

基于异构信息网络元路径作张量分解的深度 学习推荐系统

许荣海, 王昌栋

中山大学 计算机学院, 广州 中国 510000

摘要 过去基于学习用户和物品的表征向量的推荐系统算法在大规模数据中取得了较好的结果。相比早期经典的基于矩阵分解(matrix factorization, MF)的推荐算法,近几年流行的基于深度学习的方法,在稀疏的数据集中具有更好的泛化能力。但许多方法只考虑了二维的评分矩阵信息,或者简单的对各种属性做嵌入表征,而忽略了各种属性之间的内部关系。异构信息网络(heterogeneous information network, HIN)相比同构网络能够存储更加丰富的语义特征。近几年结合异构信息网络与深度学习的推荐系统,通过元路径挖掘关键语义信息的方法成为研究热点。

为了更好地挖掘各种辅助信息与用户喜好的关联性,本文结合张量分解、异构信息网络与深度学习方法,提出了新的模型hin-dcf。首先,基于数据集构建特定场景的异构信息网络;对于某一元路径,根据异构图中的路径信息生成其关联性矩阵。其次,合并不同元路径的关联性矩阵后,得到包含用户、物品、元路径三个维度的张量。接着,通过经典的张量分解算法,将用户、物品、元路径映射到相同维度的隐语义向量空间中。并且将分解得到的隐语义向量作为深度神经网络的输入层的初始化。考虑到不同用户对不同元路径的关联性偏好不同,融入注意力机制,学习不同用户、物品,与不同元路径的偏好权重。在实验部分,该模型在精确度上有效提升,并且更好地应对了数据稀疏的问题。最后提出了未来可能的研究方向。

关键词 异构信息网络; CP 张量分解; 深度学习; 推荐系统; 注意力机制; 可解释推荐

中图分类号 TP181 DOI号 10.19363/J.cnki.cn10-1380/tn.2021.09.06

Deep learning recommender system based on tensor decomposition of meta-paths of heterogeneous information networks

XU Ronghai, WANG Changdong

Department of Computer Science and Engineering, Sun Yat-Sen University, Guangzhou 510000, China

Abstract The existing representation learning based recommendation methods have achieved satisfactory results in large-scale data. Compare to the early classical matrix decomposition methods (matrix factorization, MF), the popular deep learning based methods have better generalization capability in sparse data in recent years. However, many methods only consider the two-dimensional scoring matrix information or simply do embedding vectors for various attributes, while ignoring the internal relationships between various attributes. Heterogeneous information networks (HIN) can store richer semantic features than homogeneous networks. In recent years, by combining heterogeneous information networks with deep learning for recommender systems, it becomes a hot research topic to mine key semantic information through meta-paths.

Aiming at mining the relevance between the various auxiliary information and users' preferences, by combining tensor decomposition methods, path information of heterogeneous information networks and deep learning methods, this paper propose a new model called hin-dcf. Firstly, a heterogeneous information network of a specific scene is constructed based on dataset. And a correlation matrix for certain meta-path is generated according to the path information of heterogeneous network. Secondly, after merging the correlation matrices of different meta-paths, a tensor is obtained which contains three dimensions of user, item and meta-path. Then, by the classical tensor decomposition methods, the user, item, and meta-path are mapped into the same dimensional hidden semantic vector space. The hidden semantic vectors obtained by tensor decomposition are used as the initialization of the embedding layer of the deep neural network. Considering the different relevance preferences of different users for different meta-paths, an attention mechanism is incorporated to learn the preference weights of different users, item pairs, for different meta-paths. Experiments show that the hin-dcf model has achieved improvements in terms of both accuracy and convergence speed, and better copes with the sparse data problem. Finally, the possible research directions in the future are proposed.

通讯作者: 王昌栋, 博士, 副教授, Email: changdongwang@hotmail.com

本课题得国家自然科学基金(No. 61876193)资助。

收稿日期: 2021-04-30; 修改日期: 2021-08-05; 定稿日期: 2021-08-10

Key words heterogeneous information networks; CP tensor decomposition; deep learning; recommendation system; attention mechanism; explainable recommendation

1 引言

1.1 选题背景及意义

互联网爆炸式增长的今天, 在电商邻域^[1], 淘宝、京东等电商平台拥有数以亿计的商品数目, 每日上千万的用户级流量。在社交网络领域, 新浪微博坐拥千万级日活量, 知乎作为新兴的知识型社区总回答数也过亿。这些新型的平台极大地改变了传统的信息传递方式^[2], 使得信息能够即时共享, 用户与他人与世界能够更加高效地交互、作用、影响。

然而, 面对海量的信息, 一方面用户逐渐提高了对信息准确性和有效性的要求, 一方面却日益难以获取到自己真正需要的信息。在信息爆炸的时代, 如何从庞大的数据中获取不同用户所需的有效信息成了一个重要的问题。而个性化推荐算法可以有效地缓解此类问题^[3]。推荐算法本质在做的是挖掘不同用户的不同喜好, 基于用户的偏好来推荐信息。这个信息可以是某一商品信息, 也可以是某一条微博包含的文本信息, 也可以是好友推荐的用户信息。推荐算法过滤掉大部分无效信息, 留下少量用户可能感兴趣的内容, 在这一意义上, 推荐算法在应用层面改变了信息的传递方式。

然而, 基于过去的方法, 我们仍然面临着许多挑战^[4]。如常见的由数据稀疏带来的冷启动问题^[5]。商品的信息是庞大的, 但新用户却缺乏历史购物数据, 使得推荐的误差极大。为了应对这一问题, 许多方法加入了辅助信息进行缓解。新用户虽然缺乏购物信息, 通过考虑用户年龄段、性别、职业、兴趣、好友等辅助信息能较好地应对冷启动问题。对这类辅助信息, 异构信息网络^[6]相比同构网络能够存储更加丰富的语义特征。

此外, 深度学习在推荐系统取得了不错的成果^[7]。基于神经网络的深度学习模型一般是深层非线性结构的。能够自动地学习不同的多维的异构数据, 深度地挖掘其中的特征。从而缓解了数据稀疏的问题。基于此, 本文将结合异构信息网络和深度学习的方法, 提出新的推荐模型, 以此缓解数据稀疏问题。

1.2 国内外研究现状和相关工作

协同过滤算法^[8], 有分为基于用户^[9]的协同过滤和基于物品^[10]的协同过滤方法。其主要思想是基于邻域考虑用户对商品偏好的关联性。基于物品的方

法(item-based)考虑相似度高的物品做推荐, 基于用户的方法(user-based)考虑相似度高的用户的物品做推荐。这一类方法由于只考虑了领域间的信息传递, 导致泛化能力较差。

