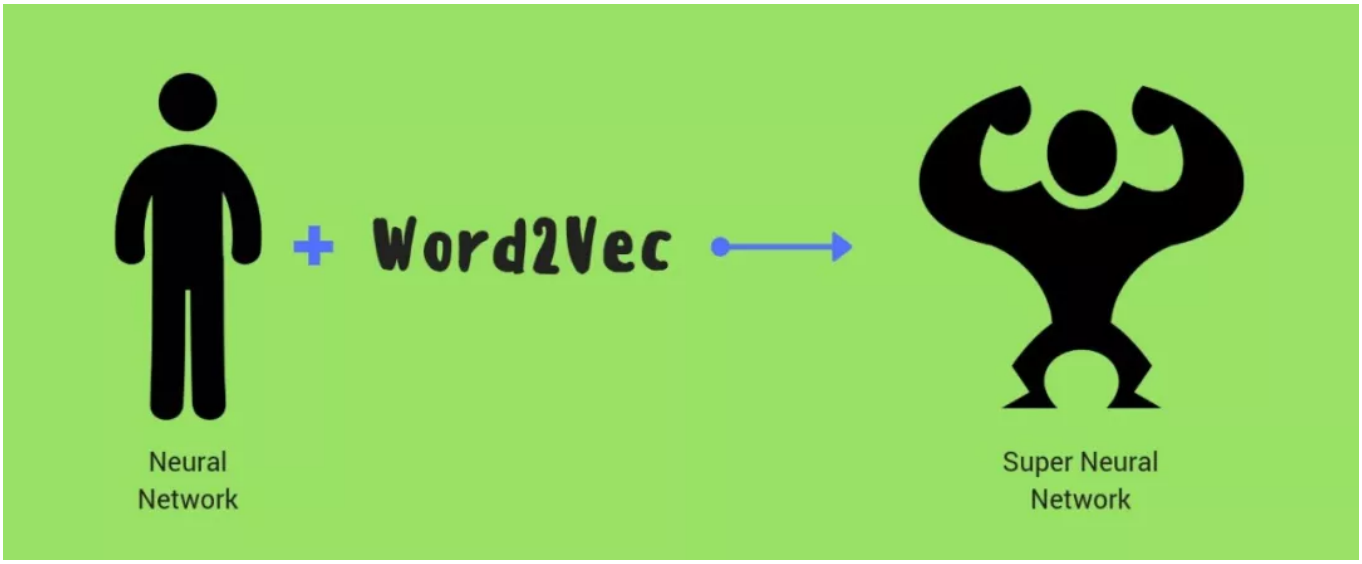
第六篇：Word2Vec的介绍

原创 冯小闲 OOOCaptain 2018-06-17



Word2Vec就是教计算机学会语义信息。

不是很复杂的语义，只是近义词而已。

比方说，【love】和【like】。

如果，计算机能理解近义词，这代表什么呢~~?

这代表——根据我们人类的思维，我们已知like是喜爱的意思，但是我们没学过love。若有人告诉我们这两个词是近义词，那我们自然知道，love这个词也是喜爱的意思。发散性来看，【I like this movie】 和 【I love this movie】，这是两句也几乎一样的意思了。

WHAT：什么是Word2Vec?

1、Word2Vec，就是 convert word to vector，**词语的向量表达方式。**

2、**分布假说：上下文相似的词，其语义也相似。词的语义由其上下文决定。**

- 【I like this movie】中，当Center Word 中心词是【like】时，Context 上下文是【I】，【this】，【movie】这些词。
- 对于【I love this movie】，当中心词是【love】时， 上下文依旧是【I】，【this】，【movie】这些词。
- 从而，计算机可以推断【like】与【love】语义相似的结论。

3、Word2Vec中，把中心词和上下文当做输入和输出，我们就可以这样找到**词与词之间的联系**！上下文相近的词，就会有相似的表达方式~

HOW： Word2Vec的原理

Word2Vec为三层的神经网络结构，分别为输入层Input layer，隐藏层Hidden layer和Output layer输出层。Input layer 和 Output layer 都是 One-hot 的向量。

4、One-hot编码。

One-hot是一种常用的编码方法。上例中， 我们有两个句子“I like this movie”和“I love this movie”， 训练文本包含5个唯一且不重复的单词。这5个单词组成词汇表，如下图，进行One-hot编码。每一个单词向量的维度都是5，都可以被唯一的表示。

