CS224N

侯丽微

Reinforce Learing

loss:

$$\mathcal{L}_{XE} = \frac{1}{T}\sum_{t=1}^{T} -((1-\gamma)\log P^t_{attn}(w|x_{1:t}) + \gamma \log P^t_{cbdec}(w|x_{1:t})) \quad (1)$$

$$\mathcal{L}_{RL} = \frac{1}{T}\sum_{t=1}^{T}(r(\hat{y}) - r(y^s))\log P^t_{attn}(w^s_{t+1}|w^s_{1:t}) \quad (2)$$

$$\mathcal{L}_{XE+RL} = \lambda\mathcal{L}_{RL} + (1-\lambda)\mathcal{L}_{XE} \quad (3)$$

注：公式(1)是原文公式截图，感觉写错了。

类似于：A Deep Reinforced Model for Abstractive Summarization(2017)
Romain Paulus, Caiming Xiong, and Richard Socher.

Reinforce Learing

引入原因：

（1）exposure bias：训练和测试时decoder输入不一致，产生误差。

（2）类似解空间，组成句子的排序等不止只有一种，ROUGE在处理这个问题上比maximum-likelihood objective极大似然目标函数更灵活。

[teacher-forcing算法的问题是：一旦产生了前几个单词，训练就会被误导：严格遵守一个官方正确的摘要，但不能适应一个潜在正确但不同的开头。]

（3）RL没有直接去估算reward，而是使用了自己在测试时生成的句子作为baseline。sample时，那些比baseline好的句子就会获得正的权重，差的句子就会被抑制。

Reinforce Learing

$$L_{rl} = (r(\hat{y}) - r(y^s) \sum_{t=1}^{n} log\, p(y_t^s | y_1^s, \ldots, y_{t-1}^s, x)$$

公式的意思就是：对于如果当前sample到的词比测试阶段生成的词好，那么在这次词的维度上，整个式子的值就是负的（因为后面那一项一定为负），这样梯度就会上升，从而提高这个词的分数；而对于其他词，后面那一项为正，梯度就会下降，从而降低其他词的分数。

Lecture3

Lecture 3

LSA

浅层语义分析（LSA）是一种自然语言处理中用到的方法，其通过“矢量语义空间”来提取文档与词中的“概念”，进而分析文档与词之间的关系。LSA的基本假设是，如果两个词多次出现在同一文档中，则这两个词在语义上具有相似性。LSA使用大量的文本上构建一个矩阵，这个矩阵的一行代表一个词，一列代表一个文档，矩阵元素代表该词在该文档中出现的次数，然后再此矩阵上使用奇异值分解（SVD）来保留列信息的情况下减少矩阵行数，之后每两个词语的相似性则可以通过其行向量的cos值（或者归一化之后使用向量点乘）来进行标示，此值越接近于1则说明两个词语越相似，越接近于0则说明越不相似。