另一种基于矩阵分解(matrix factorization, MF)^[11]的方法很好地解决了这一问题, 矩阵分解方法将用户和物品映射到同一维度的向量空间中。但矩阵分解只考虑了二维的用户评分信息, 在用户评分较少时, 无法很好地应对数据稀疏的问题。

近年来, 深度神经网络在各个领域都有了很成功的应用^[12]。在推荐系统领域中, He 等^[13]提出了 NCF 模型, 用 MLP 代替了 MF 中的点积操作, 同时融合了改进的矩阵分解方法, 增强了泛化能力。Deng 等提出的 DeepCF^[14], 将协同过滤算法分为表征学习过程和融合学习过程, 结合 MLP 提出的 CFnet, 提高了推荐的准确率。

在辅助信息方面, 异构信息网络(heterogeneous information network, HIN)^[15]是包含多种类型的节点和边的图。基于此产生了许多基于异构信息网络的推荐模型。这类模型的一个基本思想是基于元路径(meta-path)产生路径实例。通过路径本身包含的语义信息, 可以计算出用户、物品间相似性, 以此作推荐。Sun 等^[15], 提出新的基于路径的相似度计算方式(path-sim), 做 top-k 推荐。Yu 等^[16], 通过不同元路径, 建立用户和物品的不同关联性矩阵, 在结合传统的矩阵分解做推荐。Shi 等^[17], 通过不同路径提取出用户序列和物品序列, 通过序列训练生成用户和物品的表征, 在通过融合方程做预测。以上方法多致力于结合异构信息网络的辅助信息到推荐中。但依然存在挑战, 路径的语义如何与推荐关联, 在提高准确性的同时导致的可解释性丢失的问题, 以及如何利用异构信息网络充分地挖掘用户的偏好特征等问题。

在矩阵分解的基础上, 为了考虑更高维度的特征, 在推荐系统中也有结合张量分解的模型^[18]。例如加入标签信息^[19], 文本信息等等。但这类方法都是直接加入显性的辅助信息, 进行张量分解, 相比传统矩阵分解的提升有限。本文考虑结合异构信息网络具有丰富语义的优势, 通过元路径做信息的提取。相比常见的基于标签的张量分解方法更好地利用了辅助信息。

1.3 主要贡献

1) 提出基于张量分解方法(Tensor Decomposition)和异构信息网络(heterogeneous information network)的深度学习推荐算法。主要分为三个阶段: 首先根据数据集建立对应的异构信息网络, 并提取出元路径信息; 随后通过张量分解算法将用户、物品、元路径映射到同维度的隐语义向量空间中; 最终结合深度协同过滤算法, 将前一步得到的隐语义向量作为输入, 增强了模型的精确度。

2) 将张量分解得到的隐语义向量作为深度神经网络的输入层的初始化, 参考 DeepCF^[14] 的双塔模式, 一部分做表征学习、一部分做匹配学习, 通过多层感知机(MLP)增强泛化能力。考虑到不同用户对不同元路径的关联性偏好不同, 融入注意力机制(attention mechanism), 学习不同用户、物品, 与不同元路径的偏好权重。

3) 使用 movielens, douban movie 和 Amazon 数据集进行训练和测试, 与其他推荐算法做多组不同指标的对比试验, 本文提出的 hin-dcf 模型有了张量分解的嵌入向量初始化后, 模型的精确度和收敛速

度皆有效提升, 并且在数据稀疏情况下更好地应对了数据稀疏的问题。

2 预备知识

2.1 异构信息网络

定义一: 异构信息网络。异构信息网络^[6]是一个有向图 $G = (V, E)$, V 表示节点集合, E 为有向图边的集合。且每个结点有对应的节点类型, 即存在映射关系 $\phi: V \rightarrow A$ 。每条边有对应的边类型, 即映射关系 $\varphi: E \rightarrow R$ 。其中 A 和 R 分别表示结点类型集合和边类型集合, 且 $|A| + |R| > 2$ 。

例一: 图 1 列举了异构信息网络(Hin)的例子。该 Hin 为电影场景下的网络图, 包含五类节点类型, 分别是用户 user(u)、电影 movie(m)、导演 director(d)、小组 group(g)、电影类型 type(t)。每个节点表示某一结点类型的实例, 如图中 u_1 特指某一用户、 t_1 特指某一电影类型(如恐怖电影类型)。Hin 中的边的意义是直观的, 如边类型 u-g, 表示用户加入了该小组; 边类型 u-m, 表示用户看过该电影并给出评分。

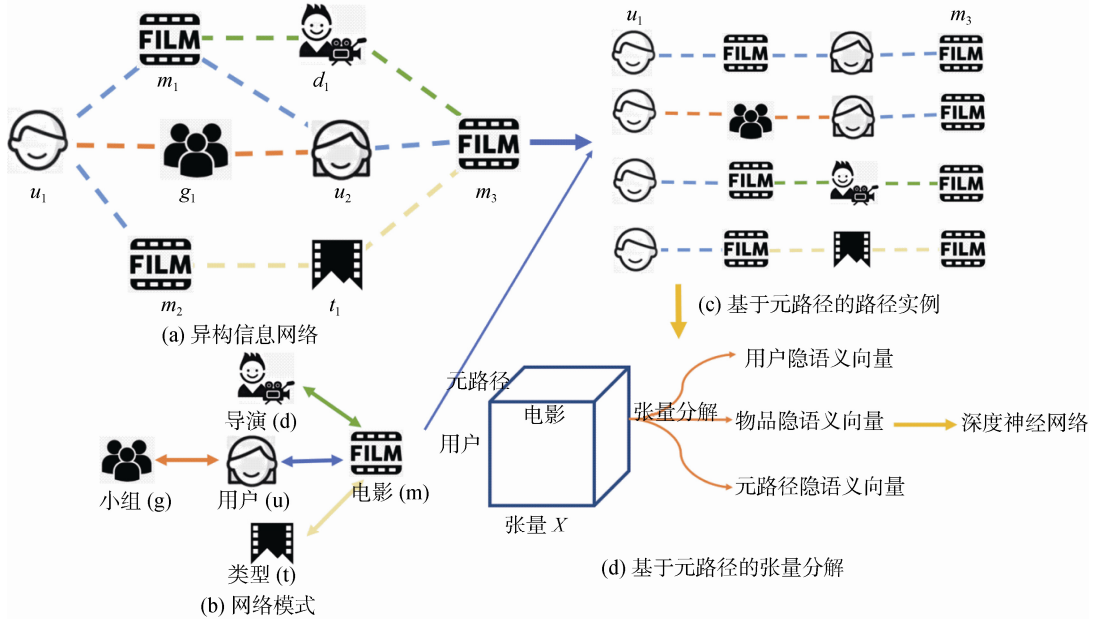


图 1 HIN 异构信息网络的实例以及基于元路径建立张量的思路

Figure 1 Example of heterogeneous information network and the idea of building a tensor based on meta-paths

