自然语言处理导论

Detecting Sentiment Polarity

实验报告

班级	2014211304
学号	2014210336
姓名	杨炫越
日期	2016.11.14

1. 问题

- Detecting sentiment polarity: 30 points
  - Given text about movie reviews
  - Can we detect sentiment, like whether a comment is
    - Positive?
    - Negative?
  - Can we tell to what extent is a comment positive of negative?
- Data:
  - 5331 positive snippets
  - 5331 negative snippets
- Other resources:
  - The Subjectivity Lexicon

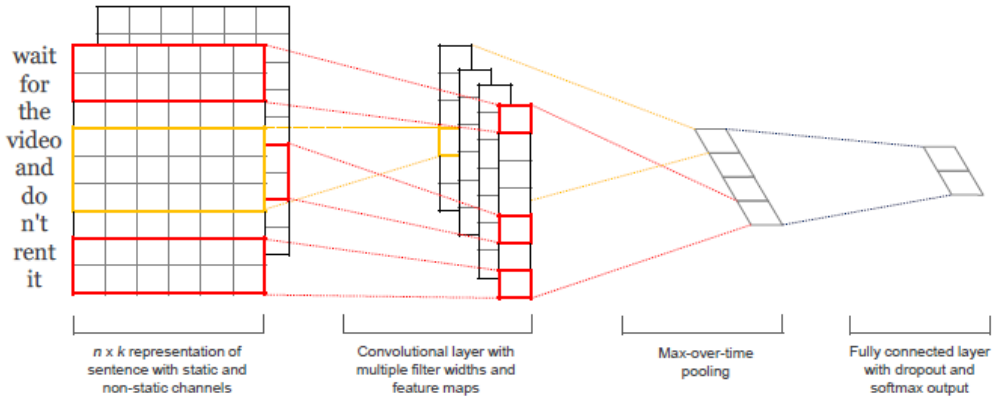
2. 模型

该问题为一个二分类问题, 本实验采用卷积神经网络解决.

2.1. 卷积神经网络(Convolutional Neural Network, CNN)

本实验参考论文[Kim, Yoon. "Convolutional Neural Networks for Sentence Classification." Eprint Arxiv (2014)](以下提到“论文”无其他说明均指该篇), 实现了一个**端到端**的句子分类模型, 主要实验版本未引入先验或预训练数据, 网络参数完全在随机初始化基础上进行学习.

网络结构及各层作用如下:



2.1.1. 嵌入层

该层将输入的每一个单词嵌入到一个低维词向量(所谓低维是相较于词的独热表示而言的), 此向量将作为词在神经网络中的表示形式. 在本程序中该向量通过按高斯分布随机初始化得到, 并随其他网络参数协同训练, 期望可学习得一些词义、上下文约束之类的信息.

具体地, 对句子 $w_1 w_2 \ldots w_l$, 将每个词 w_i 对应的词向量 $\mathbf{x}_i$ 并接而成的矩阵

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, .. \mathbf{x}_l]$$

作为该层的输出.

2.1.2. 卷积层

该层通过将多个不同大小的 filter 与 $\mathbf{X}$ 作卷积, 每个 filter 提取某种局部语义特征, 并输出一个 feature map 代表该特征在句子上的强度分布.

具体地, 对第 i 个 filter $\mathbf{W}_i$, 其与 $\mathbf{X}$ 的卷积加上偏置向量 $\mathbf{b}$, 再通过激活函数 $f(\cdot)$ 得到该 filter 的输出 $\mathbf{c}_i$:

$$\mathbf{c}_i = f(\mathbf{W}_i \otimes \mathbf{X} + \mathbf{b}) .$$

2.1.3. 池化层

该层对每一个 filter 只保留其提取出的强度最大的特征, 使得下一层输入神经元数量得以减少, 只需考虑最重要的特征.

具体地, 对每一 $\mathbf{c}_i$ 作 max-pooling, 即只保留其最大值c_i^*, 将所有最大值串接得句子的特征向量 $\mathbf{z}$:

$$\mathbf{z} = [c_1^*,\ c_2^*, \ldots,\ c_m^*] .$$

2.1.4. 全连接层

该层将 $\mathbf{z}$ 通过一个单层全连接感知器得到在各个分类上的输出数值, 最大数值对应的类别即为该句子的估计分类, 也可将各数值通过 softmax(·) 得到估计的对各类别的概率分布.

