Recommender System with KB by Qiao

Tuesday, September 8, 2020   2:42 PM

知识图谱和推荐系统。
By Qiao for NLP7 2020-09-13

Reference:
https://web.stanford.edu/class/cs246/
https://towardsdatascience.com/introduction-to-recommender-systems-6c66cf15ada
以及后面部分的论文

主要内容:
1. 回顾推荐系统任务
2. 基于内容和协同过滤的基础推荐方法
3. 三种利用知识图谱的方法:
    a. 融入KB Embeddings
    b. 利用KB 结构
    c. 结合使用

推荐系统:
- 文章推荐
- 商品推荐
- 电影推荐
- 音乐推荐
- 用户推荐
- ...

Users	User-item interactions matrix	Items
suscribers	rating given by a user to a movie (integer)	movies
readers	time spent by a reader on an article (float)	articles
buyers	product clicked or not when suggested (boolean)	products

...

关于推荐系统做什么的快速回顾:

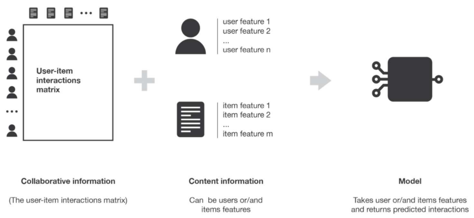
模型的形式化描述:
- X: 用户集合
- S: 对象集合(文章/商品..)
- Utility function $u: X \times S \rightarrow R$
    - R: 用户与对象交互结果的取值集合
    - R 通常是有序集合
    - 例如，1-5的打分分数、[0, 1]区间内的实值分数。

核心数据: User-item interaction matrix

	HP1	HP2	HP3	TW	SW1	SW2	SW3
A	4			5	1		
B	5	5	4				
C				2	4	5	
D		3					3

关键问题:
1. 收集已经产生的交互数据（收集矩阵中的已知元素）:
    通过众包、用户行为日志、历史互动记录等
2. 根据矩阵中已知部分，推测未知部分:
    推荐算法: 基于内容、协同过滤
3. 评估指标:
    - Rank Correlation, Top10Precision, RMSE等
    - Online A/B Test

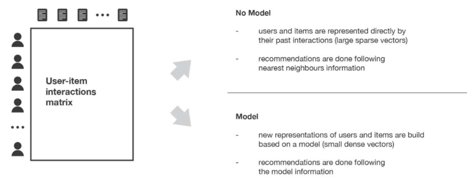
基于内容（Content-based）的推荐系统



Overview of the content based methods paradigm.

- **主要思想**: 为用户推荐与其访问历史对象最相似的对象
- **表示**
    - *表示用户*: 用户画像（特征向量）
        - 兴趣标签、Demographic信息
        - item画像的加权和（或其他Aggregation方法）
    - *表示对象*: Item画像（特征向量）
        - 电影: 作者、题目、演员、导演
        - 文本: tfidf 表示
        - ...
        - 用户画像的加权和（所有与该Item有交互的用户）
- **推荐**:
    - User-item画像的相似度计算（同一向量空间）
    - 机器学习

协同过滤（Collaborative filtering）系统



Overview of the collaborative filtering methods paradigm.

- **用户-用户（user-user）的协同过滤**

- **推荐User x Item** i:

$$r_{xi} = \frac{1}{K} \sum_{y \in KNN(x)} r_{yi}$$

或者加权平均:

$$r_{xi} = \frac{\sum_{y \in KNN(x)} s_{x,y} \cdot r_{yi}}{\sum_{y \in KNN(x)} s_{x,y}}$$

取与i有交互的用户集合中最接近x的top K个用户，计算用户相似度加权的分数和

- **对象-对象（item-item）的协同过滤**

- **推荐User x item** i:

$$r_{x,i} = \frac{\sum_{j \in KNN(i;x)} s_{i,j} \cdot r_{x,j}}{\sum_{j \in KNN(i;x)} s_{i,j}}$$