定义二: 网络模式(network schema)。对给定的 $G = (V, E)$, 且有映射关系 $\phi: V \rightarrow A$ 和 $\varphi: E \rightarrow R$ 。则称 $S = (A, R)$ 为其网络模式。

例二: 图 1 为(a)的网络模式, 网络模式可以简单理解为包含所有节点类型和边类型的最小图结构。相反, 根据网络模式我们也可以生成一个新的异构信息网络。

定义三: 元路径(meta-path)。对给定的网络模式 $S = (A, R)$, 定义元路径 p 为 $a_1 \xrightarrow{r_1} a_2 \xrightarrow{r_2} \dots \xrightarrow{r_l} a_{l+1}$ 。

其中 a_k 表示元路径上第 k 个节点所对应的节点类型, r_k 表示第 k 条边所对应的边类型。

例三: 基于图 1(b)的网络模式, 我们能生成很多不同的元路径(meta-path)。对于推荐问题, 将元路径的长度限制在 4, 且头节点为用户类型, 尾节点为电

影类型。如 $UMUM$, $UMDM$, $UMTM$, $UGUM$ 。这些元路径包含了各自特定的语义, 例如 $UMUM$ 表示联系建立在看过共同电影的用户上。基于这些语义能够在推荐的可解释上发挥一定作用。以图 1(c) 为例子, 基于元路径(meta-path)在用户 u_1 和电影 m_3 间能建立多条路径关系。例如图中的 $u_1 - m_1 - u_2 - m_3$ 属于 $UMUM$, 还有 $u_1 - g_1 - u_2 - m_3$, $u_1 - m_2 - d_1 - m_3$, $u_1 - m_2 - t_1 - m_3$, 分别属于不同的元路径。

定义四: 基于异构信息网络的推荐(hin-based recommendation)。在推荐系统中, 给定一个异构信息网络 $G = (V, E)$, 我们最关心的实体是用户和物品类型的实体。三元组集合 $R = \{ \langle u, m, r_{u,m} \rangle \}$ 表示用户 u 曾经给物品 m 的评分为 $r_{u,m}$ 的集合。且有 $u \in V, m \in V, R \subset E$ 。给定异构信息网络 $G = (V, E)$, 目标是对用户 u 预测 $r_{u,m'}$, 其中 m' 是该用户未评分过的物品。

例四: 在图 1(a)中, 用户 u_1 和电影 m_1 以及电影 m_2 之间有 $U - M$ 的边类型, 即这两部电影被评分了。而用户 u_1 未曾给过电影 m_3 评分。推荐系统的目标为, 判断用户 u_1 和电影 m_3 建立连边的可能性。

2.2 CP 张量分解

张量分解的一种主流方法是 CP 分解^[20] (Canonical Polyadic Decomposition)。CP 分解已在信号处理、图像处理、降噪等方面有了广泛的应用。本文考虑采用三维的 CP 张量分解提取前文提到的基于元路径的张量矩阵中的用户隐藏的辅助信息和物品的隐藏的辅助信息。CP 分解的分解形似如下:

$$X = \sum_{i=1}^r a_i \otimes b_i \otimes c_i \quad (1)$$

$$X \in \mathbb{R}^{N \times M \times P}, a_i \in \mathbb{R}^{N \times 1 \times 1}, b_i \in \mathbb{R}^{1 \times M \times 1}, c_i \in \mathbb{R}^{1 \times 1 \times P}$$

即对 X 分解得到长度为 r 的隐语义向量(latent factors), 其中 a_i 、 b_i 、 c_i 分别表示 CP 分解得到的对应维度的向量。

2.3 基于张量分解提取隐语义

建立张量 $X \in \mathbb{R}^{N \times M \times P}$, 其中 $X_{i,j,k}$ 表示用户 u_i 与物品 m_j 之间在建立的异构信息网络中元路径 p_k 的数量。用户的数量为 N , 物品的数量为 M , 元路径的数量为 P 。

经过 R 阶 CP 分解后, 对每个用户 i , 物品 j 和元路径 k 都得到一个长度为 r 的隐语义向量, 分别为 u_i, m_j, p_k 。如图 1(d) 所示。且有:

$$X_{i,j,k} = \sum_{r=1}^R u_{i,r} \times m_{j,r} \times p_{k,r} \quad (2)$$

$$u_i \in \mathbb{R}^{1 \times R}, m_j \in \mathbb{R}^{1 \times R}, p_k \in \mathbb{R}^{1 \times R}$$

其中 $u_{i,r}$ 表示向量 u_i 的第 r 个位置的数值。 $m_{j,r}$ 与

$p_{k,r}$ 同理。

2.4 张量分解损失函数

R 阶张量分解的损失函数为实际张量的值与近似张量值得平方误差。为了防止过拟合现象, 加入正则化项。损失函数如下:

$$\text{Loss} = \min \sum_{ijk} (X_{i,j,k} - \sum_{r=1}^R u_{i,r} \times m_{j,r} \times p_{k,r})^2 + \lambda (\|u_i\|_F + \|m_j\|_F + \|p_k\|_F) \quad (3)$$

其中, λ 表示正则化项的系数, $\|\cdots\|_F$ 表示 F 范数。

本文将会把这三种基于线性关系的隐语义向量作为深度学习网络的输入, 以此捕获更多的信息。

表 1 主要的符号及其意义

Table 1	The main symbols and their meanings
$\mathcal{U}, \mathcal{M}, \mathcal{P}$	用户集合, 物品集合, 元路径集合
N, M, P	用户数量, 物品数量, 元路径数量
$X \in \mathbb{R}^{N \times M \times P}$	基于异构信息网络的元路径建立的张量
$X_{i,j,k}$	用户 i 与物品 j 之间在建立的异构信息网络中元路径 k 的数量的索引
R	表示 CP 张量分解的阶数为 R
$u_i \in \mathbb{R}^{1 \times R}$	基于张量分解得到的用户 i 的隐语义向量的索引
$m_j \in \mathbb{R}^{1 \times R}$	基于张量分解得到的物品 j 的隐语义向量的索引
$p_k \in \mathbb{R}^{1 \times R}$	基于张量分解得到的元路径 k 的隐语义向量的索引
p^k	表示元路径 k , 例如 $UM, UMUM, UGUM$ 。
Y^{p^k}	表示基于元路径 p^k 建立的关联性矩阵
\otimes	张量积
$y \in \mathbb{R}^{U \times M}$	用户对物品的评分矩阵
$y_{i,j}$	用户 i 对物品 j 的评分的索引
$\hat{y}_{i,j}$	用户 i 对物品 j 的预测评分的索引