Window based co-occurrence matrix

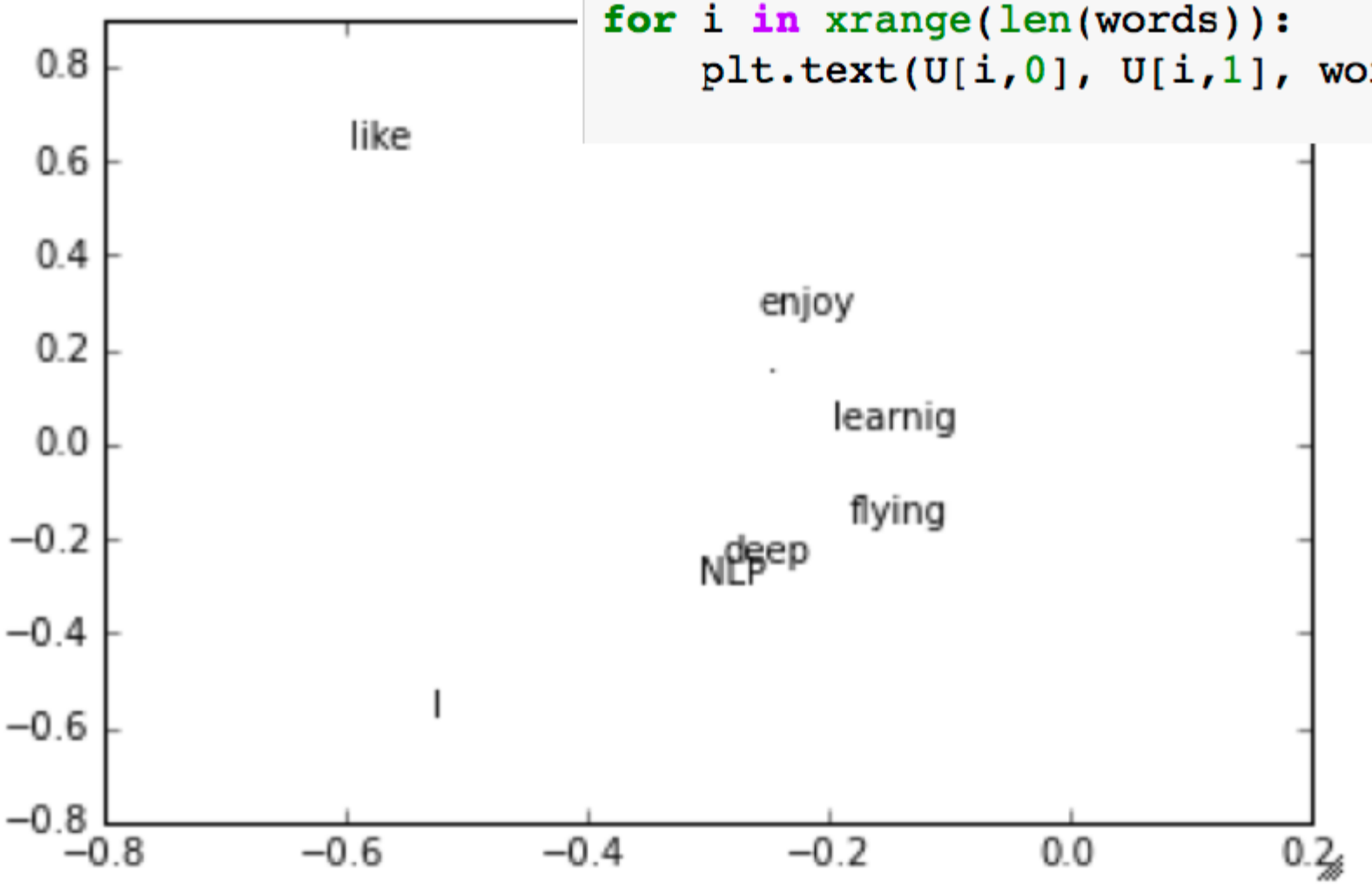
- Example corpus:
  - I like deep learning.
  - I like NLP.
  - I enjoy flying.

counts	I	like	enjoy	deep	learning	NLP	flying	.
I	0	2	1	0	0	0	0	0
like	2	0	0	1	0	1	0	0
enjoy	1	0	0	0	0	0	1	0
deep	0	1	0	0	1	0	0	0
learning	0	0	0	1	0	0	0	1
NLP	0	1	0	0	0	0	0	1
flying	0	0	1	0	0	0	0	1
.	0	0	0	0	1	1	1	0