取x评过分的item集合中与i最相似的top K个，计算item相似度加权的分数和

User 和 Item 的表示:
1. 直接用User-Item矩阵中的行和列
2. 对User-Item矩阵Y进行分解，得到低维表示: $\mathbf{Y} = \mathbf{U} \cdot \mathbf{V}^{\mathrm{T}}$
    目标函数: $\min_{\mathbf{U},\mathbf{V}} \sum_{i,j} (\mathbf{U}_i \cdot \mathbf{V}_j^{\mathrm{T}} - \mathbf{Y}_{i,j})^2$ 或者写成矩阵形式 $\min_{\mathbf{U},\mathbf{V}} \left\|\mathbf{Y} - \mathbf{U} \cdot \mathbf{V}^{\mathrm{T}}\right\|_F^2$

知识图谱应用于推荐系统的三种思路（列举三篇代表性论文）：

比较全面的Survey:

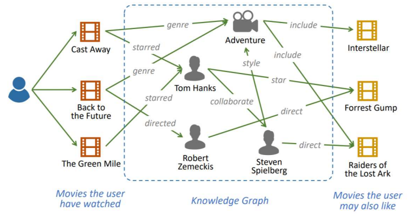
[A Survey on Knowledge Graph-Based Recommender Systems]

1. 在基于内容的推荐框架下使用Knowledge Embeddings丰富特征向量的表示
   DKN: Deep Knowledge-Aware Network for News Recommendation

2. 在协同过滤的框架下利用知识图谱的结构信息（路径近似度）约束/丰富矩阵分解后的表示。
   Collaborative Filtering with Entity Similarity Regularization in Heterogeneous Information Networks
   Personalized Entity Recommendation: A Heterogeneous Information Network Approach

3. 结合知识图谱嵌入于与图谱结构于一身的RippleNet

   RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems



i. 融合Embedding 的方法

DKN: Deep Knowledge-Aware Network for News Recommendation

Hongwei Wang[1,2], Fuzheng Zhang[2], Xing Xie[2], Minyi Guo[1*]
[1]Shanghai Jiao Tong University, Shanghai, China
[2]Microsoft Research Asia, Beijing, China

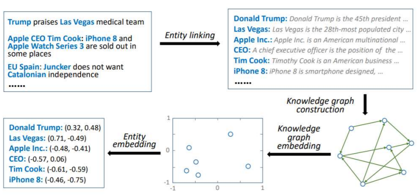


Figure 4: Illustration of knowledge distillation process.

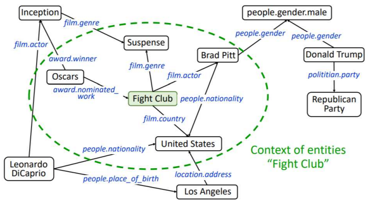


Figure 5: Illustration of context of an entity in a knowledge graph.

1. 实体连接（entity linking），匹配文本mention知识图谱中的实体；
2. 知识图谱构建。根据所有匹配到的实体，在原始的知识图谱中抽取子图。
3. 知识图谱特征学习。使用知识图谱特征学习算法（如TransE等）进行学习得到实体和关系向量。

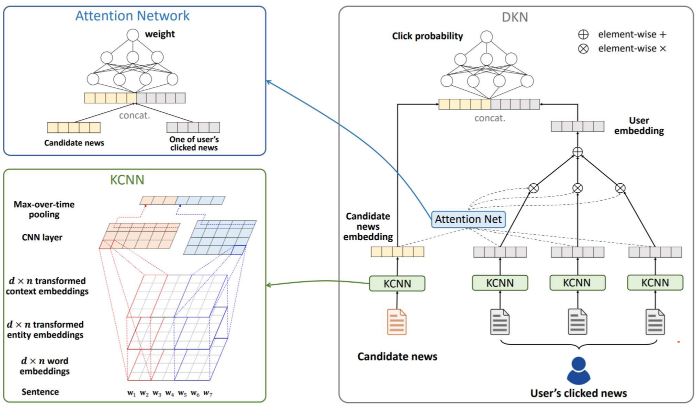
$$context(e) = \{e_i \mid (e, r, e_i) \in \mathcal{G} \text{ or } (e_i, r, e) \in \mathcal{G}\}$$

$$\bar{\mathbf{e}} = \frac{1}{|context(e)|} \sum_{e_i \in context(e)} \mathbf{e}_i$$