3 提出的模型

在这一部分, 我们将提出基于异构信息网络中元路径作张量分解的深度学习推荐算法 hin-dcf。主要分为两部分: 首先基于 hin 元路径信息做张量分解, 生成用户、物品、元路径的隐语义向量作为 embedding。其次, 将该 embedding 作为输入, 结合当前的流行的深度学习推荐模型, 采用双塔结构, 兼顾表征学习和匹配学习^[19]。同时, 结合 MLP 与基于元路径的注意力机制, 提高了模型的预测准确率。

3.1 基于元路径建立张量

3.1.1 基于元路径建立用户-物品的关联性矩阵

上文中已经提到元路径以及路径的定义和相关例子, 接下来, 将基于此建立能够表达用户和物品间关联度的矩阵 Y 。参考图 1, 对特定的元路径, 如 $UMUM$, 寻找用户 u_1 到物品 m_3 间所有的属于该元路

径的路径。例如 $u_1 - m_1 - u_2 - m_3$, 显然这样的路径, 表达了用户 u_1 和物品 m_3 的一种正向关联性, 因为购买相同物品的人更可能有相同的偏好, 并且这种关联是显示语义的, 具有直接的可解释性。直观地, 这样的路径数目越多, 物品 m_3 就具有更高的可能性满足用户 u_1 的喜好, 同时该物品也更可能与用户 u_1 购买过的历史物品具有更高的相似性。此外, 路径反应用户和物品的关系程度也与元路径本身直接相关。因此, 对于元路径 p^k , 可以建立用户与物品的关联性矩阵 Y^{p^k} 。

特别地, 对于长度为 1 的元路径, 如 UG , 当 $u_i - g_j \in E$ 时 $Y_{i,j}^{UG} = 1$, 否则为 0。其中 $Y_{i,j}^{UG}$ 表示矩阵 Y^{p^k} 第 i 行、第 j 列的取值。

3.1.2 关联性矩阵的递归生成

如上文, 对长度为 1 的元路径, 生产关联矩阵的方法是简单的。参考图 2, 对于长度大于 1 的情况, 可以基于更短的元路径的关联性矩阵递归的生成。例如对元路径 $UGUM$, 可以通过如下矩阵乘法递归生成:

$$\begin{aligned} Y^{UGU} &= Y^{UG} \times (Y^{UG})^T \\ Y^{UGUM} &= Y^{UGU} \times Y^{UM} \end{aligned} \quad (4)$$

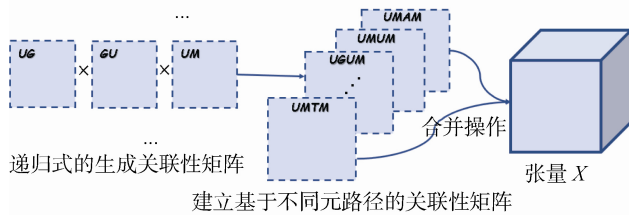


图 2 基于元路径建立关联性矩阵及张量

Figure 2 Establishing correlation matrix and tensor based on meta-path

3.1.3 建立张量

参考图 2, 本文只考虑用户和物品间的关系, 因此只选取的头节点为用户尾节点为物品的元路径, 即 $U \cdots M$ 。选取 P 条元路径, 分别为 p^1, p^2, \dots, p^P 。基于此可以建立 P 个不同的关联性矩阵, 考虑到不同元路径生成的关联性矩阵数值范围差别很大, 可能导致接下来的张量分解中有效信息的损失, 因此在建立张量前, 对每个关联性矩阵进行标准化处理。参考图 2, 最后将这 P 个关联性矩阵进行合并得到张量 X :

$$X = \begin{bmatrix} Y^{p^1} \\ \vdots \\ Y^{p^P} \end{bmatrix} \quad (5)$$

3.2 嵌入层(embedding)的初始化

将张量分解得到用户、物品和元路径的隐语义向量作为 hin-dcf 模型的 embedding 层。基于 CP 分

解后, 这些表征向量在同一向量空间中, 并且是线性关系的, 可以直接通过点积反映其关联程度。同时, 更为重要的是, 由于张量是基于元路径建立的, 分解得到的表征向量本质上反映的是用户和物品间的路径的关系, 并不能直接用于推荐问题里的评分预测。另一方面它又包含了许多 hin 的重要信息以及用户的评分偏好信息。基于此, 本文直接将其作为 hin-dcf 模型的 embedding 的初始化, 通过后续的表征学习过程和匹配学习过程, 使其更加适应于推荐问题。

3.3 基于元路径的注意力机制(attention mechanism)

不同用户有不同的喜好模式, 例如某一用户可能会更关注相同兴趣小组成员看过的电影, 而其他用户可能更偏向于看同一导演的系列作品, 或者偏爱于喜剧类型电影。而这样的喜好模式恰好是元路径所表达的, 基于不同的用户有不同的喜好模式的假设, 本文提出新的基于元路径的注意力机制, 以此挖掘出不同的偏好模式信息。

用户、物品、元路径张量 X 经过 CP 分解后分别得到用户、物品、元路径的隐语义向量 u_i, m_j, p_k 。参考图 3 的注意力机制部分, 考虑不同的用户、物品对, 存在更倾向于某一喜好模式(元路径)的情况。本文通过不同的(用户, 物品, 元路径)三元组, 学习不同元路径的权重。采用两层的网络结构做注意力机制:

$$\begin{aligned} v_{i,j,k}^{(0)} &= [u_i, m_j, p_k] \\ v_{i,j,k}^{(1)} &= a(W_1^T v_{i,j,k}^{(0)} + b_1) \\ v_{i,j,k}^{(2)} &= a(W_2^T v_{i,j,k}^{(1)} + b_2) \end{aligned} \quad (6)$$

其中, W_1 和 W_2 为权重矩阵, b_1 和 b_2 为偏移系数。 $a(\cdot)$ 为激活函数。 $v_{i,j,k}^{(0)}$ 表示合并向量。 $v_{i,j,k}^{(2)}$ 表示不同元路径的权重系数向量。

最终, 将得到的注意力权重系数通过 softmax 函数进行归一化。得到的权重可以理解为, 不同元路径与同一用户、物品对的关联程度。

$$w_{i,j,k} = \text{softmax}(v_{i,j,k}^{(2)}) = \frac{\exp v_{i,j,k}^{(2)}}{\sum_{k=1}^P \exp v_{i,j,k}^{(2)}} \quad (7)$$

其中, $w_{i,j,k}$ 表示元路径 k 对于用户 i 、物品 j 的权重系数。

通过注意力权重系数, 可以聚合得到总的元路径 embedding 向量 p' 。通过简单地加权求和得到:

$$\begin{aligned} w_{i,j} &= [w_{i,j,1}, \dots, w_{i,j,P}] \\ p' &= \phi_0(w_{i,j}, \bigcup_{k=1}^P p_k) = w_{i,j} \times \begin{bmatrix} p_{1;} \\ \vdots \\ p_{P;} \end{bmatrix} \end{aligned} \quad (8)$$

