文章编号：2095-2163(2020)04-0022-06 中图分类号：TP391 文献标志码：A

基于知识图谱表示学习的推荐算法优化

郝 卫，魏 赟
（上海理工大学 光电信息与计算机工程学院，上海 200093）

摘 要：本文提出基于知识图谱表示学习的推荐算法 TransH-CF。通过表示学习方法，将电影数据集中的实体映射成对应的实体向量，嵌入到低维空间中来，计算不同电影之间的语义相似度，与协同过滤计算出的电影相似度相结合，将混合后的结果推荐给用户。本文选取 TransH 翻译方法，与改进后基于物品的协同过滤算法相结合，弥补了传统协同过滤算法在热门电影相似度计算时的劣势，也解决了 TransE 翻译方法在一对多，多对一，多对多关系建模的劣势。实验结果表明，，此算法有效的提高了电影推荐的准确率、召回率、F 值等评估因素。

关键词：知识图谱；表示学习；协同过滤；电影推荐

Optimization of recommendation system based on representation learning of knowledge graph

HAO Wei, WEI Yun
(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

【Abstract】 This paper proposes a recommendation algorithm based on knowledge graph representation learning TransH-CF: maps the entities in the movie data set into corresponding entity vectors through the representation learning method, and embeds them in the low-dimensional space to calculate the semantic similarity between different movies, and collaboration The filtered calculated movie similarity is combined to recommend the mixed result to the user. In this paper, the TransH translation method and the improved item-based collaborative filtering algorithm are combined to make up for the disadvantages of the traditional collaborative filtering algorithm in calculating the similarity of popular movies, and also solve the disadvantage of the TransE translation method in one-to-many, many-to-one, many-pair multi-relational modeling. Experimental results show that this algorithm can effectively improve the accuracy of movie recommendation, recall rate, F value and other evaluation factors.

【Key words】 kowledge graph; representation learning; collaborative filtering; movie recommendation

0 引 言

近年来随着信息技术的迅速发展，互联网数据量呈指数型增长，给我们生活方式带来便利的同时也明显的带来了信息过载问题[1]。在巨大的数据量和复杂的信息面前，用户如何得到自身感兴趣或自身需要的信息成了当前研究的热点问题，具有代表性的解决方案包括搜索引擎和推荐系统的研究[2-3]。相比于搜索引擎只能被动的为用户提供筛选和过滤的功能，推荐系统可以主动的为更多的用户服务千人千面的需求。当前推荐系统已经运用在电子商务，新闻媒体等领域。具有代表性的应用包括淘宝、京东等电子商务平台；今日头条、趣头条等新闻媒体；Netfli、YouTube 等视频软件。

2012 年谷歌正式将知识图谱应用于应用谷歌搜索中，用来提高搜索引擎展示答案的质量和用户查询的效率。知识图谱的概念最早可以引申到上个世纪五十年代提出的语义网络[4]，区别在于知识图谱侧重于实体间关系的描述，语义网络侧重于本体与本体间关系的描述。知识图谱包含了大量的实体间关联关系，可以融合多种数据源和大量的数据来丰富物品/项目的语义信息，在众多的场景中都可以与推荐系统相结合，包括电影推荐，购物推荐等等。通过将知识图谱与推荐系统相结合，可以有效的提高推荐结果的准确性、可解释性等评估指标。

一种基于表示学习的推荐算法，以此来提高推荐算法的推荐效果：通过知识图谱表示学习方法计算不同电影之间的语义相似度，使用加权型混合方法中的线性模型方法与改进后的协同过滤[5]计算出的电影相似度结果进行融合，得到最终的推荐结果，推荐给用户。

基金项目：国家重点研发计划项目(2018YFB1700902)。

作者简介：郝 卫(1993-)，男，硕士研究生，主要研究方向：推荐系统、知识图谱；魏 赟(1976-)，女，博士，副教授，研究方向：智能交通、深度学习。

收稿日期：2020-02-21

1　理论介绍

1.1　协同过滤推荐算法

目前协同过滤算法广泛地运用在电子商务，新闻媒体，视频软件等领域。协同过滤算法主要包括基于用户，基于物品和基于模型三种类型。基于用户的协同过滤算法是利用相似统计的方法找到具有相似爱好的用户，再将这些用户感兴趣的物品/项目推荐给目标用户；基于物品的协同过滤算法是通过计算物品/项目间的相似度，并结合用户的历史行为，生成推荐列表，将结果推荐给用户；基于模型的协同过滤算法是采用数据挖掘和机器学习的方法，对物品的评分矩阵样本集训练学习，建立合适的推荐模型，生成推荐列表，将结果推荐给用户。

