知乎 首发于 深海遨游

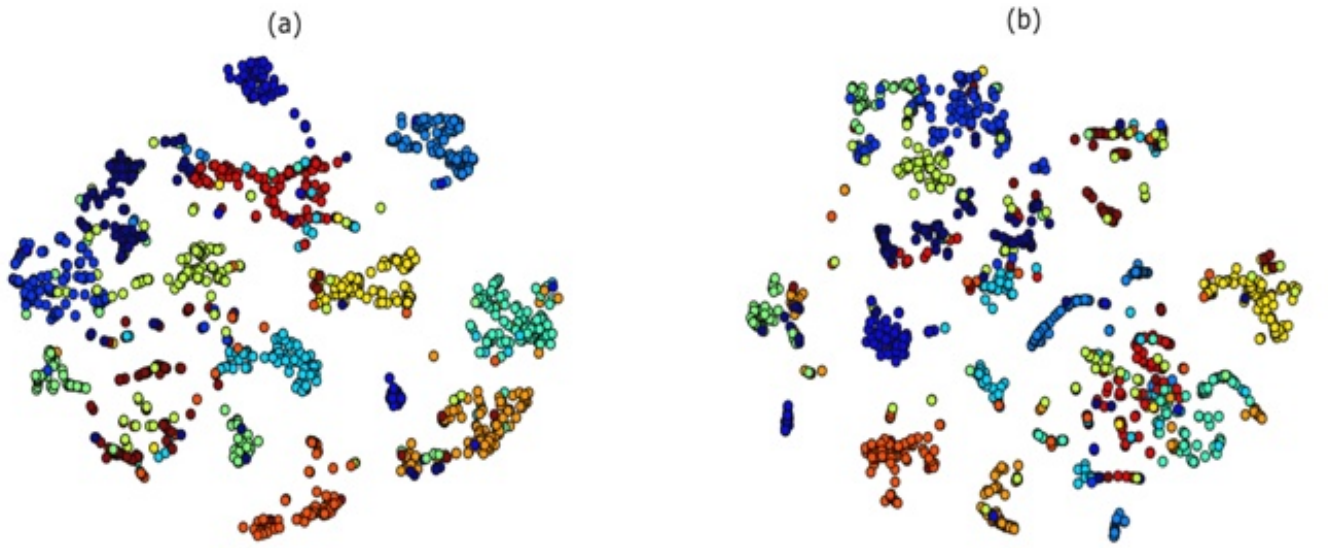


Figure 2: t-SNE embedding for the item vectors produced by Item2Vec (a) and SVD (b). The items are colored according to a web retrieved genre metadata.

DNN论文分享 - Item2vec

清淞
勇敢闯一闯

关注他

118 人赞同了该文章

清淞：Lazada搜索算法团队招人了～（阿里-搜索推荐事业部算法技术...
zhuanlan.zhihu.com

本篇文章在 ICML2016 Machine Learning for Music Discovery Workshop

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。

---

0 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

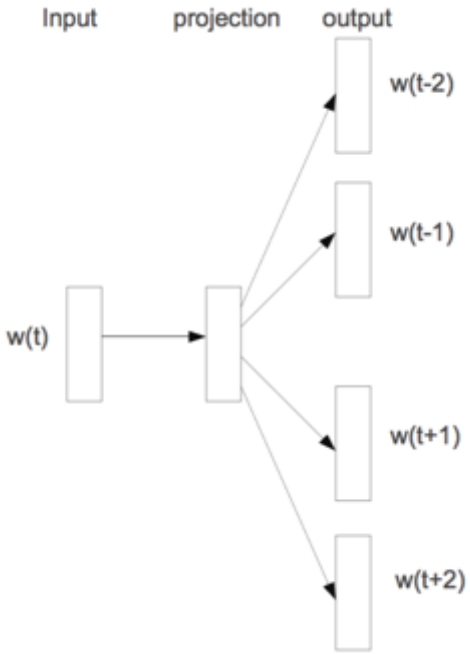
google发布的的word2vec（Skip-gram with Negative Sampling，SGNS），利用item-based CF 学习item在低维 latent space的 embedding representation，优化i2i的计算。