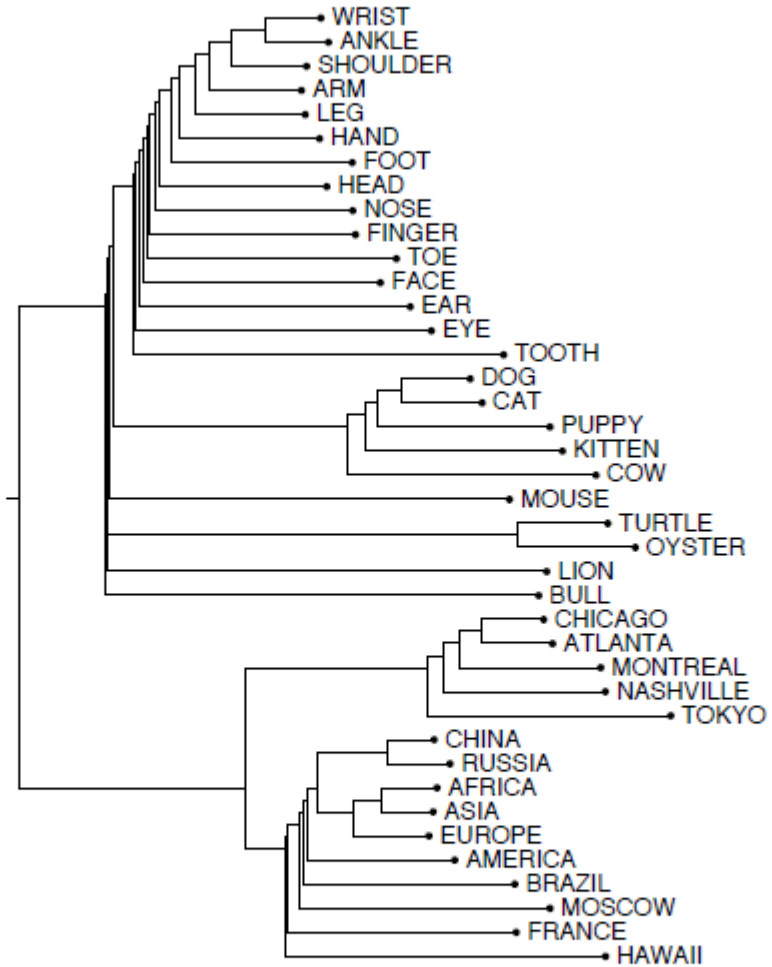
Simple SVD word vectors in Python

Corpus: I like deep learning. I like NLP. I enjoy flying.
Printing first two columns of U corresponding to the 2 biggest singular values

```
for i in xrange(len(words)):
    plt.text(U[i,0], U[i,1], words[i])
```