I	1	0	0	0	0
like	0	1	0	0	0
love	0	0	1	0	0
movie	0	0	0	1	0
this	0	0	0	0	1

上面的表可以看成一个look-up table 查找表。第一个数值如果是"I"，那么对应的单词就是I，第二个数值对应的是"like"，以此类推。

Word2Vec的输入层和输出层都是One-hot的编码方式，它们都可以对应到单词上。当（Center word，Context）为 （"like"，“I"）时，输入层点亮的是第二个神经元，输出层点亮的是第一个神经元。

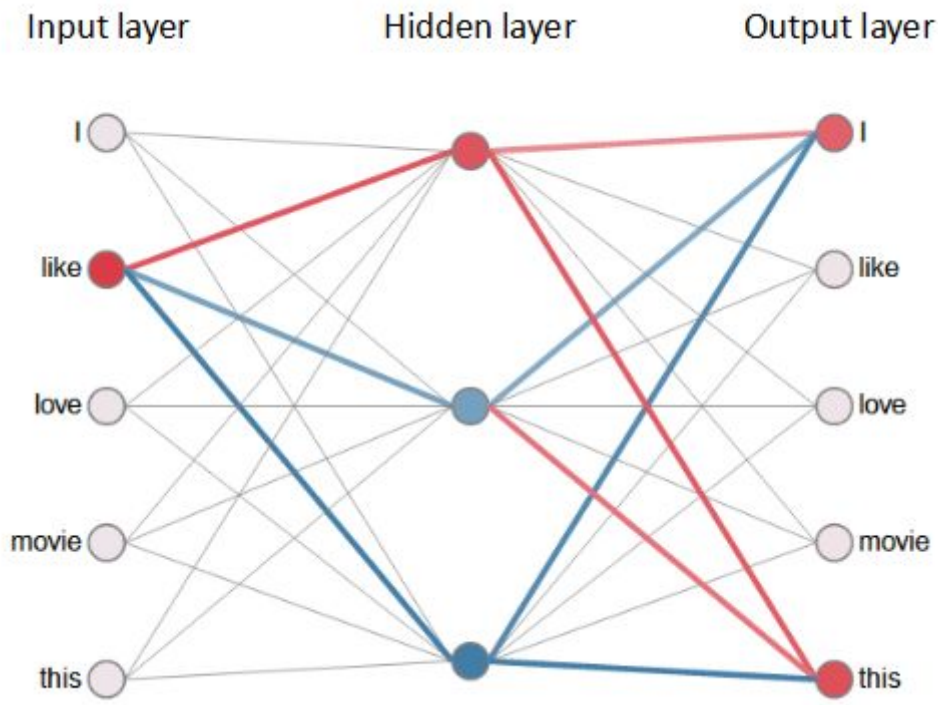
5、三层神经网络。

下图是skipgram model，也就是对一个输入层的输入为中心词，输出层的目标为上下文。它是一对多的模型，一个中心词对应多个上下文词。

神经网络通过学习，训练样本--（Input Word，Output Word）

> （Center Word，Context），比如（"like"，"I"），（"like"，"this"）

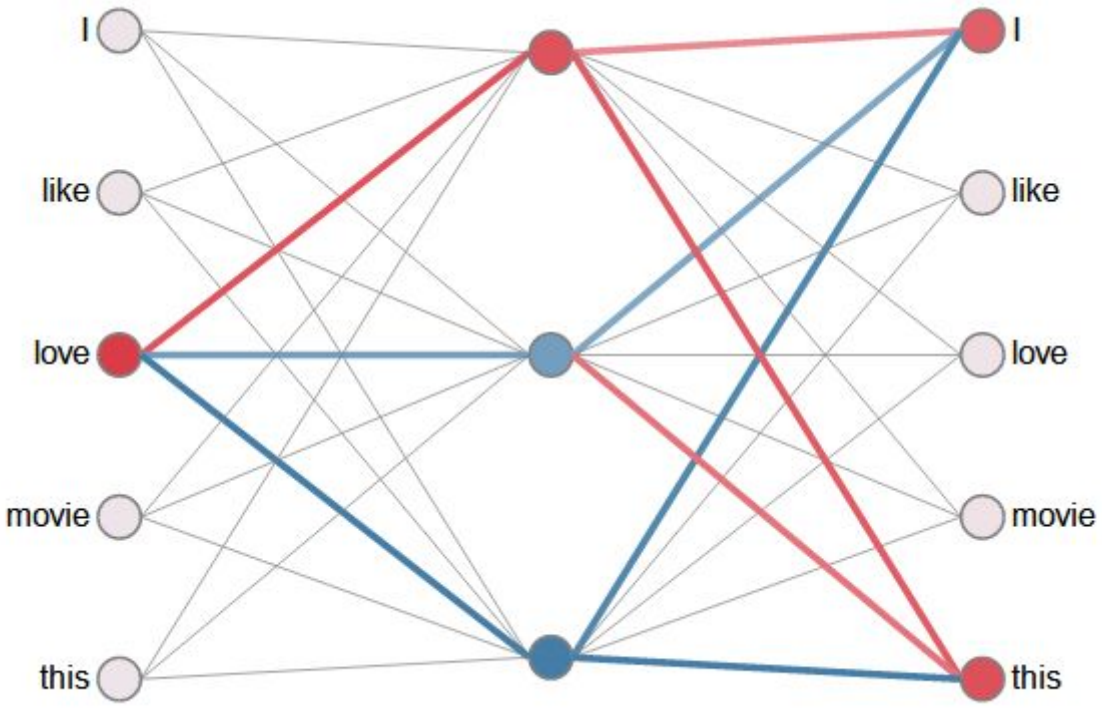
会不断优化隐藏层的权值，以达到目标输出。此时的上下文选取的是中心词的前后一个单词。



要注意的是，也是Word2Vec高级的地方，我们要做的并不是预测，我们要的是一种向量表达方式。也就是说，模型训练好以后，它并不是想要输出，需要的是隐藏层。**隐藏层的权值就是我们要找的"like"这个词的 Word2Vec 向量**!

"love"和"like"的语境一样，上下文都是"I" 和"this"。经过训练，它们的隐藏层的值也几乎一样。所以，它们的 Word2Vec 的向量表达也几乎一样。**语义相近的词，向量表达也是相近的。**

只认识数字向量的计算机，通过识别相似的向量，就这样识别近义词了呗。



总结一下，神经网络做了什么事情呢？

- 得到每一个词的包含语义的向量表达，也就是隐藏层的权值。
- 相比于one-hot的编码方式，降低了向量维度。

  len(Input Layer) = len(Output Layer) > len(Hidden Layer)

Word2Vec就是这样实现的了。