1.2　知识图谱

知识图谱是结构化的语义知识库，通过将关键词映射到语义知识库，准确的匹配到库中的实体，从而将用户需要的答案反馈给用户。通常用实体来表示人，公司，概念等现实中的实体，用关系来描述实体间的联系。三元组基本形式是“实体(entity)-关系(relation)-实体(entity)”或“实体(entity)-属性(attribute)-属性值(value)”，通过这样的三元组可以表示库中各实体间的关系和实体的的属性。通过多个三元组之间的关联关系，可以生成特定领域内的知识图谱，知识图谱可以更加直接形象的描述实体之间的关联关系，当两个实体在知识图谱中很相似，就意味着二者在语义上十分接近，就可以判定二者是近邻。

1.3　知识图谱在推荐系统中应用介绍

随着知识图谱研究的兴起，依靠知识图谱较强的可解释性以及语义相似度计算，一些学者试着将知识图谱与协同过滤算法进行融合，以此来提高推荐系统的准确性等参数：Dodwad 提出基于本体的层次结构对概念进行加权的应用方法，可以将用户的偏好分析的粒度划分的更细，使得结果更加的精准；Noia 通过利用基于开放链接数据语义丰富的特点，效的提高电影推荐的准确率；László 提出把电影数据集与用户信息嵌入到同一个向量空间内，计算电影与用户之间的空间距离，使得电影推荐的结果更准确，解释性更高。

目前常用的通过知识图谱中的表示学习方法，利用知识图谱将三元组嵌入低维空间进行向量化，可以进行实体间语义相似度计算。表示学习实现了实体和关系的分布式表示，可以显著提高计算效率，有效的缓解数据稀疏性等。目前表示学习主要运用在语义相似度计算，知识图谱补全，关系抽取和自动问答等方面[6]。

目前知识图谱表示学习实现方法较多，考虑到构建的电影知识图谱电影数量众多并且评分矩阵具有一定的稀疏性，所以选择了翻译模型[7]作为本算法的实现方法。

2　基于知识图谱表示学习的推荐算法

2.1　基于 TransH 算法的知识图谱表示学习

运用较为广泛的翻译模型主要包括 TransE，TransH，TransR 等模型；最为基础的 TransE 算法使用 (h,r,t) 来表示知识图谱的三元组，h 表示头实体，r 表示关系，t 表示尾实体。通过 TransE 模型计算出实体间的语义信息，与协同过滤算法融合后进行推荐，混合推荐效果得到了一定的提升，如图 1 所示。

TransE 算法的缺陷在于不能很好的解决一对多，多对一，多对多关系建模的难题。本文选用的 TransH 算法可以很好的解决上述问题。与 TransE 不同的是，TransH 引入了两个新的定义：超平面和关系向量，对于超平面 W，可以用法向量 W_r 来表示，用 d_r 来表示 $h_\perp$ 、$t_\perp$ 的间距。。关系向量 $h_\perp$ 是 h 向量在超平面上的投影，$t_\perp$ 是 t 向量在超平面上的投影。通过此操作可以在保证复杂度与 TransE 算法相近的前提下，

保留复杂的关系映射属性，如图 2 所示。

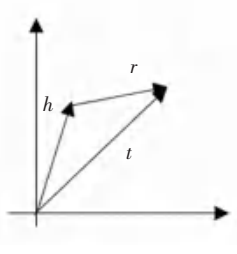

图 1　TransE 模型

Fig. 1　TransE model

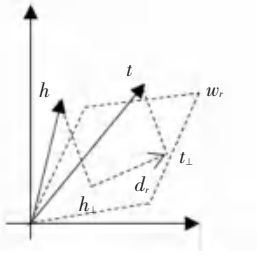

图 2　TransH 模型

Fig. 2　TransH model

为了满足任一实体在不同的关系中的意义不同，与此同时不同实体在相同关系中的意义相同，三元组需要满足 $h_\perp + d_r = t_\perp$。对于超平面 W，可以用法向量 W_r 来表示可以得到头实体和尾实体映射到超平面的公式(1)和公式(2)：

