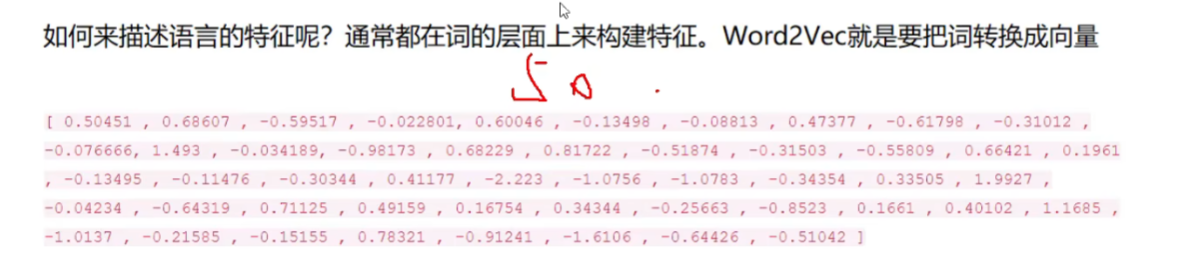
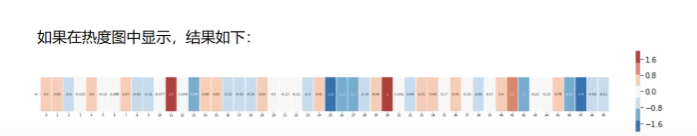
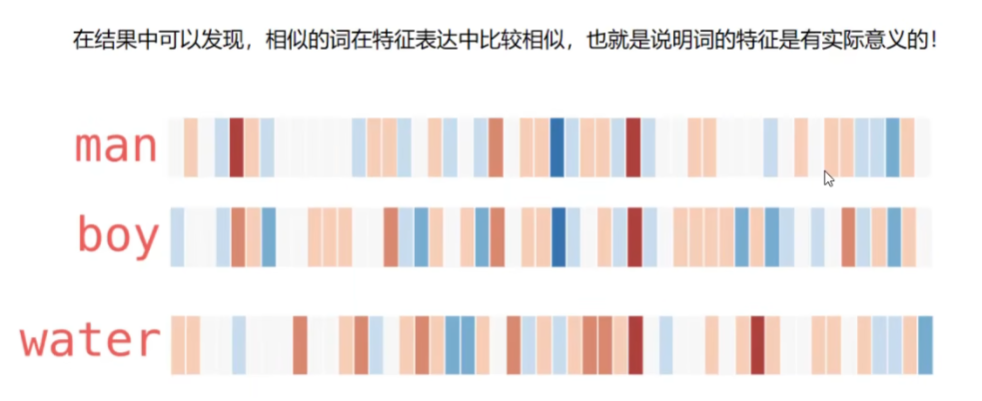
50个维度，50个特征







