Text Classification

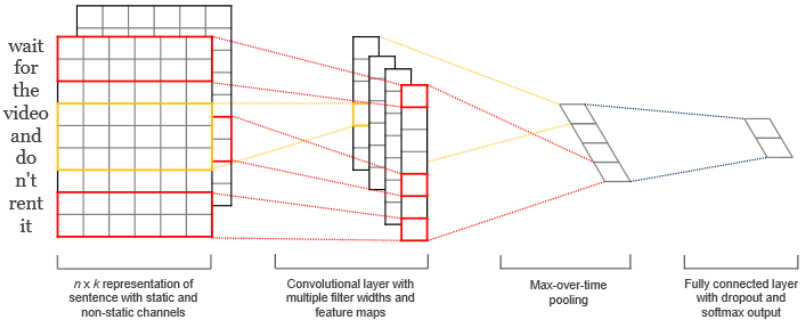
ZZL

2017 年 7 月 14 日

1. Introduction

文本分类是自然语言处理当中一个重要的任务，例如某些情感分析任务也可看作简单的文本分类。传统的文本分类任务主要关注三个方面：特征工程、特征选择、用不同的机器学习算法。传统的特征工程，用的最广的词袋模型；特征选择目的是降噪，比如去掉一些停用词（eg：the）；机器学习算法比如逻辑回归、贝叶斯、支持向量机等。但这些方法都有数据稀疏的问题。这两篇论文在特征工程上选用分布式词向量，用的是 word2vec。这篇文章分别介绍了基于 CNN 与 RCNN 的文本分类模型。

2. 基于 CNN 的分类模型



2.1 模型

$x_i \in R^k$ 表示文章中的第 i 个词的词向量（本文中 k=300）。

$\boldsymbol{x}_{1:n} = x_1 \oplus x_2 \oplus \cdots \oplus x_n$表示有 n 个单词的文本。

$\oplus$是连接符号，将词向量拼接成一个长的文章向量（这里 $\mathbf{x}$ 是一个长向量，图中为矩阵可能是为了便于与图像结合理解）。

定义一个 filter $\mathrm{w} \in R^{hk}$，w作用于 h 个单词，提取一个特征。比如将 filter 作用于第 i 个单词窗口即第 i 个词到 i+h-1 个词，提取出的特征为$c_i = f(w \cdot x_{i:i+h-1} + b)$，b 是偏置（标量），f 是非线性函数（eg：sigmoid，tanh）。

将 w 作用于所有的单词窗口$\{x_{1:h}, x_{2:h+1} \cdots x_{n-h+1:n},\}$产生一组特征$\mathbf{c} = \{c_1, c_2 \cdots c_{n-h+1}\}$,然后我们在 c 上应用池化层（此处为 max 层）取$c' = \max\{\boldsymbol{c}\}$作为这个 filter 作用与这个文本得到的特征。这个模型采用多组 filter，每组 filter 经过池化层后得到一个特征，将这些特征与 softmax 输出层全连接，输出每个类别的概率。

2.2 正则化

为防止过拟合，在全连接层采用 dropout 技术，即以 p 的概率选择某些隐藏单元不参与某次迭代训练，即$\mathrm{y} = \mathrm{w} \cdot (\mathrm{c} \circ \mathrm{r}) + \mathrm{b}$(∘是按位运算符，r 是概率为 p 的伯努利分布产生的 01 向量)，重复多次，直至收敛。在预测阶段，权重矩阵 w 的值缩小为 w*p。