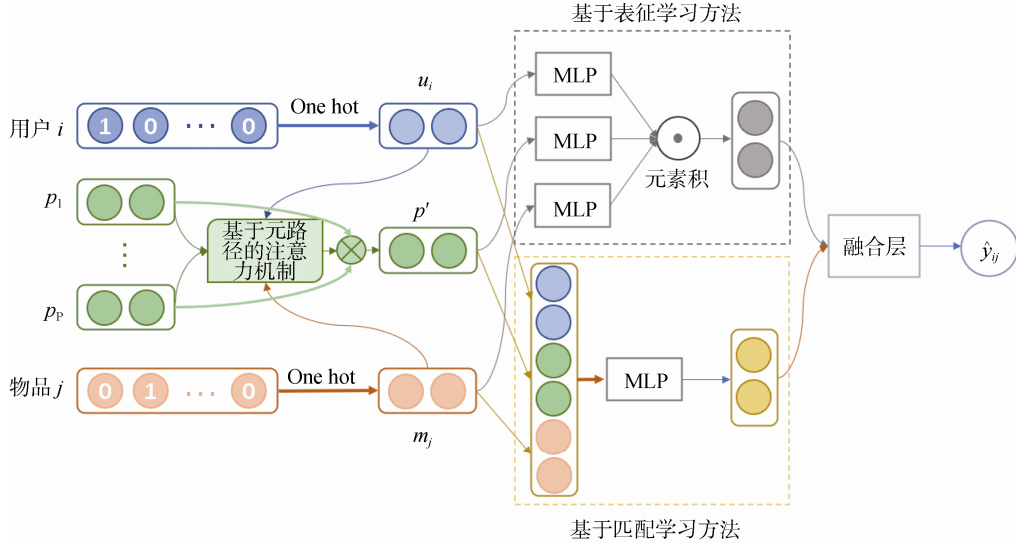


图3 基于异构信息网络元路径注意力机制的深度学习推荐系统 hin-dcf

Figure 3 Deep learning recommendation system based on meta-path attention mechanism of heterogeneous information network hin-dcf

3.4 基于表征学习方法

参考图3, 本文提出了结合元路径的表征学习过程 (presentation learning)。对用户和物品, 采用 one-hot 向量输入进预先初始化的 embedding 层, 得到对应的表征向量。对于元路径, 通过基于元路径的注意力机制, 我们学得元路径的融合向量。在本文, 我们采用多层感知机^[21](MLP)作为表征学习函数, 学习用户、物品、元路径的非线性信息。因此, 表征学习过程的定义如下:

$$\begin{aligned} \mathbf{a}_0 &= \mathbf{u}_i \\ \mathbf{a}_1 &= a(\mathbf{W}_1^T \mathbf{a}_0 + \mathbf{b}_1) \\ &\vdots \end{aligned} \quad (9)$$

$$\mathbf{u}_i^* = \mathbf{a}_X = a(\mathbf{W}_X^T \mathbf{a}_{X-1} + \mathbf{b}_X)$$

其中 \mathbf{W}_X 为权重矩阵, \mathbf{b}_X 为偏移系数, \mathbf{a}_X 为输出向量; 其中 k 表示第 k 层神经网络。 $a(\cdot)$ 为激活函数, 采用 $\text{ReLU}(\cdot)$ 激活函数。最终输出的 \mathbf{u}_i^* 为学习得到的用户的表征向量。

用同样的表征学习方法, 可以得到 m_j 和 p_k 的表征向量 m_j^* , p_k^* :

$$\hat{y}_{i,j} = \sigma(\mathbf{W}_{\text{out}}^T (\mathbf{u}_i^* \odot \mathbf{m}_j^* \odot \mathbf{p}_k^*)) \quad (10)$$

其中, \mathbf{W}_{out} 为权重矩阵。 \odot 表示向量的元素积 (element wise product)。相比传统的点积操作或者余弦相似度, 元素积保留了向量对应维元素的交互信息, 能做到非线性地融合变换。其中 $\sigma(\cdot)$ 为 $\text{Sigmoid}(\cdot)$ 激活函数。 $\hat{y}_{i,j}$ 为最终输出的用户 i 对物品 j 的预测评分。

3.5 基于匹配学习方法

参考图2, 本文提出了结合元路径的匹配学习过程 (interaction learning)。基于匹配学习的方法更侧重

学习用户、物品、元路径的匹配融合过程。本文先合并用户 i 、物品 j 的张量分解得到的嵌入表征向量, 以及基于注意力机制得到的元路径的融合向量。通过多层感知机 (MLP) 学习三者间的匹配过程。具体过程如下:

$$\begin{aligned} \mathbf{a}_0 &= \phi_1(\mathbf{u}_i, \mathbf{m}_j, \mathbf{p}') = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{m}_j \\ \mathbf{p}' \end{bmatrix} \\ \mathbf{a}_1 &= a(\mathbf{W}_1^T \mathbf{a}_0 + \mathbf{b}_1) \\ &\vdots \\ \mathbf{a}_Y &= a(\mathbf{W}_Y^T \mathbf{a}_{Y-1} + \mathbf{b}_Y) \\ \hat{y}_{i,j} &= \sigma(\mathbf{W}_{\text{out}}^T \mathbf{a}_Y) \end{aligned} \quad (11)$$

其中 ϕ_1 表示聚合函数, 本文采用最简单的合并方式, 但能够最大限度地保留原始的信息。其中 \mathbf{W}_Y 和 \mathbf{W}_{out} 为权重矩阵, \mathbf{b}_Y 为偏移系数, k 表示第 k 层神经网络。 $a(\cdot)$ 为激活函数, 采用 $\text{ReLU}(\cdot)$ 激活函数。 $\hat{y}_{i,j}$ 为最终输出的用户 i 对物品 j 的预测评分。

3.6 结合表征学习与匹配学习

表征学习更好地学习用户的表征向量, 而匹配学习更深刻地挖掘了用户、物品、元路径三元组间的融合方式, 将结合这两种方法, 使得模型同时拥有这两个学习过程的优势。

$$\hat{y}_{i,j} = \sigma(\mathbf{W}_{\text{out}}^T \begin{bmatrix} \mathbf{a}_X^{r,l} \\ \mathbf{a}_Y^{i,l} \end{bmatrix}) \quad (12)$$

