创 cbow与skip-gram的比较

cbow和skip-gram都是在word2vec中用于将文本进行向量表示的实现方法，具体的算法实现细节可以去看word2vec的原理介绍文章。我们这里大体讲下两者的区别，尤其注意在使用当中的不同特点。

在cbow方法中，是用周围词预测中心词，从而利用中心词的预测结果情况，使用GradientDesent方法，不断的去调整周围词的向量。当训练完成之后，每个词都会作为中心词，把周围词的词向量进行了调整，这样也就获得了整个文本里面所有词的词向量。

要注意的是， cbow的对周围词的调整是统一的：求出的gradient的值会同样的作用到每个周围词的词向量当中去。

可以看到，cbow预测行为的次数跟整个文本的词数几乎是相等的（每次预测行为才会进行一次backpropgation, 而往往这也是最耗时的部分），复杂度大概是O(V);

而skip-gram是用中心词来预测周围的词。在skip-gram中，会利用对周围词的预测结果情况，使用GradientDecent来不断的调整中心词的词向量，最终所有的文本遍历完毕之后，也就得到了文本所有词的词向量。

可以看出，skip-gram进行预测的次数是要多于cbow的：因为每个词在作为中心词时，都要使用周围词进行预测一次。这样相当于比cbow的方法多进行了K-1次（假设K为窗口大小），因此时间的复杂度为O(KV)，训练时间要比cbow要长。

但是在skip-gram当中，每个词都要收到周围的词的影响，每个词在作为中心词的时候，都要进行K次的预测、调整。因此， 当数据量较少，或者词为生僻词出现次数较少时， 这种多次的调整会使得词向量相对的更加准确。因为尽管cbow从另外一个角度来说，某个词也是会受到多次周围词的影响（多次将其包含在内的窗口移动），进行词向量的跳帧，但是他的调整是跟周围的词一起调整的，grad的值会平均分到该词上， 相当于该生僻词没有收到专门的训练，它只是沾了周围词的光而已。

因此，从更通俗的角度来说：

在skip-gram里面，每个词在作为中心词的时候，实际上是 1个学生 VS K个老师，K个老师（周围词）都会对学生（中心词）进行“专业”的训练，这样学生（中心词）的“能力”（向量结果）相对就会扎实（准确）一些，但是这样肯定会使用更长的时间；

cbow是 1个老师 VS K个学生，K个学生（周围词）都会从老师（中心词）那里学习知识，但是老师（中心词）是一视同仁的，教给大家的一样的知识。至于你学到了多少，还要看下一轮（假如还在窗口内），或者以后的某一轮，你还有机会加入老师的课堂当中（再次出现作为周围词），跟着大家一起学习，然后进步一点。因此相对skip-gram，你的业务能力肯定没有人家强，但是对于整个训练营（训练过程）来说，这样肯定效率高，速度更快。