---

1 回顾下google的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 w_1 , w_2 ,..., w_T ， 模型的目标函数是最大化平均的log概率:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t)$$

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， $P(w_{t+j}|w_t)$ 利用softmax函数定义如下:

$$p(w_O|w_I) = \frac{}{\sum_{w=1}^{W} \exp\left({v'_w}^{\top} v_{w_I}\right)}$$

W是整个语料库的大小。上式的梯度的计算量正比如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：

$$\log \sigma({v'_{w_O}}^{\top} v_{w_I}) + \sum_{i=1}^{k} \mathbb{E}_{w_i \sim P_n(w)} \left[\log \sigma(-{v'_{w_i}}^{\top} v_{w_I})\right]$$

$P_n(w)$ 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 $U(w)^{3/4}/Z$，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 w_i：

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}}$$

其中 $f(w_i)$ 是 w_i 的词频，t的确定比较trick，启发式获得。实际中t大约在 10^{-5} 附近。

---

2 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 positive。对于集合 $w_1, w_2, \ldots, w_K$ 目标函数：

$$\frac{1}{K}\sum_{}^{K}\sum_{}^{K}$$

知乎 首发于 深海遨游

$$p(w_j \mid w_i) = \sigma(u_i^T v_j) \prod_{k=1}^{N} \sigma(-u_i^T v_k)$$

subsample的方式也是同word2vec：

$$p(discard \mid w) = 1 - \sqrt{\frac{\rho}{f(w)}}$$

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。

---

3 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：

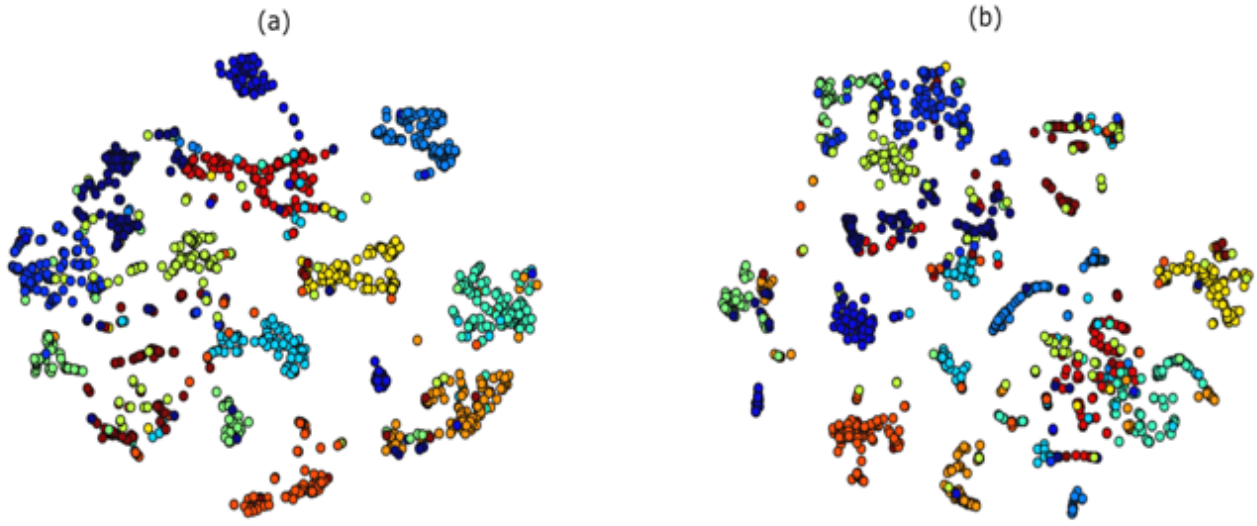


Figure 2: t-SNE embedding for the item vectors produced by Item2Vec (a) and SVD (b). The items are colored according to a web retrieved genre metadata.

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。