具体地, $\mathbf{z}$ 乘以权值矩阵 $\mathbf{W}$ 再加上偏置向量 $\mathbf{b}$, 通过激活函数 $f(\cdot)$ 得到输出 $\mathbf{o}$:

$$\mathbf{o} = f(\mathbf{W}\mathbf{z} + \mathbf{b}) ,$$

再通过 softmax(·) 可得:

$$\Pr\{\hat{y} = i\} = \frac{\exp(\mathbf{o}_i)}{\sum_j \exp(\mathbf{o}_j)} .$$

2.1.5. 罚函数及正则化方法

对该网络采用的罚函数为交叉熵与 L_2 正则化项的和:

$$\mathcal{L} = -(\sum_i \Pr\{y = i\} \log(\Pr\{\hat{y} = i\})) + \lambda \sum_{\mathbf{W}} \|\mathbf{W}\|_2.$$

为减轻过拟合, 除加了 L_2 正则化项之外, 还采取了如下方法:

(1) L_2 范数限制: 原论文中限制全连接层的权值矩阵 $\mathbf{W}$ 的 L_2 范数 $\|\mathbf{W}\|_2$ 不能超过某个阈值, 在完成每轮反向传播后,

若$\mathbf{W}$ 的 L_2 范数超过该阈值, 将其压缩到 L_2 范数等于该阈值.

(2) Dropout: 在池化层与全连接层间加入的 dropout, 按一定概率屏蔽某些神经元的输出, 减少特征间的供适应.

(3) Early Stopping: 当在验证集评估结果达到峰值时即停止训练.

2.2. 结果评估

本实验采用$\text{accuracy} = \frac{\text{number of sentences correctly classified}}{\text{total number of sentences}}$作为模型的评估指标.

另为了表征一个句子分属某个类的“程度”, 即题目中的“tell to what extent is a comment positive of negative”, 可利用模型输出的经 softmax 归一化得到的概率分布. 对于该二分类问题, 假设某个句子被分为正类 1, 可用下式来表征该程度:

$$\frac{\Pr\{\hat{y}=1\}}{\Pr\{\hat{y}=0\}},$$

反之亦然.

