

第七节课笔记 Chapter 3.3 因子分解机

https://github.com/cystanford/Recommended_System 老师的GitHub

第七节课笔记

课堂笔记

第二次反馈

第三次反馈

讲义内容

以上的MF (SVD, FunkSVD, BiasSVD, SVD++)，我们都只考虑user和item特征，但实际上一个预测问题包含的特征维度可能很多

FM因子分解机

Feature vector x															Target y							
$x^{(1)}$	1	0	0	...	1	0	0	0	...	0.3	0.3	0.3	0	...	13	0	0	0	0	...	5	$y^{(1)}$
$x^{(2)}$	1	0	0	...	0	1	0	0	...	0.3	0.3	0.3	0	...	14	1	0	0	0	...	3	$y^{(2)}$
$x^{(3)}$	1	0	0	...	0	0	1	0	...	0.3	0.3	0.3	0	...	16	0	1	0	0	...	1	$y^{(3)}$
$x^{(4)}$	0	1	0	...	0	0	1	0	...	0	0	0.5	0.5	...	5	0	0	0	0	...	4	$y^{(4)}$
$x^{(5)}$	0	1	0	...	0	0	0	1	...	0	0	0.5	0.5	...	8	0	0	1	0	...	5	$y^{(5)}$
$x^{(6)}$	0	0	1	...	1	0	0	0	...	0.5	0	0.5	0	...	9	0	0	0	0	...	1	$y^{(6)}$
$x^{(7)}$	0	0	1	...	0	0	1	0	...	0.5	0	0.5	0	...	12	1	0	0	0	...	5	$y^{(6)}$
A B C ... User				TI NH SW ST ... Movie				TI NH SW ST ... Other Movies rated				Time		TI NH SW ST ... Last Movie rated								

一个分数一行，onehot编码

FM中的特征交叉：CTR (Click Through Rate) 预估是个二分类问题

libFM (FM软件) --通过terminal调用

- 下载地址：<https://github.com/srendle/libfm>
- 使用文档：<http://www.libfm.org/libfm-1.42.manual.pdf>
- FM论文作者Steffen Rendle提供的工具 (2010年)
- 在KDD CUP 2012, 以及Music Hackathon中都取得不错的成绩
- 不仅适用于推荐系统, 还可以用于机器学习 (分类问题)
- 实现三种优化算法: SGD, ALS, MCMC
- 支持2种输入数据格式: 文本格式 (推荐) 和二进制格式

数据格式, 例如:

4 0:1.5 3:-7.9

2 1:1e-5 3:2

-1 6:1

第一列是y值，后面index:value

FFM算法:

Field-aware Factorization Machines for CTR

Prediction

<https://www.csie.ntu.edu.tw/~cjlin/papers/ffm.pdf>

通过引入field的概念，FFM把相同性质的特征归于同一个field，即把一个向量拆成多个，即权重的作用更加精细化

FM: label feat1:val1 feat2:val2

FFM: label field1:feat1:val1
field2:feat2:val2

criteo_ctr数据集:原始数

据: <https://labs.criteo.com/2013/12/download-terabyte-click-logs/>

xlearn工具输入数据格式: LR , FM 算法: CSV 或者
libsvm FFM 算法: libffm 格式

DeepFM算法: -- 高阶的情况用到DNN, 华为提出, DeepCTR里调取

DNN是高阶部分, FM是一阶和二阶项

DeepFM: A Factorization-Machine based Neural Network for CTR Prediction, 2017

<https://arxiv.org/abs/1703.04247> FM可以做特征组

合, 但是计算量大, 一般只考虑2阶特征组合 如何既考虑低阶 (1阶+2阶), 又能考虑到高阶特征 =>

DeepFM=FM+DNN 设计了一种end-to-end的模型结构

=> 无须特征工程 在各种benchmark和工程中效果好
Criteo点击率预测, 4500万用户点击记录, 90%样本用于训练, 10%用于测试 Company*游戏中心, 10亿记录, 连续7天用户点击记录用于训练, 之后1天用于测试

Dense embedding??

什么是Embedding:

TIPS: MAS学习法

- **Multi-dimensional,** 当我们接触越来越多的内容, 越需要总结和分析
- **Ask,** 带着问题思考, 一个好的问题是收获的开始
- **Sharing,** 分享是一种学习方式



Question?

