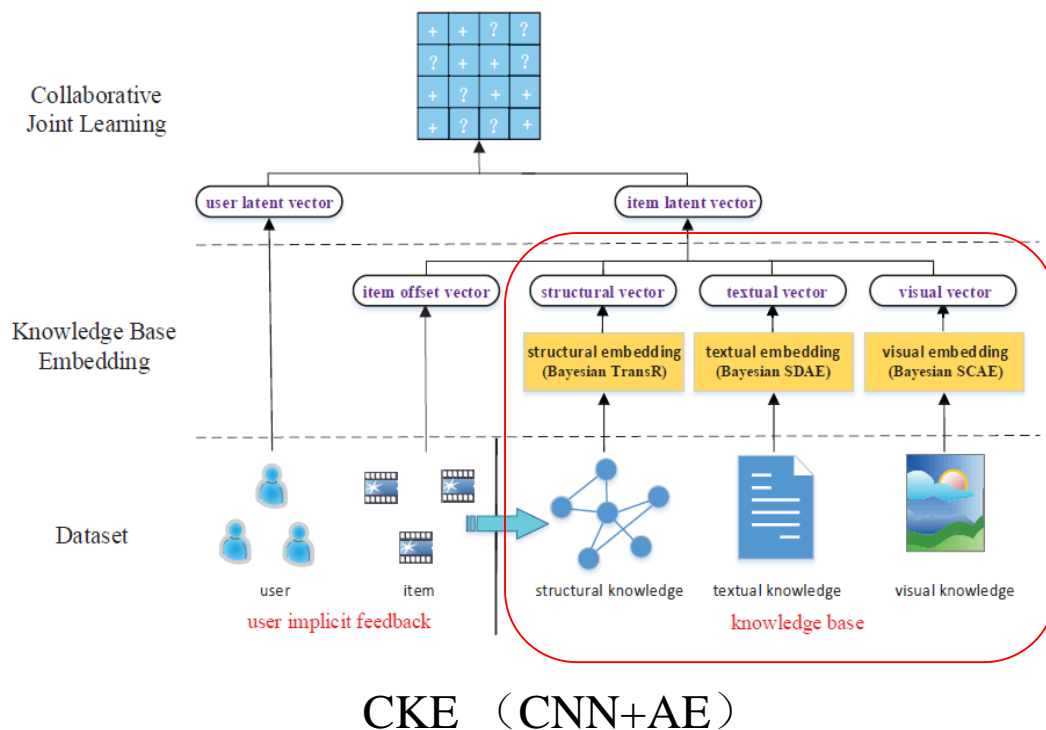
Collaborative knowledge base embedding for recommender systems

■ 三种知识基

结构性知识, 利用Bayesian TransR

文本性知识, 利用6-layer Bayesian SDAE

视觉性知识, 利用6-layer Bayesian SCAE





论文学习

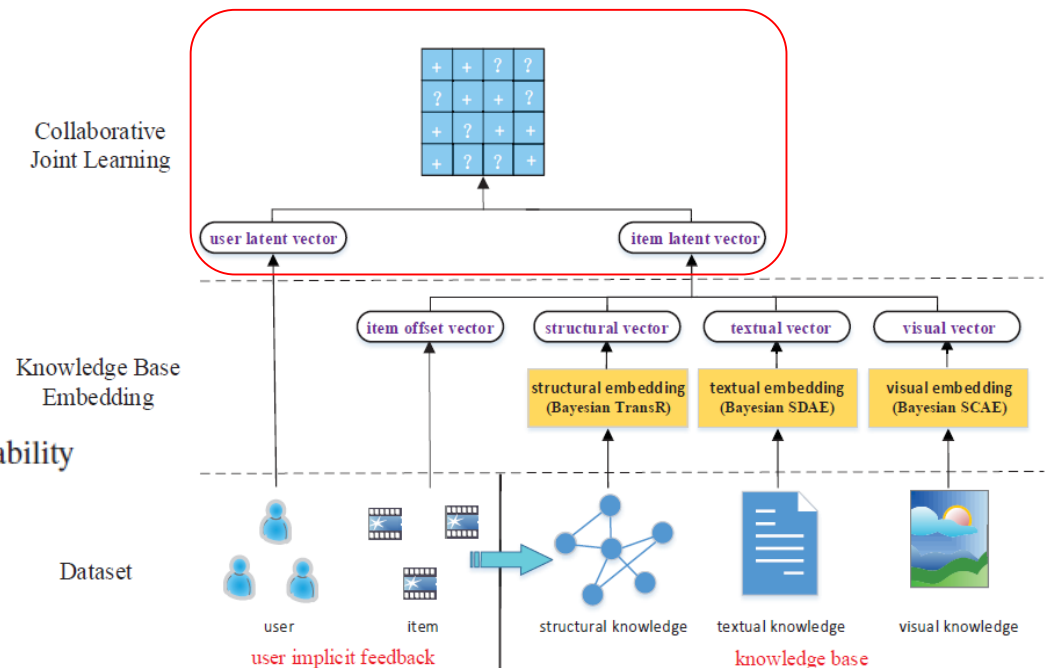
Collaborative knowledge base embedding for recommender systems

物品隐式表达

$$\mathbf{e}_j = \boldsymbol{\eta}_j + \mathbf{v}_j + \mathbf{X} \frac{L_t}{2}, j^* + \mathbf{Z} \frac{L_v}{2}, j^*$$

$$\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \lambda_U^{-1} \mathbf{I}).$$

For each triple $(i, j, j') \in \mathcal{D}$, draw from the probability $\sigma(\mathbf{u}_i^T \mathbf{e}_j - \mathbf{u}_i^T \mathbf{e}_{j'})$.



CKE (CNN+AE)

谢谢！