其中 \mathbf{W}_{out} 为权重矩阵, $\mathbf{a}_X^{r,l}$ 为表征学习方法中用户、物品、元路径三者间元素积得到的向量, $\mathbf{a}_Y^{i,l}$ 为匹配学习方法中倒数第二层得到的向量, 即 \mathbf{a}_Y 。通过全连接层和激活函数 $\sigma(\cdot)$ 最终学习用户 i 对物品 j 的预测评分。

表 2 实验数据集

Table 2 Datasets

数据集	关系 (A-B)	#A	#B	#A-B
Movielens	User-Movie	6040	3952	1000209
	User-Age	6040	7	6040
	User-Occupation	6040	21	6040
	Movie-Genre	3952	18	6408
Douban Movie	User-Movie	13367	12677	1068278
	User-Group	13367	2753	570047
	Movie-Actor	12677	6311	33587
	Movie-Director	12677	2449	11276
	Movie-Type	12677	38	27668
Amazon	User-Item	6170	2753	195791
	Item-View	2753	3857	5694
	Item-Category	2753	22	5508
	Item-Brand	2753	334	2753

3.7 目标函数

在点对点(point-wise)的推荐模型中, 以及显示反馈信息的情况下, 假设评分服从正态分布, 可以采用经典的均方误差(MSE)作为损失函数:

$$\text{Loss}_{MSE} = \frac{1}{|D|} \sum_{(i,j) \in D} (y_{i,j} - \hat{y}_{i,j})^2 \quad (13)$$

在隐式反馈的情况下, 由于没有显示的评分信息, 只有隐式的反馈, 例如是否有点击等。这样的反馈服从 0-1 的二项分布。更适合采用交叉熵作为损失函数:

$$\text{Loss}_{BCE} = - \sum_{(i,j) \in D} y_{i,j} \log \hat{y}_{i,j} + (1 - y_{i,j}) \log(1 - \hat{y}_{i,j}) \quad (14)$$

4 实验部分

4.1 实验设置

4.1.1 实验数据集

本文采用了几个常用的测试数据集^①。其中 Movielens 为电影数据集, 包含了用户对电影的评分、用户年龄、职业等相关信息。还有关于豆瓣的数据集, 豆瓣是国内最大的评分网站之一, 因此具有较高的实际意义, 相比 Movielens 数据集, 该数据集包含了更有意义的群组信息, 加入相同小组的用户具有相同的兴趣。亚马逊(Amazon)数据集, 包含商品的品牌、类型等信息。关于实验数据集具体的细节可以查看表 2。

“#A”和“#B”一列表示分别表示 A 类型和 B 类型节点的数量。“#A-B”表示 AB 间交互的数量。例如, 对于每个数据集的第一行, 分别表示了用户

数量、物品数量和用户物品间交互数量。由表 3, 展示了在不同数据集中选取的元路径。考虑到计算的复杂度, 本文只选取了长度为三的元路径。因为元路径越长, 所表达的用户和物品间的关联性越弱, 也就包含了更多的无用的噪音语义信息。为了避免这一问题, 采用限制元路径的长度的方法。

表 3 元路径

Table 3 Meta-path

数据集	元路径
Movielens	UMUM, UMGU, UAUM, UOUM
Douban-movies	UMUM, UMAM, UMDM, UMTM, UGUM
Amazon	UIUI, UIVI, UICI, UIBI

以 Movielens 数据集为例, A 表示不同年龄段, UAUM 表示反映了相同年龄段电影喜好的元路径。O 表示职业类型, UOUM 表示反映了相同职业类型电影喜好的元路径。

4.1.2 评估方法

在本文的实验部分中, 由于数据集为有评分数据, 所以属于显式反馈记录。本文采用常见的均方根误差(RMSE)和平均绝对误差(MAE)来评价模型误差。其数值越小, 表示模型越精确, 预测效果越好。具体公式如下:

$$\text{RMSE} = \sqrt{\frac{1}{|D|} \sum_{(i,j) \in D} (y_{i,j} - \hat{y}_{i,j}')^2} \quad (15)$$

$$\text{MAE} = \frac{1}{|D|} \sum_{(i,j) \in D} |y_{i,j} - \hat{y}_{i,j}| \quad (16)$$

其中, D 表示测试集数据。

① <https://github.com/librahu/HIN-Datasets-for-Recommendation-and-Network-Embedding>

4.1.3 对比方法

NeuCF: ncf^[13]是第一个将神经网络应用到协同过滤上的方法。模型分为广义的矩阵分解方法和基于 MLP 的模块。

DCF: dcf^[14]是在 2019 年, 提出的基于协同过滤分为表征学习过程和匹配学习过程的方法。也是本文所学习的思路。

DMF: dmf^[22]将传统的矩阵分解方法融合神经网络, 大大提升了模型的泛化能力。

Hin-dcf: hin-dcf 是本文提出的模型。

4.1.4 实现细节

CP 张量分解部分采用 python 的 tensorly 库^[23]来实现。hin-dcf 模型基于 python 的 torch^[24]深度学习库

实现的。具体地, 模型参数的初始化采用正态分布, 优化部分采用 Adam^[25](Adaptive Moment Estimation) 方法。而训练的 batch 大小设置为 256。学习率设置为 0.001。

为了保证训练的公平性, 生成元路径的数据只考虑了训练集, 元路径的数量和类型见表 3。其中大写字母对应表 2 中关系一列的首字母。

4.1.5 实验结果

实验的对比结果参考表 4。主要的结论如下:

在三个数据集中, 实验结果大部分都是 hin-dcf 最佳, 除了 movielens 80%略差于 dmf 模型。这一结果证明了在加入异构信息网络中元路径的 hin-dcf 模型在推荐问题上准确度有了提升。

表 4 实验结果

Table 4 Experimental result

数据集	训练比例(%)	dcf		ncf		dmf		hin-dcf	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Movielens	20	0.774	0.976	0.744	0.941	0.771	0.969	0.728	0.922
	40	0.750	0.948	0.727	0.92	0.744	0.939	0.714	0.906
	60	0.725	0.919	0.723	0.913	0.717	0.907	0.703	0.894
	80	0.702	0.894	0.706	0.901	0.691	0.875	0.698	0.887
Amazon	20	0.824	1.092	0.825	1.092	0.842	1.085	0.729	1.007
	40	0.779	1.051	0.755	1.023	0.79	1.024	0.701	0.971
	60	0.742	1.018	0.717	0.995	0.75	0.978	0.681	0.949
	80	0.711	0.988	0.706	0.974	0.727	0.955	0.679	0.936
Douban-movies	20	0.658	0.824	0.598	0.756	0.654	0.818	0.567	0.722
	40	0.628	0.787	0.575	0.728	0.618	0.774	0.554	0.708
	60	0.600	0.755	0.566	0.717	0.589	0.739	0.549	0.700
	80	0.579	0.731	0.559	0.709	0.563	0.708	0.546	0.697