$$\mathbf{e}(t) = [\tilde{c}^{h_1} \ \tilde{c}^{h_2} \ \ldots \ \tilde{c}^{h_m}]$$

$$\tilde{c}^{h} = \max\{c_1^n, c_2^n, \ldots, c_{n-l+1}^n\}$$

$$c_i^n = f(\boldsymbol{h} * \mathbf{W}_{i:i+l-1} + b),$$



得到实体特征和实体上下文特征e, ē后，构建推荐模型，是一个基于CNN和注意力机制的新闻推荐算法：

1. 利用卷积神经网络提取文本特征：将新闻标题的词向量（word embedding）、实体向量（entity embedding）和实体上下文向量（context embedding）对齐，作为多个通道（类似于图像中的红绿蓝三通道），在CNN的框架下进行融合；
2. 利用注意力机制融合用户历史兴趣：在判断用户对当前新闻的兴趣时，使用注意力网络（attention network）给用户历史记录分配不同的权重，以该权重加权对历史阅读内容求和。
3. 将加权求和结果和candidate news的向量表示拼接后用预测模型预测得分。

$$\mathbf{W} = \left[[\mathbf{w}_1 \, g(\mathbf{e}_1) \, g(\bar{\mathbf{e}}_1)] \, [\mathbf{w}_2 \, g(\mathbf{e}_2) \, \bar{g}(\mathbf{e}_2)] \ldots [\mathbf{e}_n \, g(\mathbf{e}_n) \, g(\bar{\mathbf{e}}_n)]\right] \in \mathbb{R}^{d \times n \times 3}$$

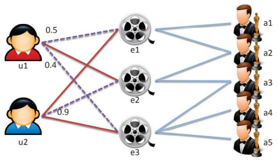
ii. 利用图谱结构的方法

Collaborative Filtering with Entity Similarity Regularization in Heterogeneous Information Networks

Xiao Yu[†] Xiang Ren[†] Quanquan Gu[‡] Yizhou Sun[‡] Jiawei Han[†]
[†] University of Illinois at Urbana-Champaign, Urbana, IL
[‡] Northeastern Univeristy, Boston, MA

Personalized Entity Recommendation: A Heterogeneous Information Network Approach

Xiao Yu[†], Xiang Ren[†], Yizhou Sun[†], Quanquan Gu[†],
Bradley Sturt[†], Urvashi Khandelwal[†], Brandon Norick[†], Jiawei Han[†]
[†]University of Illinois at Urbana-Champaign, Urbana, IL, USA



user — movie — actor — movie

Figure 4: User preference diffusion score calculation (Example 2). The solid red links represent observed user implicit feedback while the purple doted links represent diffused user preferences.

Meta-Paths:

user − item − ∗ − item

- $\mathcal{P}_1$: $user \xrightarrow{Viewed} movie \xrightarrow{Viewed^{-1}} user \xrightarrow{Follows} actor \xrightarrow{StarredIn} movie$
- $\mathcal{P}_2$: $user \xrightarrow{Viewed} movie \xrightarrow{StarredIn^{-1}} actor \xrightarrow{StarredIn} movie \xrightarrow{StarredIn^{-1}} actor \xrightarrow{StarredIn} movie$

路径关联分:

$$s_{x,y} = \frac{2 \times |\{p_{x \rightsquigarrow y} : p_{x \rightsquigarrow y} \in \mathcal{P}\}|}{|\{p_{x \rightsquigarrow x} : p_{x \rightsquigarrow x} \in \mathcal{P}\}| + |\{p_{y \rightsquigarrow y} : p_{y \rightsquigarrow y} \in \mathcal{P}\}|}$$

1. 利用路径关联分数约束矩阵分解(下为Weighted Non-negative Matrix Factorization, WNMF)

$$\begin{aligned} \min_{U,V,\boldsymbol{\theta}} \quad & \|Y \odot (R - UV^T)\|_F^2 + \lambda_0(\|U\|_F^2 + \|V\|_F^2) + \\ & \frac{\lambda_1}{2} \cdot \sum_{i,j} \sum_{l=1}^{L} \theta_l S_{ij}^{(l)} \|V_i - V_j\|_2^2 + \lambda_2 \|\boldsymbol{\theta}\|_2^2, \\ \text{s.t.} \quad & U \geq 0,\ V \geq 0,\ \boldsymbol{\theta} \geq 0,\ \text{and} \sum_{l=1}^{L} \theta_l = 1, \quad (4) \end{aligned}$$