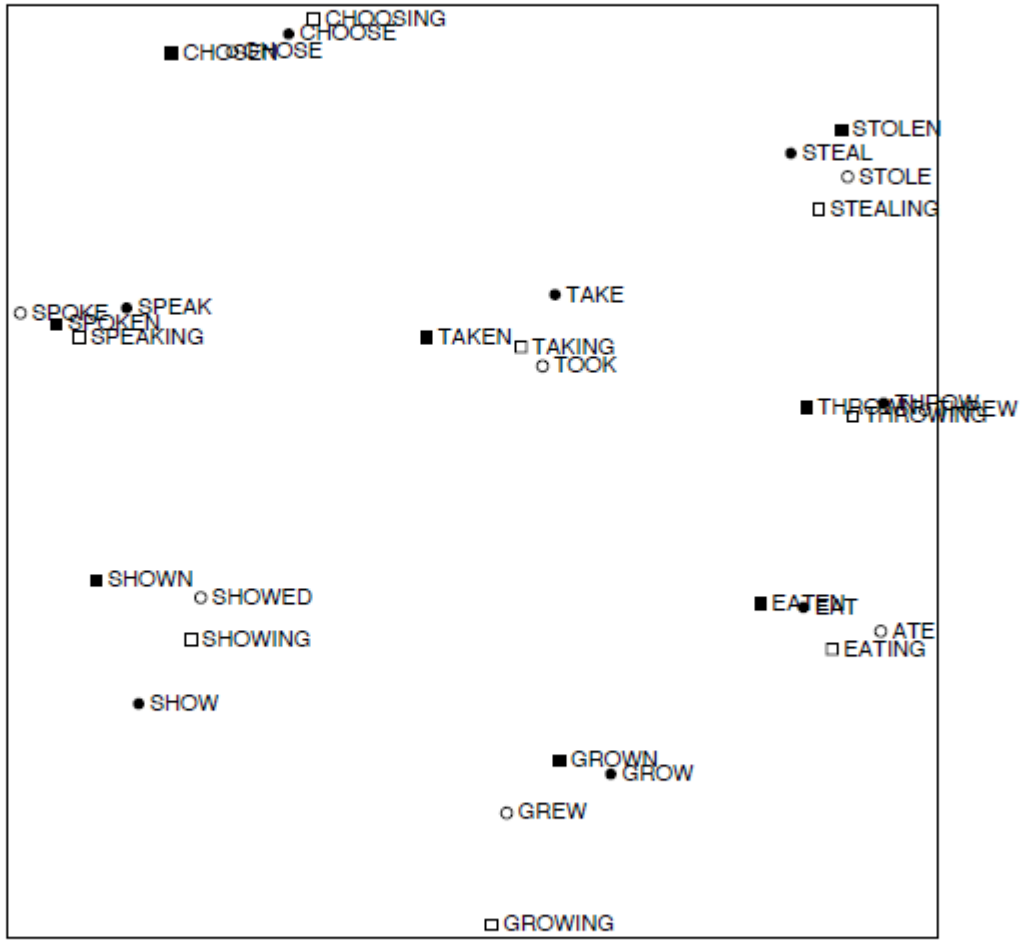


Interesting semantic patters emerge in the vectors



An Improved Model of Semantic Similarity Based on Lexical Co-Occurrence
Rohde et al. 2005

Interesting syntactic patters emerge in the vectors



An Improved Model of Semantic Similarity Based on Lexical Co-Occurrence
Rohde et al. 2005

GloVe

这个模型充分有效的利用了语料库的统计信息，仅仅利用共现矩阵里面的非零元素进行训练，skip_gram 没有很有效的利用语料库中的一些统计信息

GloVe

一、几个概念：

X_{ij}：词j在词i的上下文里共现次数。

X_i ：出现在词i上下文里的所有词的次数。

$P_{ij} = P(j|i) = X_{ij}/X_i$ ：词j出现在词i上下文中的概率

二、共现词概率中的某些规律/含义：

1 比如i = ice and j = steam,

2 要想了解这两个词的关系，可以通过研究他们与各种探测词k的共现概率的比率来获得。

3 四个探测词：solid、gas、water、fashion

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

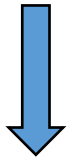
结论：词向量的学习应该是共现概率的比值而不是他们自己本身的概率。概率P_{ik}/P_{jk}依赖于三个词I, j, k。

GloVe

三、推导：

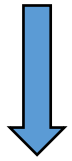
（1）结论由三个词向量推导出概率之比 $F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$ F可以是很多种形式，只要让F拟合右边的概率之比即可。

对F进行一些外力，
使之符合我们的要求



（2）First，F要将Pik/Pjk的信息编码到词向量空间里，换句话说，要将Pik和Pjk的差距距离在向量空间中表现出来，而向量空间是固有的线性结构，最自然的方法便是做向量差。

$$F(w_i - w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}. \qquad (2)$$



（3）Next，公式(2)右边是个标量，左边是向量。F虽然可以用复杂的网络比如nn，但这样会混淆我们试图捕获的线性结构，为避免这个问题，可以使用点乘。

$$F\left((w_i - w_j)^T \tilde{w}_k\right) = \frac{P_{ik}}{P_{jk}}, \qquad (3)$$

GloVe

三、推导：

（4）Next，对于word-word co-occurrence matrices,词和上下文词之间的角色可以任意转换，换句话就是公式应该是对称的，但公式(3)不符合要求，而该问题可以通过如下几步来解决：
第一步：我们要求F在 groups(R;+)和(R>0; x)之间是同形态的，即：

$$F\left((w_i - w_j)^T \tilde{w}_k\right) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}, \qquad (4)$$

第二步：通过公式(3)得到：

$$F(w_i^T \tilde{w}_k) = P_{ik} = \frac{X_{ik}}{X_i}. \qquad (5)$$

第三步：得到(4)这种公式的方法是$F = exp$（可直观推导出），将F替换为exp代入公式(5),得：

$$w_i^T \tilde{w}_k = \log(P_{ik}) = \log(X_{ik}) - \log(X_i). \qquad (6)$$

第四步：公式(6)损害了公式的对称性的主要原因是log(X_i),而这个和k没有依赖关系，因而将其吸收进w_i中，将其变为一个偏置b_i，为保证对称性也为w_k添加一个偏置b_k

