# W2V & CNN 应用于文本分类任务

## 目标

情感分析本质上是文本分类任务，目前N-gram模型的算法不能很好地表示句子中多个词语的关系，因此我们想要尝试使用深度学习的模型，比如CNN，RNN等。论文Convolutional Neural Networks for Sentence Classification(论文作者Yoon Kim)即在这一类问题上做了尝试。

## 模型输入

NLP任务的输入不再是像素点了，大多数情况下是以矩阵表示的句子或者文档。矩阵的每一行对应于一个分词元素，一般是一个单词，也可以是一个字符。也就是说每一行是表示一个单词的向量。通常，这些向量都是word embeddings（一种底维度表示）的形式，如word2vec和GloVe，虽然也可以用one-hot向量的形式，即根据词在词表中的索引，但是这种特征对造成向量维度过长的问题。

## 基本模型

CNN模型结构如下：

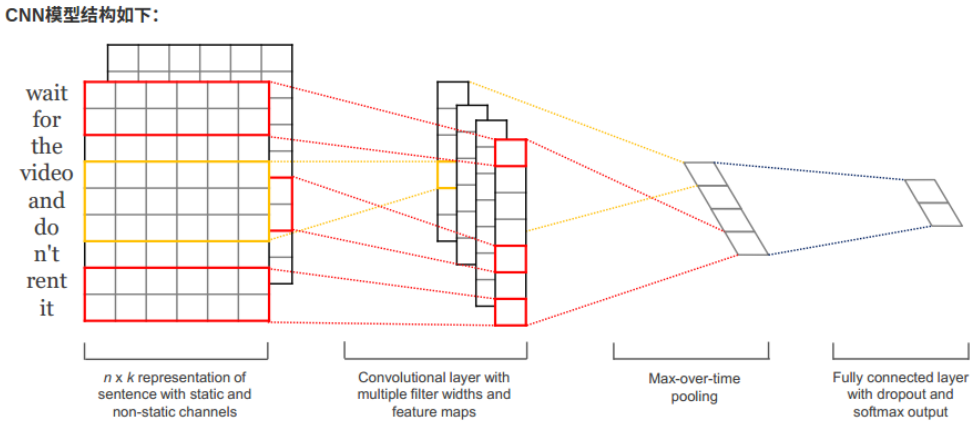


Figure 1: Model architecture with two channels for an example sentence.

### 1. 输入层

如图所示，输入层是句子中的词语对应的wordvector依次（从上到下）排列的矩阵，假设句子有 n 个词，vector的维数为k，那么这个矩阵就是 n × k 的(在CNN中可以看作一副高度为n、宽度为k的图像)。

2. 第一层卷积层

输入层通过卷积操作得到若干个Feature Map，卷积窗口的大小为 h×k ，其中 h 表示纵向词语的个数，而 k 表示word vector的维数。通过这样一个大型的卷积窗口，将得到若干个列数为1的Feature Map。

3. 池化层

文中用了一种称为Max-over-timePooling的方法。这种方法就是简单地从之前一维的Feature Map中提出最大的值，文中解释最大值代表着最重要的信号。可以看出，这种Pooling方式可以解决可变长度的句子输入问题（因为不管Feature Map中有多少个值，只需要提取其中的最大值）。最终池化层的输出为各个Feature Map的最大值们，即一个一维的向量。

4. 全连接+softmax层

池化层的一维向量的输出通过全连接的方式，连接一个Softmax层，Softmax层可根据任务的需要设置（通常反映着最终类别上的概率分布）。

# 训练方案

1. 矩阵的类型:
(1) 静态矩阵(static): word vector是固定不变的, 真实训练的时候对其切断补齐.
(2) 动态矩阵(non static): 在模型训练过程中，word vector也当做是可优化的参数，通常把反向误差传播导致word vector中值发生变化的这一过程称为Fine tune.
(3) Multi-Channel：输入句子时，使用两个通道（channel，可以认为是输入copy一份），都用word2vec初始化，其中一个词的向量保持不变（static），另一个是non-static，在BP过程不断修改，最后再pooling前对两个通道得到的卷积特征进行累加。

2. 未登录词的vector: 用0或者随机小的正数来填充

3. 在倒数第二层的全连接部分上使用Dropout技术，Dropout是指在模型训练时随机让网络某些隐含层节点的权重不工作，不工作的那些节点可以暂时认为不是网络结构的一部分，但是它的权重得保留下来（只是暂时不更新而已），因为下次样本输入时它可能又得工作了，它是防止模型过拟合的一种常用的trikc。同时对全连接层上的权值参数给予L2正则化的限制。这样做的好处是防止隐藏层单元自适应（或者对称），从而减轻过拟合的程度。

4. 在样本处理上使用minibatch方式来降低一次模型拟合计算量，使用shuffle_batch的方式来降低各批次输入样本之间的相关性(在机器学习中，如果训练数据之间相关性很大，可能会让结果很差、泛化能力得不到训练、这时通常需要将训练数据打散，称之为shuffle_batch)。

5. Pooling：使用 max-pooling 获得 feature map 中最大的值，然后使用多个 filter 获得不同 n-grams 的特征。