$$h_\perp = h - w_r^{\mathrm{T}} h\, w_r, \tag{1}$$

$$t_\perp = t - w_r^{\mathrm{T}} t\, w_r, \tag{2}$$

增加约束条件：$\| w_r \|_2^2 = 1$。可以用损失公式(3)进行训练：

$$f_r(h,t) = \| (h - W_r^T h\, W_r) + d_{r-(t-W_r^T t W_r)} \|_2^2. \tag{3}$$

TransH 模型训练的核心思想是采用最大间距。

2.2　TransH-CF 混合过滤推荐算法

结合知识图谱和协同过滤算法的各自的优势，

本文提出一种基于知识图谱表示学习的协同过滤算法:通过分析处理已经成熟的数据集,得到用户对电影的评分矩阵,利用协同过滤算法计算出电影的相似度;将构建电影知识图谱中的电影实体嵌入到低维空间中进行向量化,计算电影实体之间的语义相似度,最后将二者计算出的相似度融合,生成最终的推荐列表推荐给用户。考虑到电影领域内影片的数量远远小于用户数量,且具有更好的解释性,所以选择基于物品的协同过滤算法进行算法实现。

本文提出的 TransH-CF 算法流程如图 3 所示。

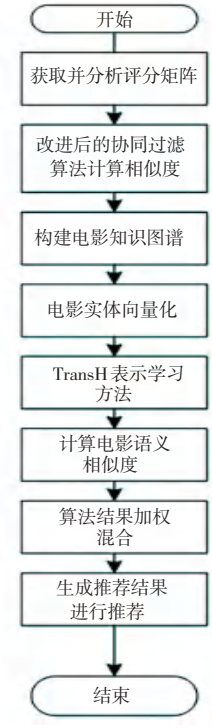

图 3 TransH-CF 算法流程图

Fig. 3 TransH-CF Algorithm Flow Chart

算法描述:

input:电影-用户评分矩阵和电影知识图谱

output:基于知识图谱表示学习的推荐算法 TransH-CF

(1)根据电影-用户评分矩阵 $R_{m\times n}$,通过协同过滤算法计算电影之间的相似度;

(2)将电影与电影知识图谱中的实体一一对应,得到实体对应列表;

(3)通过 TransH 模型计算电影之间的语义相似度;

(4)算法融合,选择合适的两种算法融合比例,并确定近邻数 K 的数值和嵌入的维度;

(5)生成推荐列表推荐,推荐给用户。

2.3 基于知识图谱的语义相似度计算

表示学习目的在于将语义信息表示为低维向量。在低维向量空间中,两个实体距离越近,说明两个实体的语义相似度越高。本文采用较为常用欧式距离对于电影实体进行相似度计算。对于 A,B 两个实体向量,相似性度量公式(4):

$$\sin_{sem}(A,B) = 1 - \frac{\| A - B \|}{\| A - B \| + 1}. \tag{4}$$

从公式(4)可以得出,当计算结果数值越接近 1,实体向量电影 A 和电影 B 越相似,在构建出的电影知识图谱中两者语义相似度越高;当数值越接近 0,表示在构建出的电影知识图谱中两者语义相似度越低。

2.4 改进后的基于物品的协同过滤及其相似度计算

户观看完电影后,会根据自己的喜好对电影进行打分评价,通过数据处理可以得到一个用户-电影的评分数据集,评分的高低代表用户对于电影的喜好程度。将用户-电影的评分数据集作为协同过滤算法的输入,计算电影的相似度,根据得出的电影相似度生成电影推荐列表,推荐给用户。

假设数据集中有 n 个用户 $I = (I_1, I_2, \cdots I_n)$,共有 m 部电影 $U = (U_1, U_2, \cdots U_m)$,用户 - 电影的评分矩阵为$R_{m\times n}$,评分矩阵$R_{m\times n}$ 可以表示为:

$$R_{m\times n} = \begin{pmatrix} R_{11} & R_{12} & \cdots & R_{1m} \\ R_{21} & R_{22} & \cdots & R_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ R_{n1} & R_{n2} & \cdots & R_{nm} \end{pmatrix}.$$