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = \log(X_{ik}). \qquad (7)$$

GloVe

三、推导：

（5）Next，模型的一个主要的drawback，该模型对待所有共现词的权重是一样的，但是共现词的频率是不相同的，有的很罕见甚至没有，这种罕见的共现词是噪音或者相对于频次高的携带的信息很少，有些词频为0的能占到数据的75-95%。解决方案：weighted least squares regression model。

$$J = \sum_{i,j=1}^{V} f\left(X_{ij}\right)\left(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij}\right)^2 , \tag{8}$$

f(Xij)的需要拥有的性质：

$$f(x) = \begin{cases} (x/x_{\max})^{\alpha} & \text{if } x < x_{\max} \\ 1 & \text{otherwise} \end{cases} . \tag{9}$$

α = 3/4

1. $f(0) = 0$. If f is viewed as a continuous function, it should vanish as $x \to 0$ fast enough that the $\lim_{x\to 0} f(x)\log^2 x$ is finite.
2. $f(x)$ should be non-decreasing so that rare co-occurrences are not overweighted.
3. $f(x)$ should be relatively small for large values of x, so that frequent co-occurrences are not overweighted.

GloVe

四、和其他模型关系：

由于学习词向量的无监督方法都是基于对语料的共现统计来获得的，所以这些模型之间有一些共同点。

推导在后一页

$$Q_{ij} = \frac{\exp(\tilde{w}_i^T \cdot \tilde{w}_j)}{\sum_{j'=1}^{V} \exp(\tilde{w}_i^T \cdot \tilde{w}_{j'})}$$

loss交叉熵：$J = -\sum_{\substack{i \in corpus \\ j \in context(i)}} \log Q_{ij}$

故，若将有相同terms的结合在一起，便可提高计算效率：即

$$J = -\sum_{i=1}^{V} \sum_{j=1}^{V} x_{ij} \log Q_{ij}$$

↓

其中，x_{ij}来自于co-occurrence matrix X

由于 $x_i = \sum_k x_{ik}$ 和 $P_{ij} = x_{ij} / x_i$，即 $x_{ij} = P_{ij} \cdot x_i$，—从论文中获取

将其代入J中为↓

$$J = -\sum_{i=1}^{V} \sum_{j=1}^{V} x_{ij} \log Q_{ij} = -\sum_{i=1}^{V} \sum_{j=1}^{V} P_{ij} \cdot x_i \cdot \log Q_{ij} = -\sum_{i=1}^{V} x_i \sum_{j=1}^{V} P_{ij} \cdot \log Q_{ij}$$

其中 $H(P_i, Q_i) = \sum_{j=1}^{V} P_{ij} \log Q_{ij}$ 是P_i和Q_i的交叉熵（公式：$-\sum_i y_i' \log y_i$）

真实　预测

由于交叉熵有两个问题 { 长尾分布被建模的很差，因为被赋予太多权重。/ 计算开销，并也不需要归一化项的性质。} 因为改变公式J.

将P,Q中的归一化分母去掉，变为：$\hat{J} = \sum_{i,j} x_i (\hat{P}_{ij} - \hat{Q}_{ij})^2$，其中 $\hat{P}_{ij} = x_{ij}$，$\hat{Q}_{ij} = \exp(w_i^T \tilde{w}_j)$

~~即 $\hat{J} = \sum_{ij} x_i (\log$~~ 由于，$x_{ij}$ 的分布太大，~~[illegible]~~ Q为很大的数，因而将x_{ij}赋予一个log值可以解决该问题，∴将公式变为 $\hat{J} = \sum_{ij} x_i (\log \hat{P}_{ij} - \log \hat{Q}_{ij})^2 = \sum_{i,j} x_i (w_i^T \tilde{w}_j - \log x_{ij})^2$

将x_i替换为$f(x_{ij})$ 即 $\hat{J} = \sum_{ij} f(x_{ij}) (w_i^T \tilde{w}_j - \log x_{ij})^2$

完