每次训练后，如何 w 的 L2 范数大于 s（文中 s 取 3），那么我们就将 w 缩小使得其 L2 范数为 s。

2.3 预训练

词向量用之前训练好的 word2vec 向量，如果任务中有词没有训练，那么就随机初始化。

2.4 模型变体

（1）CNN-rand

所有词向量都是随机初始化

（2）CNN-static

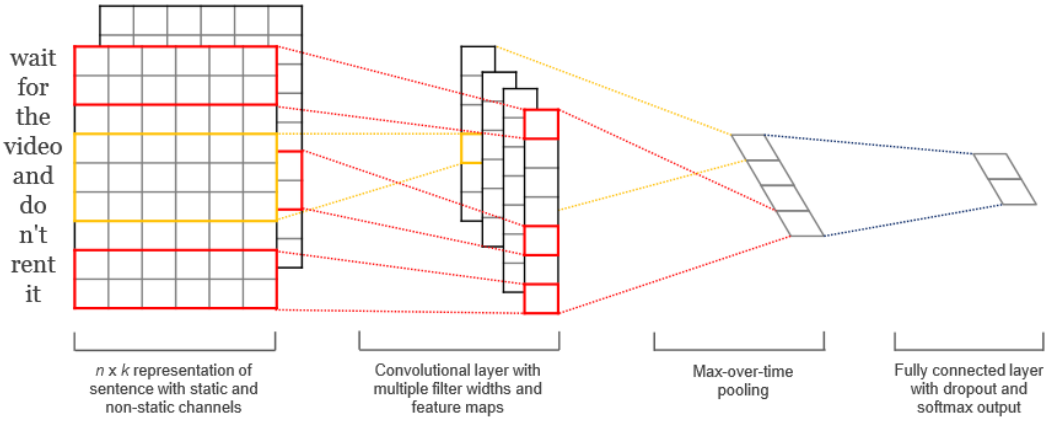
所有词用预先训练出的 word2vec 表示，没有对于向量的词随机初始化。所有词向量在训练过程中保持不变。

（3）CNN-non-static

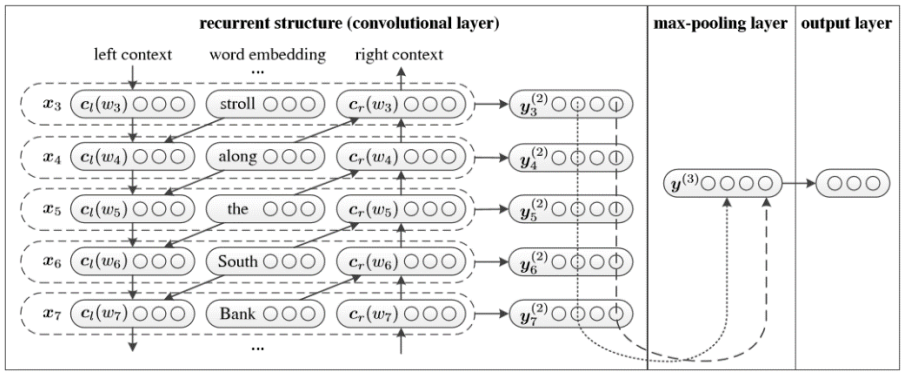
与(2)的不同之处在于，所有词向量当作参数，在训练过程中会有变化。

（4）CNN-multichannel

模型分为两个通道，一个通道用（2）处理，一个通道用（3）处理。（4）的用处在于防止、（3）的过拟合。



3．基于 RCNN 的分类模型



3.1 词语表示学习

这篇文章结合词本身和他的上下文来表示一个词。这篇文章用双向循环神经网络来学习词的左上文信息与右下文信息。

我们用$c_l(w_i)$表示左上文信息，用$c_r(w_i)$表示右下文信息。$c_l(w_i)$与$c_r(w_i)$都是|c|维实向量。用e(w_i)表示词向量本身，是|e|维向量。w_i的信息表示为[$c_l(w_i)$,e(w_i),$c_r(w_i)$]。

下面是$c_l(w_i)$与$c_r(w_i)$的计算方法

$$c_l(w_i) = f(W^{(l)}c_l(w_{i-1}) + W^{(sl)}e(w_{i-1}))$$

$$c_r(w_i) = f(W^{(r)}c_r(w_{i+1}) + W^{(sr)}e(w_{i+1}))$$

这一层的输出为 $\boldsymbol{y}_i^{(2)} = \tanh\left(W^{(2)}\boldsymbol{x}_i + \boldsymbol{b}^{(2)}\right)$

3.2 句子表示学习

这里运用了一个 max 池化层，原文认为文章的信息会集中在少数几个词语上，所以这里用 max 比用 average 好。

$\boldsymbol{y}^{(3)} = \max_{i=1}^{n} \boldsymbol{y}_i^{(2)}$ 这里的 max 作用于每个维度。

池化层之后就是一个传统网络 $\boldsymbol{y}^{(4)} = W^{(4)}\boldsymbol{y}^{(3)} + \boldsymbol{b}^{(4)}$，在 $y^{(4)}$ 上作用 softmax，

$p_i = \frac{\exp\left(\boldsymbol{y}_i^{(4)}\right)}{\sum_{k=1}^{n}\exp\left(\boldsymbol{y}_k^{(4)}\right)}$ 得到每个类别的概率。

参考文献

1. Kim Y. Convolutional neural networks for sentence classification[J]. arXiv preprint arXiv:1408.5882, 2014.
2. Lai S, Xu L, Liu K, et al. Recurrent Convolutional Neural Networks for Text Classification[C]//AAAI. 2015, 333: 2267-2273.