除了上例的Item-item similarity 约束，还可以有下列的user-user以及user-item约束

- $\min_{\mathbf{U},\Theta} \sum_{l=1}^{L} \theta_l \sum_{i=1}^{m} \sum_{j=1}^{m} s_{i,j}^l \|\mathbf{u}_i - \mathbf{u}_j\|_F^2$
- $\min_{\mathbf{U},\mathbf{V},\Theta} \sum_{l=1}^{L} \theta_l \sum_{i=1}^{m} \sum_{j=1}^{n} (\mathbf{u}_i^T \mathbf{v}_j - s_{i,j}^l)^2$

$$\begin{aligned} & s(u_i, e_j|\mathcal{P}) \quad (2) \\ = & \sum_{e \in \mathcal{I}} \frac{2 \times R_{u_i,e} \times |\{p_{e \rightsquigarrow e_j} : p_{e \rightsquigarrow e_j} \in \mathcal{P}'\}|}{|\{p_{e \rightsquigarrow e} : p_{e \rightsquigarrow e} \in \mathcal{P}'\}| + |\{p_{e_j \rightsquigarrow e_j} : p_{e_j \rightsquigarrow e_j} \in \mathcal{P}'\}|} \end{aligned}$$

2. 利用路径相似度丰富user-iterm交互矩阵

$$\begin{aligned} (\hat{U}^{(q)}, \hat{V}^{(q)}) \quad & = \quad \text{argmin}_{U,V} \|\tilde{R}^{(q)} - UV^T\|_F^2 \\ \text{s.t.} \quad & \quad U \geq 0, \quad V \geq 0, \end{aligned}$$

$\tilde{R}^{(q)}$ 是第q条Meta-Path下所有$s_{u,e}$得分组成的矩阵，维度为# *users* * # *items*。

- 推荐模型（为u_i确定是否推荐实体e_j）：
  - $r(u_i, e_j) = \sum_{q=1}^{L} \theta_q \cdot \hat{U}_i^{(q)} \hat{V}_j^{(q)T}$
  - $r^*(u_i, e_j) = \sum_{k=1}^{c} sim(C_k, u_i) \sum_{q=1}^{L} \theta_q^{\{k\}} \cdot \hat{U}_i^{(q)} \hat{V}_j^{(q)T}$

iii. 混合的方法

RippleNet: Propagating User Preferences on the Knowledge Graph for Recommender Systems

Hongwei Wang[1,2], Fuzheng Zhang[3], Jialin Wang[4], Miao Zhao[4], Wenjie Li[4], Xing Xie[2], Minyi Guo[1*]
[1]Shanghai Jiao Tong University, wanghongwei55@gmail.com, guo-my@cs.sjtu.edu.cn
[2]Microsoft Research Asia, xingx@microsoft.com, [3]Meituan AI Lab, zhangfuzheng@meituan.com
[4]The Hong Kong Polytechnic University, {csjlwang, csmiaozhao, cswjli}@comp.polyu.edu.hk

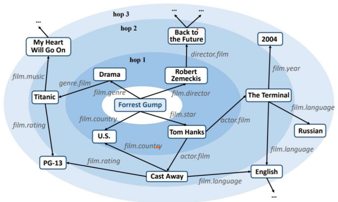


Figure 3: Illustration of ripple sets of "Forrest Gump" in KG of movies. The concentric circles denotes the ripple sets with different hops. The fading blue indicates decreasing relatedness between the center and surrounding entities.Note that the ripple sets of different hops are not necessarily disjoint in practice.