一个词可能在不同的句里面代表了不同的意思，也可以认为这个词在不同维度的不同概念

怎么利用神经网络去训练词向量呢，输入是什么？输出是什么？也就是我们的训练集是什么？

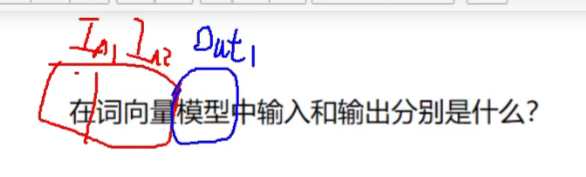
输入：一个字，一个词，一个句子

输出：一个字，一个词，一个句子

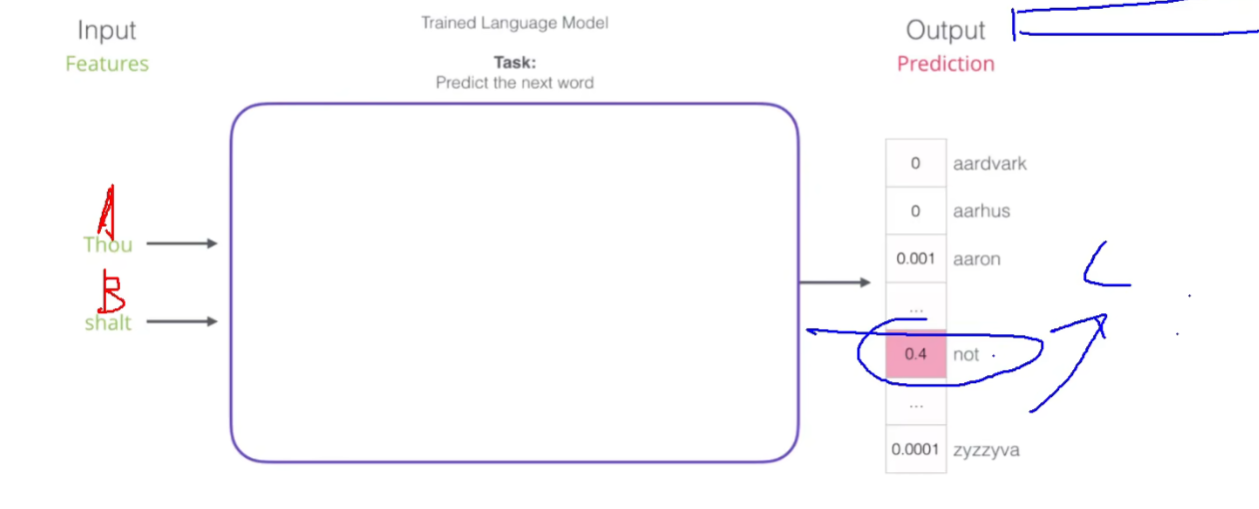
训练如下

[“我”, “是”][“我”, “爱”][“我”, “懂”][“我”, “做”]，所以我们可以理解为输入一词，预测下一个词是什么。

如下图就是训练集

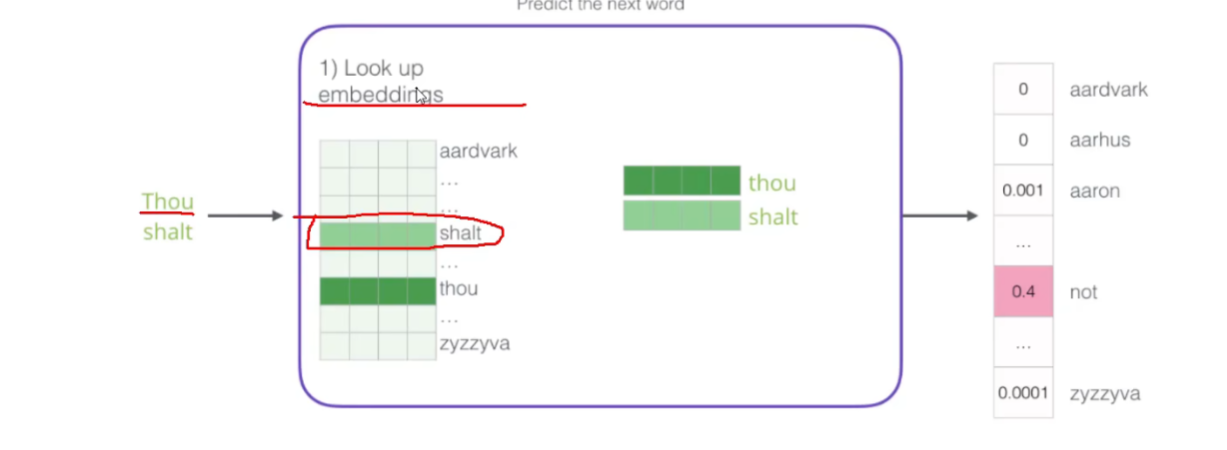


输入A和B，输出not的对应的向量的概率最大，其实就可以看做是多分类任务。

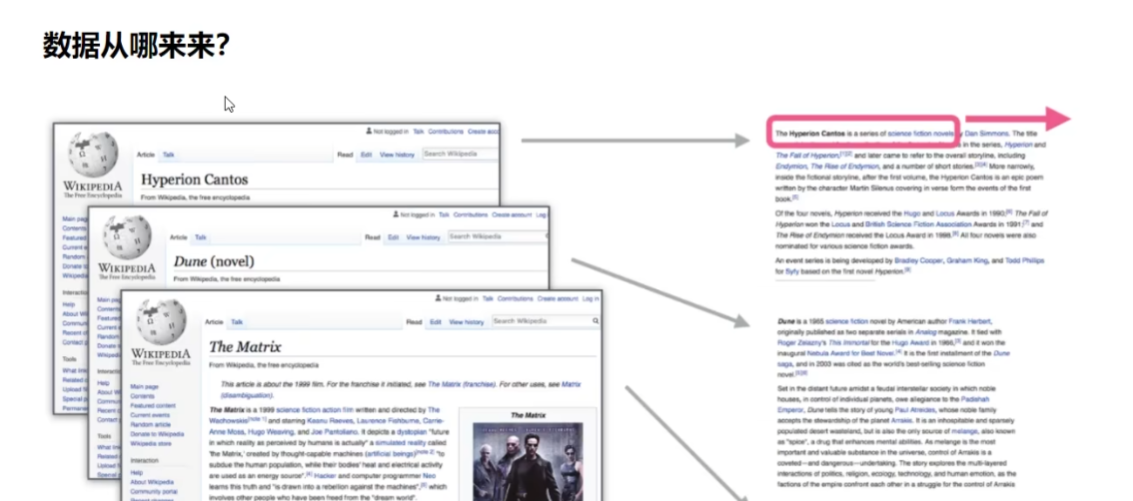


左边是一个词库，随机生成每个词的向量

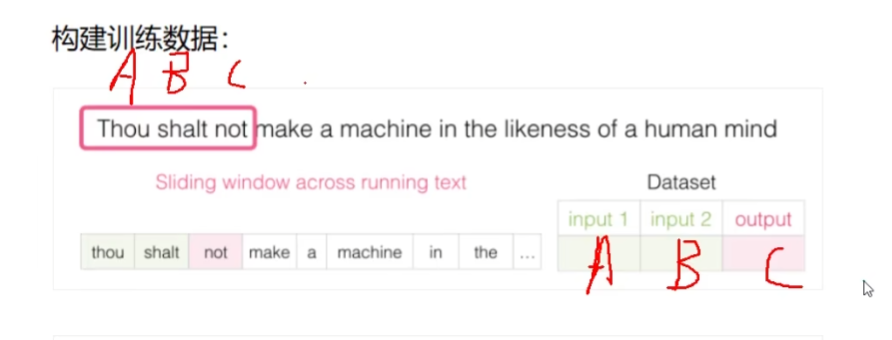
我们的神经网络不只是权重会更新，包括输入的值，也会进行更新，也就是词向量也会进行更新。