本次实验将对比了不同训练比例(training rate 即训练集占总评分数据集的比例)下, 模型的准确率。显然的, 在降低训练集的同时, 模型的准确率会下降。参考图 5, 在 movielens 和 Amazon 这两个数据集上, 随着训练比例的下降, hin-dcf 的准确率下降是最平缓的, 说明在数据集越稀疏的情况下(较低的训练比例下), 模型发挥的作用是相对稳定且有效的。

关于可解释的实例分析: 为了直观地演示 hin-dcf 基于元路径注意力机制的可解释方法。参考图 4, 本文随机从 douban-movies 测试集中选取用户电影做分析, 训练比例为 80%。图 4 中的条形统计图展现了用户 10572 与电影 437 和电影 4913 分别的注意力权重。对电影 437, 注意力权重最高的是 UMUM 元路径。表明这次推荐主要是基于看过相同电影的

用户来作推荐。对电影 4913, 注意力权重最高的是 UGUM, 表示这次推荐主要基于共同兴趣小组作推荐。我们可以看出, hin-dcf 模型可以对推荐的结果基于元路径做一定的可解释。

参考图 6, 在不改变模型结构的条件下, 有无隐语义向量作为初始化的模型在预测误差(MAE)上的表现差异是明显的。原模型 hin-dcf 随着训练比例地增大, 准确率同时稳定地增大。而采用随机初始化表征向量的对比模型, 准确率下降很大, 并且在四个不同训练比例下的表现也波动较大。其中在训练比例为 40%时, 随机模型的 MAE 为 1.273 相比 hin-dcf 的 MAE 为 0.554 高了 129.8%。在没有张量分解得到的隐语义向量作为初始化的模型表现是糟糕的, 说明了嵌入层的初始化过程在 hin-dcf 中的有效性。

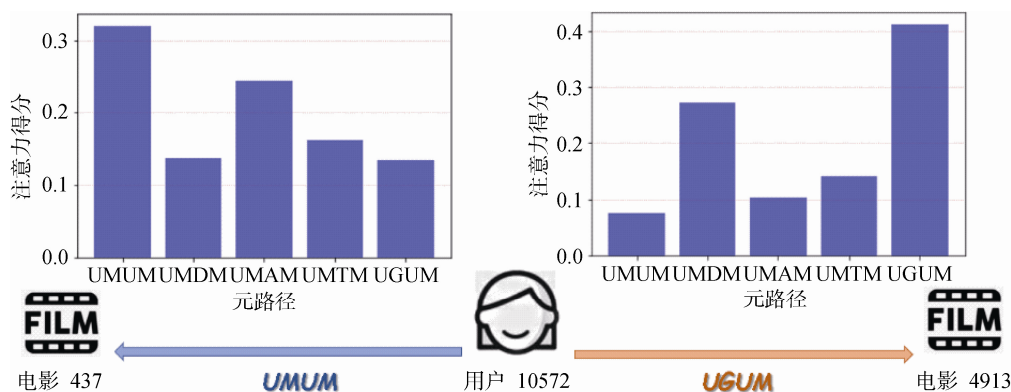


图 4 基于元路径注意力机制做可解释的例子

Figure 4 An interpretable example based on the meta-path attention mechanism

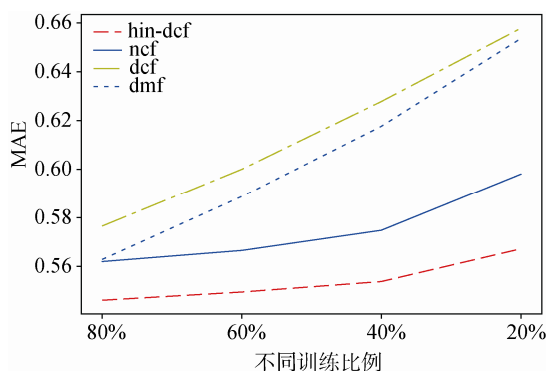


图 5 不同训练比例下 MAE 的变化

Figure 5 MAE in different training ratios

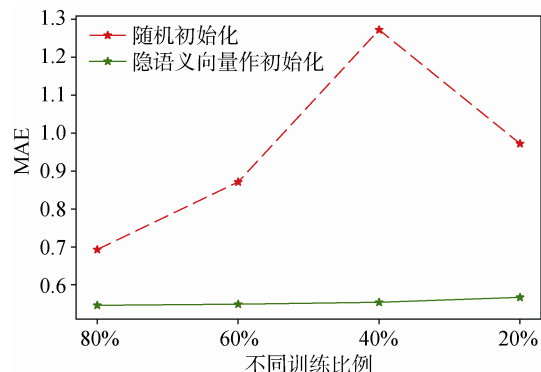


图 6 有无隐语义向量初始化模型的 MAE 误差的对比图

Figure 6 MAE comparison of models with or without latent factor initialization in different training ratios

为了验证结合表征学习方法和匹配学习方法能够提升模型性能, 参考图 7, 本文将 hin-dcf 与单独的表征学习模型和匹配学习模型在不同数据集上的表现进行对比。hin-dcf 在所有数据集上都表现最佳。其中在 movielens 数据集中, 匹配学习模型的 MAE 比 hin-dcf 提升了 2.3%。

参考图 8, 本文考虑了是否加入注意力机制对模

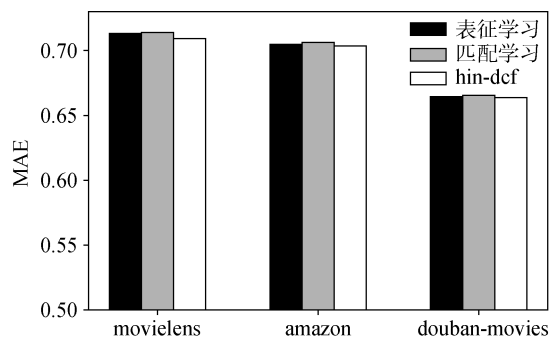


图 7 表征学习模型和匹配学习模型和 hin-dcf 在不同数据集上的 MAE 对比图

Figure 7 MAE comparison of representation learning model and interaction learning model and hin-dcf in different datasets

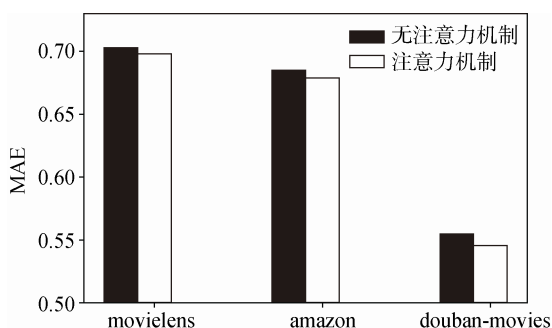


图 8 有无注意力机制模型在不同数据集上的 MAE 对比图

Figure 8 Comparison of models with or without attention mechanism on different datasets