Definition 1 (relevant entity). *Given interaction matrix* Y *and knowledge graph* $\mathcal{G}$, *the set of k-hop relevant entities for user u is defined as*

$$\mathcal{E}_u^k = \{t \mid (h, r, t) \in \mathcal{G} \text{ and } h \in \mathcal{E}_u^{k-1}\}, \quad k = 1, 2, ..., H, \quad (2)$$

where $\mathcal{E}_u^0 = \mathcal{V}_u = \{v \mid y_{uv} = 1\}$ *is the set of user's clicked items in the past, which can be seen as the seed set of user u in KG.*

Definition 2 (ripple set). *The k-hop ripple set of user u is defined as the set of knowledge triples starting from* $\mathcal{E}_u^{k-1}$:

$$\mathcal{S}_u^k = \{(h, r, t) \mid (h, r, t) \in \mathcal{G} \text{ and } h \in \mathcal{E}_u^{k-1}\}, \quad k = 1, 2, ..., H. \quad (3)$$

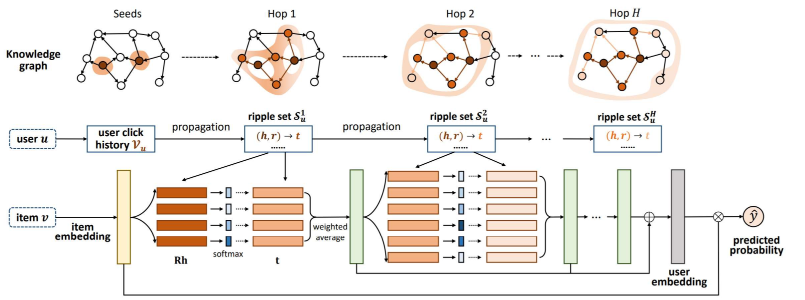


Figure 2: The overall framework of the RippleNet. It takes one user and one item as input, and outputs the predicted probability that the user will click the item. The KGs in the upper part illustrate the corresponding ripple sets activated by the user's click history.

推荐模型，为用户u推荐对象v:

$$\text{each triple } (h_i, r_i, t_i) \text{ in } \mathcal{S}_u^1$$

$$p_i = \text{softmax}\left(\mathbf{v}^T \mathbf{R}_i \mathbf{h}_i\right) = \frac{\exp\left(\mathbf{v}^T \mathbf{R}_i \mathbf{h}_i\right)}{\sum_{(h,r,t) \in \mathcal{S}_u^1} \exp\left(\mathbf{v}^T \mathbf{R} \mathbf{h}\right)}, \quad (4)$$

$$\mathbf{o}_u^1 = \sum_{(h_i, r_i, t_i) \in \mathcal{S}_u^1} p_i \mathbf{t}_i, \quad (5)$$

用$\mathbf{v}^t = \mathbf{o}_u^{t-1}$如上计算
$\mathbf{o}_u^2, \mathbf{o}_u^3, \dots \mathbf{o}_u^H$

$$\mathbf{u} = \mathbf{o}_u^1 + \mathbf{o}_u^2 + \ldots + \mathbf{o}_u^H, \quad (6)$$

$$\hat{y}_{uv} = \sigma(\mathbf{u}^T \mathbf{v}), \quad (7)$$

机器学习的损失函数

$$\begin{aligned} \min \mathcal{L} = & -\log\left(p(\mathbf{Y}|\Theta, \mathcal{G}) \cdot p(\mathcal{G}|\Theta) \cdot p(\Theta)\right) \\ = & \sum_{(u,v) \in \mathbf{Y}} -\left(y_{uv} \log \sigma(\mathbf{u}^T \mathbf{v}) + (1 - y_{uv}) \log\left(1 - \sigma(\mathbf{u}^T \mathbf{v})\right)\right) \\ & + \frac{\lambda_2}{2} \sum_{r \in \mathcal{R}} \|\mathbf{I}_r - \mathbf{E}^T \mathbf{R} \mathbf{E}\|_2^2 + \frac{\lambda_1}{2}\left(\|\mathbf{V}\|_2^2 + \|\mathbf{E}\|_2^2 + \sum_{r \in \mathcal{R}} \|\mathbf{R}\|_2^2\right) \end{aligned} \quad (13)$$

V和E是所有item和知识图谱实体的embedding矩阵
I_r 索引KG中具有关系r的知识,R是其关系embedding矩阵