但，它绝不仅仅是这样，Skipgram 这个模型外，还有 CBOW 模型。CBOW模型就是用上下文来预测中心词， 是多对一的模型。

在计算损失函数时还有 Negative Sampling 和 Hierarchical Softmax方法，道阻且长。

WHAT'S MORE:

Word2Vec的应用非常广泛，因为它不仅可以用于解决词语。

Word2Vec提供了降维的思路，One-hot的表达方式在一个大的语料库上维度会变得非常大，但是通过隐藏层，向量的维度就可以被有效的压缩了。广泛运用在NLP深度神经网络之前作为Embedding

Layer。

Word2Vec提供了一种计算关联的思路，通过转化为向量的方式，可以计算出任何term之间的关联度。除了词以外，上述的term还可以被替换为其他任何类型的item，比如book2vec、movie2vec、query2vec等等，只要你有足够的上下文语料去做训练，而现实中这种语料是非常多的，比如一个用户看过/买过/评价过的book、movie，用户搜索过的query。。。

这期到这里就结束啦

好久不见，下期再见~

bye~

————————————The End————————————

这篇文章估计一个月前就开始写了，一直没发出去，都是懒的。。。自我批评中。

对Word2Vec有兴趣的小伙伴们，可以去这个网址动手试一试https://ronxin.github.io/wevi/，该作者Xin Rong的文章也很值得读一读https://arxiv.org/abs/1411.2738，写的非常棒。我当时看了很多paper，tutorial都没理解Word2Vec，直到看到这篇文章，才感觉领悟了那么一点点。

遗憾的是，Xin Rong在去年离开了人世，年轻的生命，在一次飞行意外中丧生。写这篇文章时，才知道这个消息，惋惜。。。

所以说，时间不会等人。

关于这个公众号的一些小事情

公众号叫做**OOOCaptain**，致敬死亡诗社。同时感谢我成长的路上，那些像船长一样引导着我的人~

O Captain！ My Captain!

我接下来的每一篇文章，都是我的成长（主要是在Deep learning的路上）。希望，它们也能变成你的成长~！