Item2vec理论与实践

这是微软论文ITEM2VEC: NEURAL ITEM EMBEDDING FOR COLLABORATIVE FILTERING。

# 前置点评

这篇文章比较朴素，创新性不高，基本是参照了google的word2vec方法，应用到推荐场景的i2i相似度计算中，但实际效果看还有有提升的。主要做法是把item视为word，用户的行为序列视为一个集合，item间的共现为正样本，并按照item的频率分布进行负样本采样，缺点是相似度的计算还只是利用到了item共现信息，1).忽略了user行为序列信息; 2).没有建模用户对不同item的喜欢程度高低。

# 背景

推荐系统中，传统的CF算法都是利用item2item关系计算商品间相似性。i2i数据在业界的推荐系统中起着非常重要的作用。传统的i2i的主要计算方法分两类，memory-based和model-based。

作者受nlp中运用embedding算法学习word的latent representation的启发，特别是参考了google发布的的word2vec（Skip-gram with Negative Sampling，SGNS），利用item-based CF学习item在低维latent space的embedding representation，优化i2i的计算。

这篇论文是微软将word2vec应用于推荐领域的一篇实用性很强的文章。该文的方法简单易用，可以说极大拓展了word2vec的应用范围，使其从NLP领域直接扩展到推荐、广告、搜索等任何可以生成sequence的领域。

**还有什么是 embedding？为什么说 embedding 是深度学习的基本操作？**

简单来说，embedding 就是用一个低维的向量表示一个物体，可以是一个词，或是一个商品，或是一个电影等等。这个 embedding 向量的性质是能使距离相近的向量对应的物体有相近的含义，比如 Embedding(复仇者联盟)和 Embedding(钢铁侠)之间的距离就会很接近，但Embedding(复仇者联盟)和Embedding(乱世佳人)的距离就会远一些。

除此之外 Embedding 甚至还具有数学运算的关系，比如Embedding（马德里）-Embedding（西班牙）+Embedding(法国)≈Embedding(巴黎)

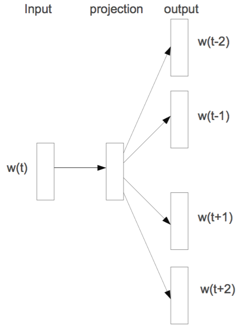
从另外一个空间表达物体，甚至揭示了物体间的潜在关系，从某种意义上来说，Embedding方法甚至具备了一些本体论的哲学意义。

言归正传，Embedding 能够用低维向量对物体进行编码还能保留其含义的特点非常适合深度学习。在传统机器学习模型构建过程中，我们经常使用one hot encoding对离散特征，特别是id类特征进行编码，但由于one hot encoding的维度等于物体的总数，比如阿里的商品one hot encoding的维度就至少是千万量级的。这样的编码方式对于商品来说是极端稀疏的，甚至用multi hot encoding对用户浏览历史的编码也会是一个非常稀疏的向量。而深度学习的特点以及工程方面的原因使其不利于稀疏特征向量的处理。因此如果能把物体编码为一个低维稠密向量再喂给DNN，自然是一个高效的基本操作。

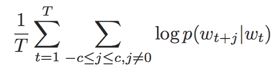
# 回顾word2vec：

自然语言处理中的neural embedding尝试把words and phrases映射到一个低维语义和句法的向量空间中。

Skip-gram的模型架构：

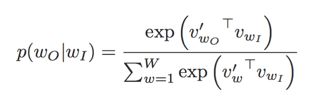


Skip-gram是利用当前词预测其上下文词。给定一个训练序列w1,w2,...,wT，模型的目标函数是最大化平均的log概率：

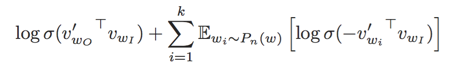


目标函数中c中context的大小。c越大，训练样本也就越大，准确率也越高，同时训练时间也会变长。

在skip-gram中，利用softmax函数定义如下：

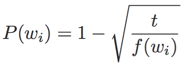


W是整个语料库的大小。上式的梯度的计算量正比于W，W通常非常大，直接计算上式是不现实的。为了解决这个问题，google提出了两个方法，一个是hierarchical softmax，另一个方法是negative sample。negative sample的思想本身源自于对Noise Contrastive Estimation的一个简化，具体的，把目标函数修正为：



是噪声分布( noise distribution )。即训练目标是使用Logistic regression区分出目标词和噪音词。具体的Pn(w)方面有些trick，google使用的是unigram的3/4方，即，好于unigram，uniform distribution。

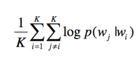
另外，由于自然语言中很多高频词出现频率极高，但包含的信息量非常小（如'is' 'a' 'the'）。为了balance低频词和高频词，利用简单的概率丢弃词wi：



其中f(wi)是wi的词频，t的确定比较trick，启发式获得。实际中t大约在10的-5次方附近。

# Item2vec算法原理

Item2vec中把用户浏览的商品集合等价于word2vec中的word的序列，即句子（忽略了商品序列空间信息spatial information）。出现在同一个集合的商品对视为正例，否则为负例。同一集合可以根据具体场景定义，例如：用户同一订单下的商品。对于集合w1,w2,…,wK目标函数：