型影响的对比实验。其中, 为了控制变量, 无注意力机制模型对不同的元路径的隐语义向量采用相同权重的加权聚合。而加入注意力机制的模型则通过注意力机制学习不同元路径的权重。在不同数据集中, 加入注意力机制后模型的准确率皆有了提升。

4.2 结论

本文研究了一种结合张量分解方法和异构信息网络中路径信息的深度学习推荐模型。将用户、物品、元路径映射到相同维度的隐语义向量空间中的。且将分解得到的隐语义向量作为深度神经网络的输入的初始化, 采用双塔的学习模式, 一部分做表征学习、一部分做匹配学习。考虑到不同用户有不同的喜好模式, 即对不同元路径的偏好不同, 本文融入基于元路径的注意力机制, 学习不同元路径的偏好权重。同时基于元路径的注意力权重可以在推荐过程中做一定的可解释。在实验部分, 与其他流行的推荐算法做多组不同指标的对比试验, **hin-dcf** 模型的精确度和收敛速度皆有效提升, 并且能够更好地应对数据集稀疏的问题。

本文讨论的方法结合了异构信息网络, 是在网络信息不缺失的情况下的研究。然而现实中信息的很难完整地获取, 需要进一步研究在异构信息网络缺失部分信息的情况下, 如何更好地应对数据集稀疏的问题。

参考文献

- [1] Ji B W. Analysis of Development Trend of E-Commerce Industry In China[J]. *Refining and Chemical Industry*, 2018, 29(1): 65-66. (纪博文. 中国电子商务行业的发展趋势分析[J]. *炼油与化工*, 2018, 29(1): 65-66.)
- [2] Xia Y H. The Structure and Mechanism of Micro-Blog Interaction: An Empirical Study on Sina Micro-Blog[J]. *Journalism & Communication*, 2010, 18(4): 60-69, 110. (夏雨禾. 微博互动的结构与机制——基于对新浪微博的实证研究[J]. *新闻与传播研究*, 2010, 18(4): 60-69, 110.)
- [3] Resnick P, Varian H R. Recommender Systems[J]. *Communications of the ACM*, 1997, 40(3): 56-58.
- [4] Sharma L, Gera A. A Survey of Recommendation System: Research Challenges[J]. *International Journal of Engineering Trends and Technology*, 2013, 4(5): 1989-1992.
- [5] Schein A I, Popescul A, Ungar L H, et al. Methods and Metrics for Cold-Start Recommendations[C]. *The 25th annual international ACM SIGIR conference on Research and development in information retrieval*, 2002: 253-260.
- [6] Shi C, Li Y T, Zhang J W, et al. A Survey of Heterogeneous Information Network Analysis[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2017, 29(1): 17-37.
- [7] Liu J T, Wu C H. Deep Learning Based Recommendation: A Survey[C]. *Information Science and Applications 2017*, 2017: 451 - 458.
- [8] Su X Y, Khoshgoftaar T M. A Survey of Collaborative Filtering Techniques[J]. *Advances in Artificial Intelligence*, 2009, 2009: 1-19.
- [9] Liu H F, Hu Z, Mian A, et al. A New User Similarity Model to Improve the Accuracy of Collaborative Filtering[J]. *Knowledge-Based Systems*, 2014, 56: 156-166.
- [10] Sarwar B, Karypis G, Konstan J, et al. Item-Based Collaborative Filtering Recommendation Algorithms[C]. *The 10th international conference on World Wide Web*, 2001: 285-295.
- [11] Koren Y, Bell R, Volinsky C. Matrix Factorization Techniques for Recommender Systems[J]. *Computer*, 2009, 42(8): 30-37.
- [12] Pouyanfar S, Sadiq S, Yan Y L, et al. A Survey on Deep Learning[J]. *ACM Computing Surveys*, 2019, 51(5): 1-36.
- [13] He X N, Liao L Z, Zhang H W, et al. Neural Collaborative Filtering[C]. *The 26th International Conference on World Wide Web*, 2017: 173-182.
- [14] Deng Z H, Huang L, Wang C D, et al. DeepCF: A Unified Framework of Representation Learning and Matching Function Learning In Recommender System[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, 33: 61-68.
- [15] Sun Y Z, Han J W, Yan X F, et al. Pathsim: Meta Path-Based Top-k Similarity Search In Heterogeneous Information Networks[J]. *Proceedings of the VLDB Endowment*, 2011, 4(11): 992-1003.
- [16] Yu X, Ren X, Sun Y Z, et al. Personalized Entity Recommendation: A Heterogeneous Information Network Approach[C]. *The 7th ACM international conference on Web search and data mining*, 2014: 283-292.
- [17] Hu B B, Shi C, Zhao W X, et al. Leveraging Meta-Path Based Context for Top- N Recommendation with a Neural Co-Attention Model[C]. *The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018: 1531-1540.
- [18] Symeonidis P. Matrix and Tensor Decomposition In Recommender Systems[C]. *The 10th ACM Conference on Recommender Systems*, 2016: 429-430.
- [19] Cai Y Z, Zhang M, Luo D J, et al. Low-Order Tensor Decompositions for Social Tagging Recommendation[C]. *The fourth ACM international conference on Web search and data mining*, 2011: 695-704.
- [20] Hitchcock F L. The Expression of a Tensor or a Polyadic As a Sum of Products[J]. *Journal of Mathematics and Physics*, 1927, 6(14): 164-189.
- [21] Noriega L. Multilayer Perceptron Tutorial[J]. *School of Computing Staffordshire University*, 2005: 1-12.
- [22] Xue H J, Dai X Y, Zhang J B, et al. Deep Matrix Factorization Models for Recommender Systems[C]. *The Twenty-Sixth International Joint Conference on Artificial Intelligence*, 2017: 3203-3209.
- [23] Kossaiji J, Panagakis Y, Anandkumar A, et al. TensorLy: Tensor

Learning In Python[J]. *Journal of Machine Learning Research*, 2019, 20: 1-6.

[24] Ketkar N. Introduction to PyTorch[M]. Deep Learning with Python.

Berkeley, CA: Apress, 2017: 195-208.

[25] Kingma D, Ba J. Adam: A Method for Stochastic Optimization[EB/OL]. 2014



许荣海 中山大学 2017 级大四在读本科生, 信息与计算科学专业。主要研究方向: 个性化推荐系统。



王昌栋 男, 广东河源人, CCF 会员, 2013 年于中山大学获计算机应用技术博士学位, 现为中山大学数据科学与计算机学院副教授、博士生导师, 主要研究领域为数据挖掘与机器学习。