数据从哪里来



构建训练数据

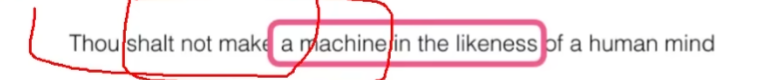


CBOW

根据周围词语推测中间词语

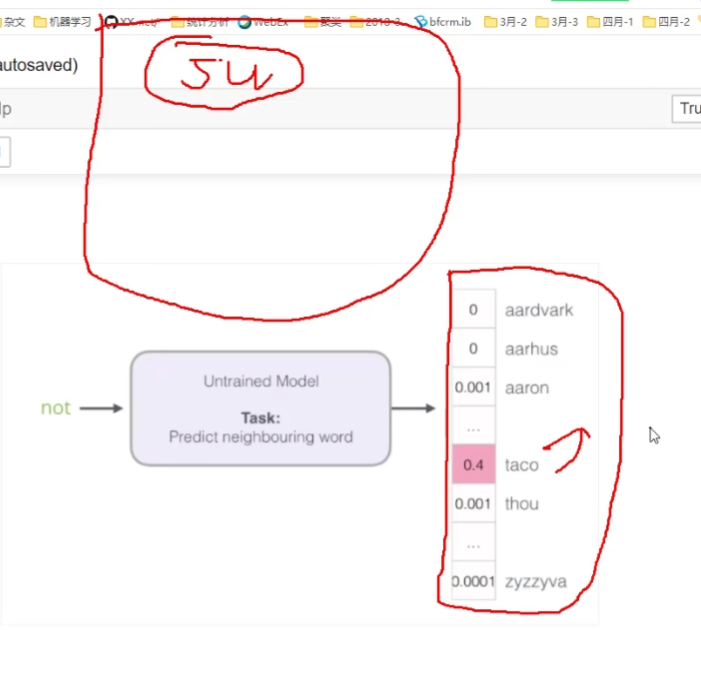
Skipgram

根据中间词推测周围词

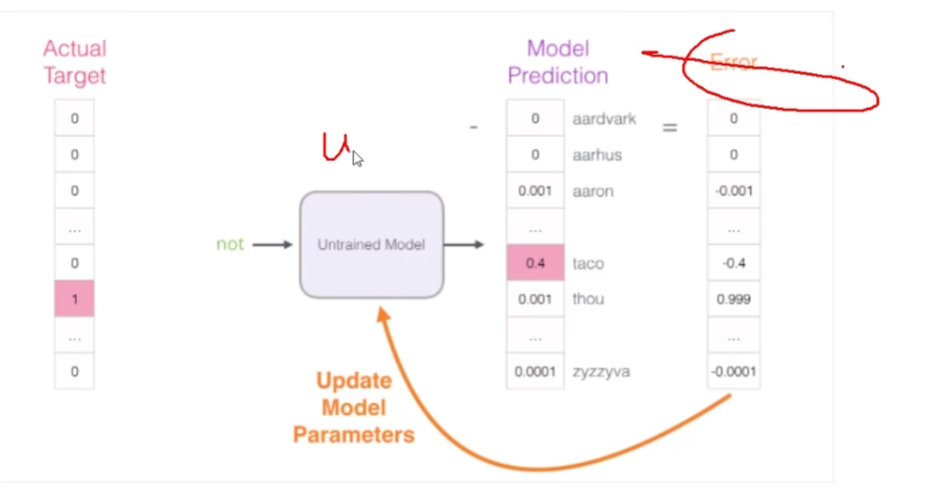


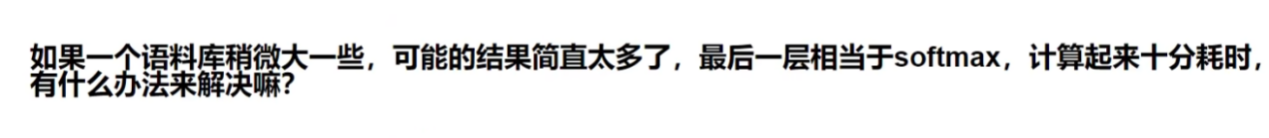


按照上面的学习，假如我们有5w个词，那么我们就有5w个分类的任务，因为我们是找输出的最大概率，看下图右边输出0.4