3. 实验

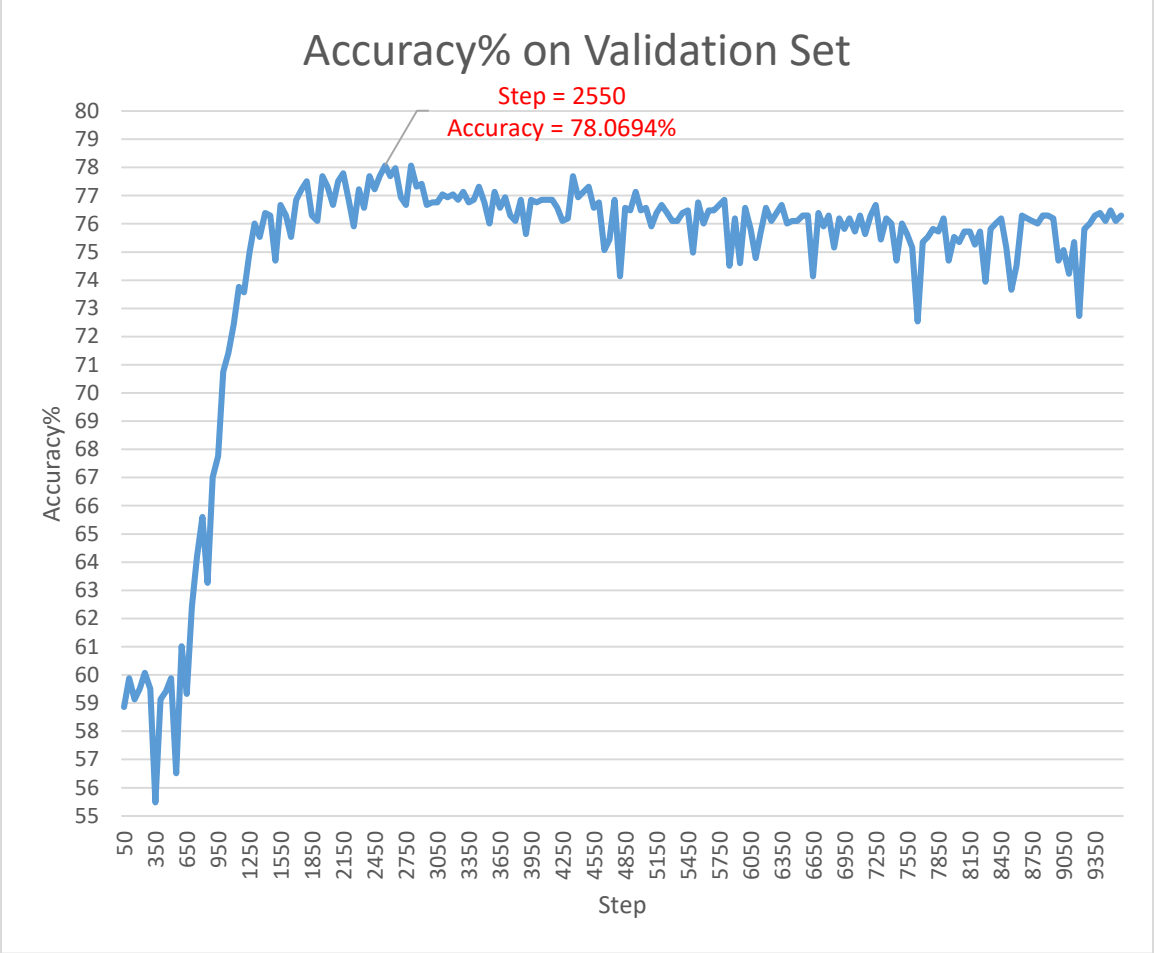
本实验基于 TensorFlow 开发, 采用的训练语料为 MR dataset, 将按 9 : 1 划分为训练集与验证集, 通过训练集训练网络参数, 并定期在采用验证集评估结果.

参考论文原参数及个人调参结果, 最终参数设置如下:

(1) 词向量维度: 300
(2) Filter: 宽度为 $[3, 4, 5, 6, 7]$ 各 100 个(论文中为 $[3, 4, 5]$)
(3) Dropout 保留概率: 0.5
(4) L_2 正则化项系数 λ : 100.0(论文中为 0.0)

(5) L_2 范数限制阈值: 3.0
(6) 训练 Batch 大小: 50

验证集的 accuracy 随训练过程变化如下:



可见在第 2550 个 step 时验证集的 accuracy 达到峰值 **78.07%**, 此结果并不理想, 但是**达到了论文中的相关 baseline** 如下(CNN-rand 即词向量是**随机**初始化的):

Model	MR
CNN-rand	76.1
CNN-static	81.0
CNN-non-static	**81.5**
CNN-multichannel	81.1

4. 分析

由实验结果及论文中的结果可见, 纯**端到端**的词向量+卷积神经网络模型在**该数据集**上表现并不好, 出现了比较明显的**过拟合**问题, 而在该实验中已采取了若干种正则化方法, 经过一番较为枯燥的调参过程, 也仅是达到了 77%~78%的 accuracy, 低于诸多基于统计方法的分类器, 个人猜测主要原因应该是 **MR 数据集过小**.

结合论文及个人想法, 可能改进如下:

(1) 论文中采用**预训练好的**词向量进行 fine-tuning, 可将 accuracy 提升至 **81.5%**, 这也是目前基于神经网络或基于词向量的方法在**该数据集**上取得的较高水平 [Zhang, Ye, and B. Wallace. "A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification." *Computer Science* (2015)]. 由于本实验期望**仅从端到端的学习**得到分类效果, 故 accuracy 难以进一步提升.

(2) 情感词表在该模型中未发挥任何作用, 作为一种先验, 可否通过某种手段将其信息融入网络中？或者可否将由情感词表得到的句子情感分也嵌入到词的向量表示中让网络学习得该项对分类的作用？现已有一系列关于在深度学习中引入先验知识的研究成果, 或可提供参考.

(3) 由同学的启发, 若采用混合分类器可能会有更高的效果.