评分矩阵中,每个 R 代表某用户对某部电影的喜好程度即评分。例如:分数 R_{ij} 表示是用户I_i 对于电影U_j 的评分, 评分区间为[1,5]。当同一用户对于不同电影评分相近时,则判定他们为近邻。

目前比较多用的相似度计算方法包括欧几里得距离,皮尔逊相关系数,余弦相似度计算等。由于评分矩阵的稀疏性,所以采用余弦相似度计算方法。余弦相似度计算方法是计算出两个评分向量之间的夹角余弦值,计算公式(5):

$$\cos(\theta)_{(A,B)} = \frac{A \cdot B}{\| A \| \cdot \| B \|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i 2)} \times \sqrt{\sum_{i=1}^{n} (B_i 2)}}. \tag{5}$$

余弦值的范围为[0,1],计算出的余弦值越接近 0 时,表明评分差异越大;计算出的余弦值越接近 1 时,表明评分差异越小。

传统的协同过滤算法通过比较用户共同评分过的电影,根据电影的评分差值进行相似度计算。这种相似度计算模型没有考虑到电影的热门程度对相似度的

影响，某些热门电影无法体现或者较少地体现用户的个性和潜在的兴趣爱好，也就无法表现出用户间的强相似性，这时应该弱化它们对相似度计算的影响。所以有研究人员推出了物品的热门程度计算公式(6)：

$$pf_c = f_c^{-a}, \tag{6}$$

其中，pf_c 是为电影 c 的热门因子，代表电影 c 热门程度对相似度的贡献；α 参数称之为平衡参数，其取值范围为 α>0，表示电影的热门程度对热门因子的影响因数。f_c 是电影的热门程度，计算公式(7)：

$$f_c = \frac{|m_c|}{|M|}, f_c \in (0,1), \tag{7}$$

其中 m_c 代表对电影进行评分的用户数量，M 代表用户总数量。

改进后的余弦相似度计算公式(8)：

$$\cos(\theta)_{(A,B)} = \frac{\sum_{i=1}^{n} A_i \times B_i \times pf_c}{\sqrt{\sum_{i=1}^{n}(A_i^2)} \times \sqrt{\sum_{i=1}^{n}(B_i^2)}}. \tag{8}$$

可见，当电影热门程度越高时，pf_c 越小，对于相似度计算的影响越小；反之当电影越冷门，pf_c 越大，相似度则会得到加强。

2.5　相似度融合

目前在推荐系统中使用较多的相似度融合方法包括加权型混合推荐中的线性模型，回归模型(Logistic Regression)，GBDT (Gradient Boosted Decision Trees)等方法，本文使用加权型混合的线性模型方法。通过设置参数，将两种算法进行混合，在数据集上反复测试，最终得到最合适的参数值，使得推荐结果最为准确。计算公式如式(9)所示：

$$\sin C = \partial \sin_{sem}(A,B) + (1-\partial)\cos(\theta)_{(A,B)} \tag{9}$$

其中，$\sin C$ 为经过算法混合后的最终电影相似度，$\sin C$ 为基于物品的协同过滤计算出的电影相似度，$\sin_{sem}(A,B)$ 为基于知识图谱表示学习计算出的语义相似度。α 为比例参数，代表基于物品的协同过滤算法计算出的电影相似度在融合算法计算出的电影相似度占比，α 的范围设定为[0,1]。

3　实验结果及分析

3.1　数据集

本文实验使用 Amazon 的电影评分数据集，数据集记录了用户对电影的评价，评分范围从 1 分到 5 分，1 代表用户对于电影喜好程度最低，5 代表用户对于电影喜好程度最高。使用 metadata 将数据集中电影的编号与该电影所在知识图谱中的实体一一对应。本文选用了 Freebase 中的电影本体作为本实验的知识图谱数据集，版本号为 2012-11-9。数据集中包括了导演，演员等本体对象。用实体的名称来代替数据集中电影的 id，提取出所需要的电影实体和电影实体之间的关系。最终得到共计 20 个知识图谱的语义关系。对于数据集采用了字符串规则匹配的方法进行数据清理，使得 Amazon 评分数据和 Freebase 数据集可以更好地匹配。