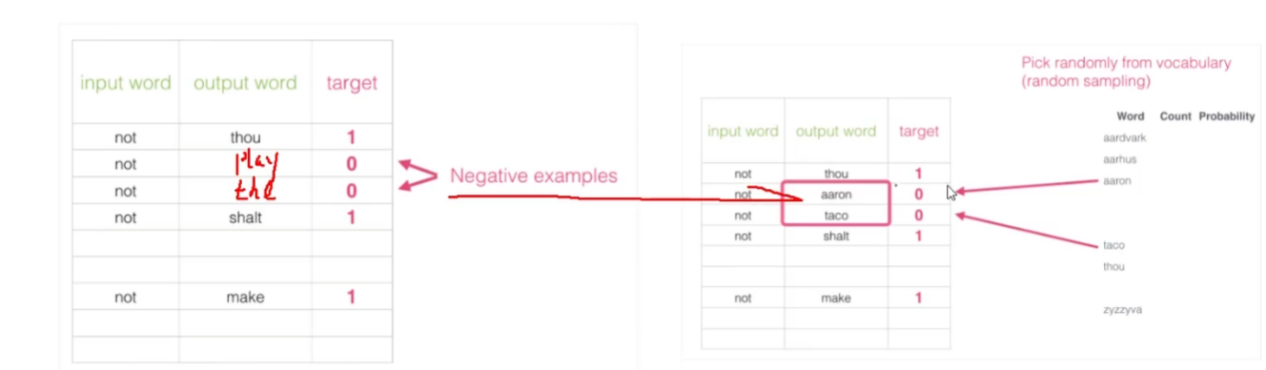


总结

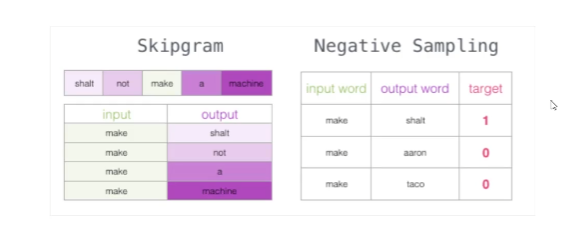




解决办法，将输入和输出同时作为输入，构造很多训练集，相邻的结果为1，同时人为设置负采样，绝对不相邻的设置为0，我们的训练集如下图



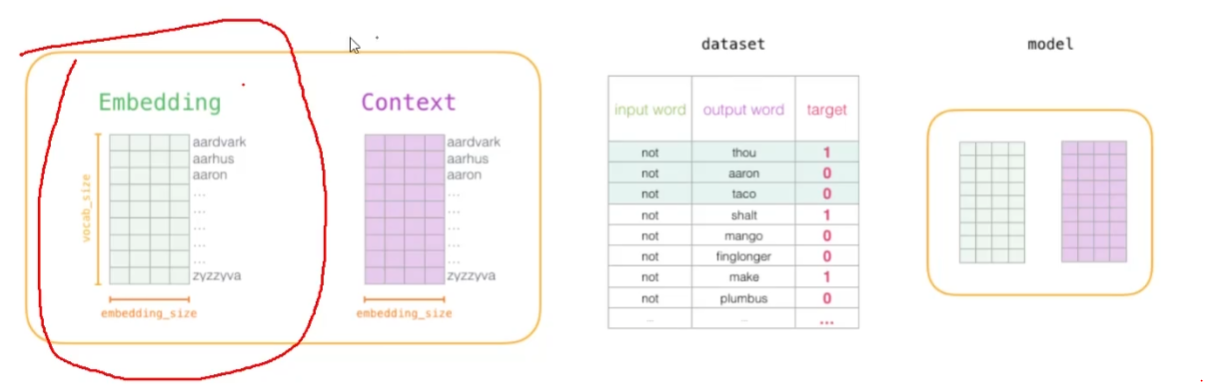
还有



# 训练开始

准备好输入词汇表，输出词汇表，这两个实际上是一样，其实是一张表。

根据句子顺序构造好数据集，如下图



我们举例三个数据集来进行更新