- 一种降维方式，将不同特征转换为维度相同的向量
- 在推荐系统中，对于离线变量我们需要转换成 one-hot => 维度非常高，可以将其转换为 embedding 向量
- 原来每个 Field i 维度很高，都统一降成 k 维 embedding 向量



1.

$$Field(i) \Rightarrow e_i$$

- 方法：接入全连接层，对于每个 Field 只有一个位置为 1，其余为 0，因此得到的 embedding 就是图中连接的红线，对于 Field 1 来说就是

$$V_{11}, V_{12}, \dots, V_{1k}$$

- FM 模型和 Deep 模型中的子网络权重共享，也就是对于同一个特征，向量 V_i 是相同的

课堂笔记

- 关于找工作

在线部署？搭建系统
Action 作业整理出来，包装
整理常见面试的问题

基于内容的推荐：

物品表示 Item Representation：为每个 item 抽取出 features 特征学习 Profile Learning：利用一个用户过去喜欢（不喜欢）的 item 的特征数据，来学习该用户的喜好特征（profile）；生成推荐列表
Recommendation Generation：通过用户 profile 与候选 item 的特征，推荐相关性最大的 item。

为酒店建立内容推荐系统

余弦相似度：

什么是 N-Gram（N 元语法）：基于一个假设：第 n 个词出现与前 $n-1$ 个词相关，而与其他任何词不相关。
 $N=1$ 时为 unigram， $N=2$ 为 bigram， $N=3$ 为 trigram
N-Gram 指的是给定一段文本，其中的 N 个 item 的序列
比如文本：A B C D E，对应的 Bi-Gram 为 A B, B C, C D, D E 当一阶特征不够用时，可以用 N-Gram 做为新的特征。

比如在处理文本特征时，一个关键词是一个特征，但有

- 基于内容的推荐是属于静态属性；基于 ItemCF 是协同过滤，需要用户的行为，用户点击产生了矩阵中的向量--冷启动使用基于内容

FM 的价值，它的作用，它有哪些工具

Word2Vec: 我们要的是中间的神经元

些情况不够用，需要提取更多的特征，采用N-Gram
=> 可以理解是相邻两个关键词的特征组合

CountVectorizer：将文本中的词语转换为词频矩阵

TfidfVectorizer：将文档集合转化为tf-idf特征值的矩阵

什么是Embedding：一种降维方式，将不同特征转换为维度相同的向量 离线变量转换成one-hot => 维度非常高，可以将它转换为固定size的embedding向量 任何物体，都可以将它转换成为向量的形式，从Trait #1到 #N 向量之间，可以使用相似度进行计算 当我们进行推荐的时候，可以选择相似度最大的
king-man+woman与queen的相似度最高

Word2Vec：

通过Embedding，把原先词所在空间映射到一个新的空间中去，使得语义上相似的单词在该空间内距离相近。

Word Embedding => 学习隐藏层的权重矩阵

输入测是one-hot编码

隐藏层的神经元数量为hidden_size (Embedding Size)

对于输入层和隐藏层之间的权值矩阵W，大小为

[vocab_size, hidden_size]

输出层为[vocab_size]大小的向量，每一个值代表着输出一个词的概率

Word2Vec的两种模式：

Skip-Gram，给定input word预测上下文

CBOW，给定上下文，预测input word（与Skip-Gram相反）

Gensim工具pip install gensim

总结Summary

- 记录作业内容
 - Thinking1：在推荐系统中，FM和MF哪个应用的更多，为什么

- Thinking2: FFM与FM有哪些区别?
- Thinking3: DeepFM相比于FM解决了哪些问题, 原理是怎样的
- Thinking4: 假设一个小说网站, 有N部小说, 每部小说都有摘要描述。如何针对该网站制定基于内容的推荐系统, 即用户看了某部小说后, 推荐其他相关的小说。原理和步骤是怎样的
- Thinking5: Word2Vec的应用场景有哪些
- Action1: 使用libfm工具对movielens进行评分预测, 采用SGD优化算法
- Action2: 使用DeepFM对movielens进行评分预测
- Action3: 使用Gensim中的Word2Vec对三国演义进行Word Embedding, 分析和曹操最相近的词有哪些, 曹操+刘备-张飞=?

•