同word2vec，利用负采样，将定义为：





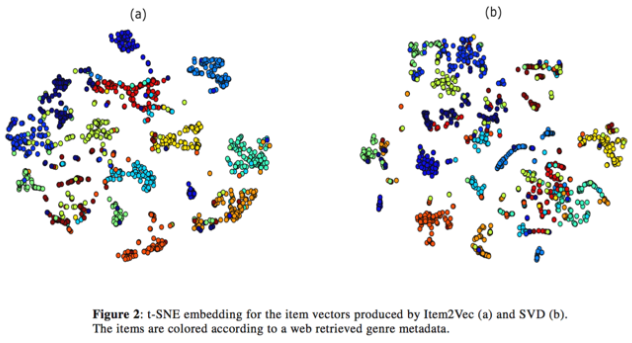
subsample的方式也是同word2vec：



最终，利用SGD方法学习的目标函数max，得到每个商品的embedding representation，商品之间两两计算cosine相似度即为商品的相似度。

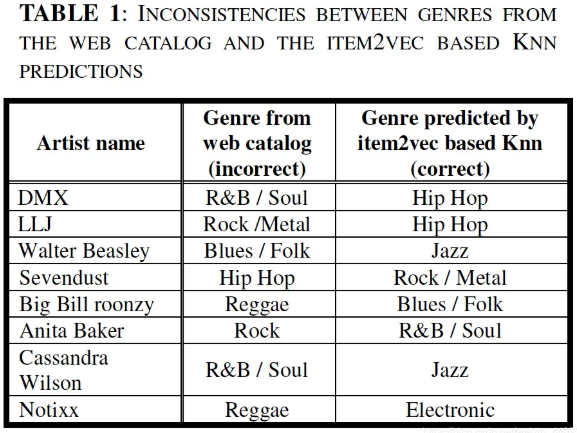
# Item2vec效果

对比的baseline方法是基于SVD方法的用户embedding得到的相似度，SVD分解的维度和item2vec的向量维度都取40，详细见paper。数据是应用在music领域的，作者利用web上音乐人的类别进行聚类，同一个颜色的节点表示相同类型的音乐人，结果对比如下：

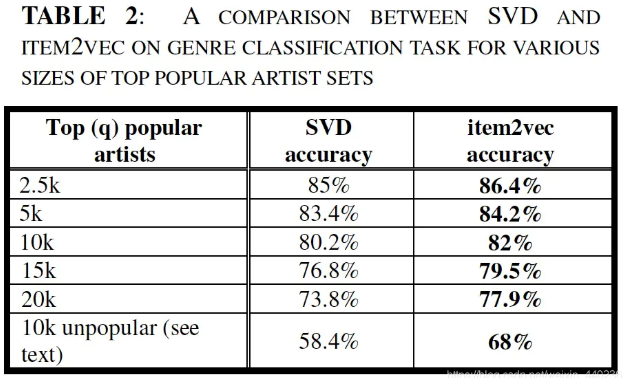


同种类型的artists用相同颜色进行标识，结果显示，图a是item2vec的聚合效果，图b是SVD分解的聚合效果，看起来item2vec的聚合效果更好些。

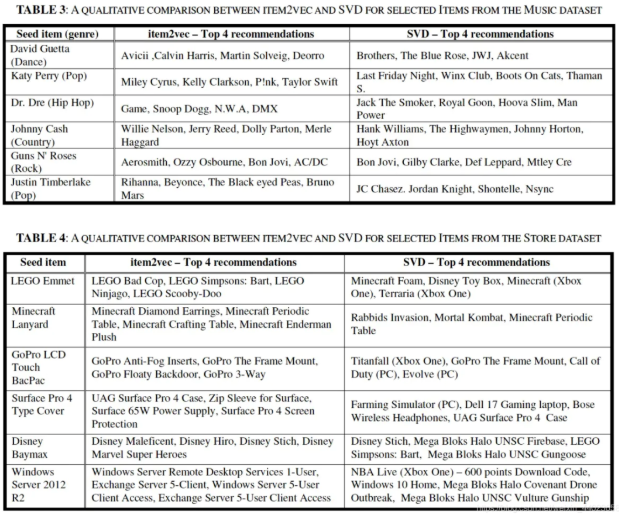
**能够使用简单的KNN方式对网上artists的类型进行纠正和标注**



**对topq的artists进行分类预测（KNN），准确率较高，且对历史数据较少的artists表示效果更好。**



item2vec和SVD直观效果比较，表中列除了seed item以及与其最相似item



可以得出结论：

1）item2vec可以探索出除了“类型”之外其他的相似性；

2）item2vec提供列更相关的item，且对于信息较少的情况下，表现较佳

# 总结

实践证明，embedding技术对于工业场景来说有着很大的价值和应用前景。

参考：

<https://zhuanlan.zhihu.com/p/24339183?refer=deeplearning-surfing>

<https://mp.weixin.qq.com/s/lnGvdLc4U6b0kniyFfPbbw>