3.2　实验结果评价指标

评估推荐系统的指标包括准确率 (*precision*)，*F* 值(*F* - *measure*)，召回率(*recall*)。准确率体现了推荐给用户喜欢的电影占总推荐电影数量的比例，召回率体现了推荐的电影占用户喜欢电影的比例，*F* 值是二者的调和平均值，能够更加合理的评估推荐算法。

3.3　算法结果分析

3.3.1　两种算法混合比例以及嵌入维度的确定

本文使用加权型混合方法中的线性模型方法与协同过滤的推荐算法进行融合，在实验中首先确定融合比例，比例不同，电影推荐的效果也有所不同。本实验中 Top-K 近邻数 K 选取 100，表示学习嵌入维度选取 200。两种推荐算法混合的比例从 0∶10 到 10∶0 分别取整数值进行计算。图为实验结果中的准确率、召回率和 *F* 值的曲线。每组实验同数据同时测 10 次，得出的结果取平均值，结果如图 4-5 所示。

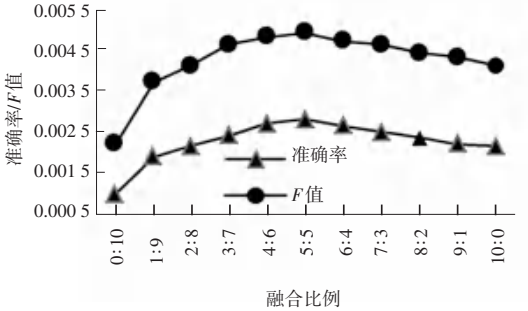

图 4　*K*=100 时的准确率和 *F* 值

Fig 4　Precision and F-measure at *K*=100

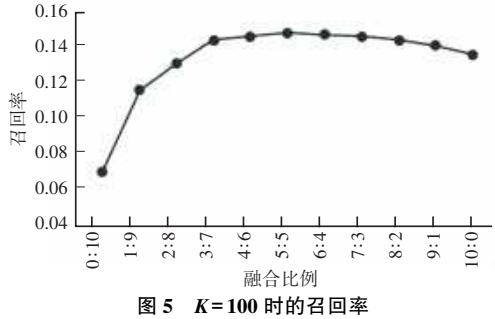

图 5　*K*=100 时的召回率

Fig. 5　Recall at *K*=100

从实验结果分析得到：在一定范围内准确率、召回率和 *F* 值随着协同过滤算法的占比增加而增加，超过了准确率、召回率和 *F* 值会逐渐下降，效果较差。对于本实验结果，采取表示学习和协同过滤算

法融合比例为 5∶5。

表示学习方法选择嵌入的维度不同也会影响最终推荐结果的评估。本实验选取 100,200,300,400,500 这 5 个维度进行试验,分别比较不同维度下的召回率、准确率和 *F* 值。每组实验相同数据测试 10 次,实验结果如 6-图 7 所示。

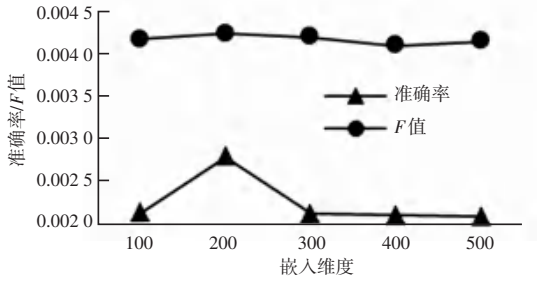

图 6 不同维度的准确率和 *F* 值

Fig. 6 Precision and *F*-measurein different dimension

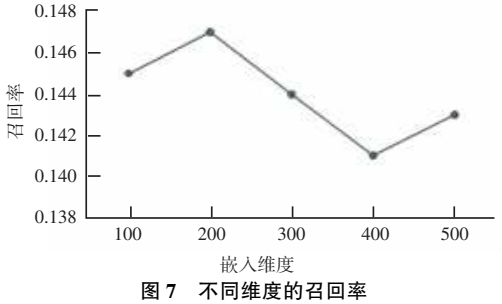

图 7 不同维度的召回率

Fig. 7 ecall in different dimension

从实验结果分析得到:当嵌入维度为 200 时,得到的三个评价指标最优,因此表示学习嵌入维度选定为 200。

3.3.2 算法结果对比

本文将提出的 TransH-CF 算法与知识图谱表示学习与协同过滤融合的 TransE-CF 算法,改进后的协同过滤算法:Cosine-CF 算法、Adjust Cosine-CF 算法、Pearson-CF 算法进行比较。在对比实验中,选择的四组近邻数分别为 60,80,100,120。每组实验做 10 次后取均值。结果如图 8~图 10 所示:

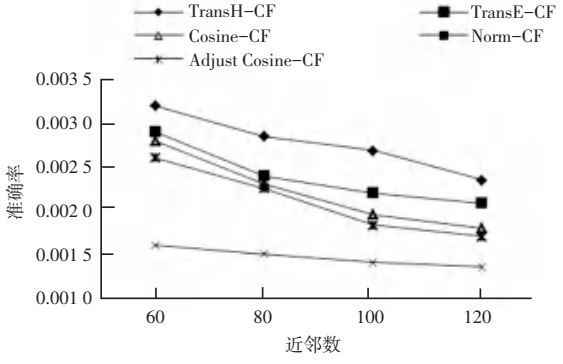

图 8 不同近邻数的准确率

Fig. 8 Precision in different *K*

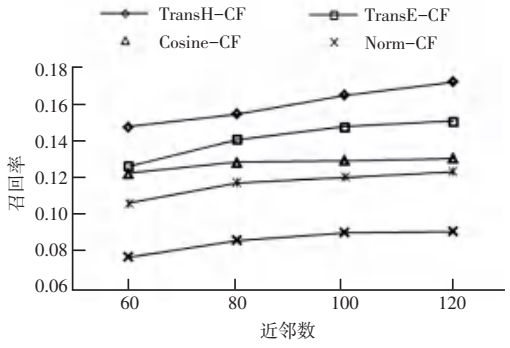

图 9 不同近邻数的召回率

Fig. 9 Recall in different *K*

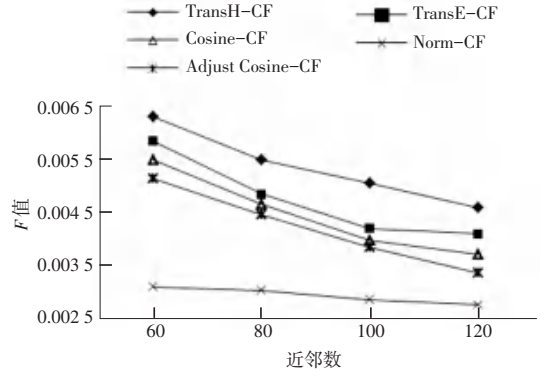

图 10 不同近邻数的 *F* 值

Fig.10 *F*-measure in different *K*

从实验结果分析得到:当选取嵌入维度为 200,近邻数为 60 到 120 之间时,TransH-CF 算法在准确率、召回率和 *F* 值 3 个评估参数均为最优。

4 结束语

本文提出了融合表示学习方法和改进后的协同过滤算法 TransH-CF:通过表示学习方法计算电影的语义相似度,与改进后的协同过滤算法计算出的电影相似度混合,得出最终的推荐结果,表示学习模型使用了 TransH 模型。实验数据表明 TransH-CF 算法可以有效的提高电影推荐的准确率、召回率以及 *F* 值等指标,TransH 算法与协同过滤混合推荐的结果也要优于 TransE 与协同过滤混合推荐的结果。利用实体的语义信息增强了推荐结果的可解释性。下一步将尝试将知识图谱运用到用户画像构建中,与协同过滤的结果进行混合推荐,进一步的优化推荐结果。

参考文献

[1] XIONG H, LIU Z. A situation information integrated personalized travel package recommendation approach based on TD-LDA model [C]// 2015 International Conference on Behavioral, Economic and Socio-cultural Computing (BESC). IEEE, 2015.

[2] JIAN Y, YIQUN C, GANG Z. Research and Development of Search Engine Technology[J]. Computer Engineering, 2005, 31(14):54-57.

(下转第 31 页)