

Word embedding

词表达：机器理解自然语言的需要

ASCII, Unicode, etc: UTF-8

我们经常使用同义词或反义词表达一个单词的意思

编码/同义词表达的缺点：无法映射到连续空间进行表达

无法用数学计算相似性；需要大量的工作

If we treat words as mere categorical: one-hot encode

one-hot表达的问题：

无法表达词与词之间的关系：相似，反义，上下位，在句子中的地位
空间占用大，数学计算不友好，词典外词的插入会带来麻烦

one-hot编码把所有词看作互相独立变量 准度过高
降维：

PCA 数据矩阵 X $\text{Cov}(X) = (X - \bar{X})^T (X - \bar{X})$

求协方差矩阵的特征值和特征向量，按特征值从大到小排序：

$$w_{(1)} = \arg \max_{\|w\|=1} \|Xw\|^2 = \arg \max_{\|w\|=1} \{w^T X^T X w\}$$

先求最大

$$w_{(1)} = \arg \max_{\|w\|=1} \left\{ \frac{w^T X^T X w}{w^T w} \right\}$$

$$\hat{x}_R = X - \sum_{s=1}^{k-1} x_{(s)} w_{(s)}$$

$$w_{(k)} = \arg \max_{\|w\|=1} \{ \|\hat{x}_R w\|^2 \} = \arg \max_{\|w\|=1} \frac{w^T \hat{x}_R \hat{x}_R w}{w^T w}$$

再依次求其他的

$$T_L = XW_L$$

本质上，把数据变换到另外一个空间去

SVD:

$$X = U \Sigma W^T$$

PCA/SVD的问题：算法本身实现不复杂，加入新词，need重新计算
计算复杂度仍然很高，无法解释多义词

什么是好的词表示：

- ① space economical
- ② adaptively update \Rightarrow 加新词，语料，能及时更新
- ③ semantic similarity \Rightarrow 能计算词义相似度

好训练

单词的意思，受到上下文影响 上下文确定词 embedding

2013 Mikolov：语料充分，各词有份，上下文，相似度，训练集优化

两种主流算法：①skip-Gram ② Continuous Bags of Words (CBOW)

Tree softmax Negative sampling

比较：中心预测两边 两边预测中间

Tree softmax

Negative Sampling 这里效果最好

Glove embedding 最近邻概念 Glove本质是矩阵分解

Bag-of-word + 权平均 1. Weighted Bag-of-words + remove some special direction

Step 1：构建平滑的加权词向量

Step 2：计算PCA，减去最明显的第一个维度