Lettuce1

> 本节内容主要为引言和词向量

计算机中单词的表示

WordNet

- 一个包含**同义词(synonym)集**和**上位词(hypernym)**（“是一个”关系） 列表的**同义词库(thesaurus)**。
- 优点是能够表示单词的多种含义
- 缺点是难以计算单词间的相似度，以及需要手工创造词表

One-Hot

- 在传统的 NLP 中， 我们将单词视为离散的符号， 这样的单词符号可用**独热(one-hot)向量**表示。 独热向量中只有一个元素的值位 1， 其余元素的值均为 0。
- 缺点1：由于一般**one-hot向量**的维度等于词表大小，故维度过大，难以计算
- 缺点2：两个**one-hot向量**之间彼此正交，因此无法表征单词的相似度

Dense-Vector

- **密集向量(dense vector)**， 其维数 d 的范围较短
- **dense vector**的出现得益于**分布语义(Distributional sem antics)**的思想， 其核心思想是： 一个词的含义是由经常出现在它附近的单词给出的
- 上述想法的具体的构建方法是：
    - 当单词 w 出现在文本中时， 其上下文是出现在附近（在固定大小的窗口内） 的一组单词；
    - 使用 w 的多个上下文来构建 w 的表示。
- 词向量的可视化示例如图：

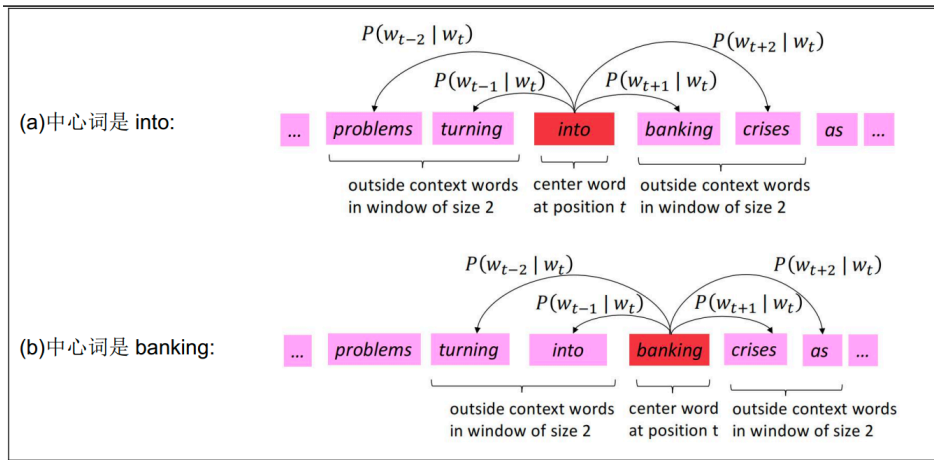
Word meaning as a neural word vector – visualization

$$expect = \begin{pmatrix} 0.286 \\ 0.792 \\ -0.177 \\ -0.107 \\ 0.109 \\ -0.542 \\ 0.349 \\ 0.271 \\ 0.487 \end{pmatrix}$$

Word2Vec

Word2Vec 概览

- **Word2vec (Mikolov et al. 2013)** 是一个学习词向量的框架。 **word2vec** 方法可以快速、 有效地进行训练， 并且可以通过代码和预训练的嵌入轻松在线获得。 Word2vec 嵌入是**静态嵌入(static em beddings)**， 这意味着该方法为词汇表中的每个单词学习一个固定的嵌入。
- **Word2vec** 主要思想是：
    - 我们有大量的文本语料库
    - 固定词汇表中的每个词都由一个向量表示
    - 遍历文本中的每个位置 t， 其中有一个中心词 c 和上下文（“外部”） 词 o
    - 使用 c 和 o 的词向量的相似度来计算 o 给定 c 的概率（反之亦然）
    - 不断调整词向量以最大化此概率
- 下面是计算概率 **P(wt+j|wt)** 的示例窗口和计算过程：



Word2Vec 目标函数

- 目标函数设计如下：

For each position $t = 1, \ldots, T$, predict context words within a window of fixed size m, given center word w_j. Data likelihood:

Likelihood = $L(\theta) = \prod_{t=1}^{T} \prod_{\substack{-m \le j \le m \\ j \ne 0}} P(w_{t+j} \mid w_t; \theta)$

θ is all variables to be optimized

sometimes called a *cost* or *loss* function

The objective function $J(\theta)$ is the (average) negative log likelihood:

$$J(\theta) = -\frac{1}{T} \log L(\theta) = -\frac{1}{T} \sum_{t=1}^{T} \sum_{\substack{-m \le j \le m \\ j \ne 0}} \log P(w_{t+j} \mid w_t; \theta)$$

Minimizing objective function ⟺ Maximizing predictive accuracy

- **Softmax**函数：是一个广义的 **Sigmoid** 函数， 它是一个具有 S 形的指数函数。softmax 函数采用 n 个任意实数值的向量 x = [x1， x2， ...， xn]， 并将其映射到概率分布， 每个值的范围为(0,1)， 所有值的总和为 1。 对于 n 维向量 x，

  softmax 定义为： $softmax(x) = \frac{exp(x_i)}{\sum_{j=1}^{n} exp(x_j)} = p_i$

- **Softmax** 函数的特性是：
  - 将任意值 **xi**映射到概率分布 **pi**；
  - **“max”**， 因为放大了最大 **xi**的概率(但这个名字有点奇怪， 因为它返回一个分布!)
  - **“soft”**， 因为仍然为较小的 **xi**分配一些概率；

Word2Vec 梯度导数

- 若想要$J(\theta)$尽可能小，可以利用梯度下降(**Gradient Descent**)的方法更新参数
- 不过梯度下降**(GD)**每一次更新参数都需要遍历数据集，消耗的时间是难以忍受的，因此使用**SGD(Stochastic Gradient Descent)**，即随机梯度下降来更新参数