王瑞环
2100013112

Homework 2

自然语言处理基础
2023 年 4 月 4 日

A 数据预处理与特征提取

Step 1. 数据预处理. 将训练和测试的文本内容去除标点符号, 并将所有字母转换为小写.

Step 2. 构建词汇表. 统计所有训练数据的词频, 将单词按词频排序, 并去除其中的 stopwords. 取前 K 个作为词汇表用于构建特征. 采用的 stopwords 来自 `nltk` 工具包.

Step 3. 特征提取. 对给定的文本, 其特征向量为 one-hot 形式, 向量长度为词汇表大小 K. 遍历其所有词汇, 若某词汇在词汇表中第 j 个位置, 则特征向量的第 j 个位置设置为 1. 在代码实现中, 由于未采用 `numpy/pytorch` 等优化矩阵和向量运算的库, 故采用 `set` 代替 `list` 作为表示特征向量的数据结构以加快运行速度. 集合存储的内容为 one-hot 向量中所有值为 1 的位置的索引.

B Log-Linear 模型

设特征向量维数为 K, 分类数目为 N. 模型参数为权重 $W = \{w_{ij}\}_{\substack{1\le i\le N\\ 1\le j\le K}}$, 初始时随机取值. 记 one-hot 特征向量 $X = (x_1, x_2, \cdots, x_K)^{\mathrm{T}}$ 对应的集合 $S_X = \{j | x_j = 1\}$.

B.1 前向传播

对于输入的文本提取特征 X, 计算其在类别的 $\mathrm{score} = WX$, 也即

$$\mathrm{score}_i = \sum_{j=1}^{K} w_{ij}x_j = \sum_{j\in S_X} w_{ij}, \forall 1 \le i \le N.$$

对 score 应用 softmax 求出文本为第 i 类的概率

$$\hat{p}_i = \frac{\exp\{\mathrm{score}_i\}}{\sum_{t=1}^{N}\exp\{\mathrm{score}_t\}}.$$

$\hat{y} = \arg\max_{1\le i\le N} \hat{p}_i$ 即为所预测的文本类别.

B.2 损失函数与权重更新

B.2.1 交叉熵分类损失

用 c 表示文本的实际类别, $P=(p_1,p_2,\cdots,p_N)$ 为真实的概率分布, 则 $p_c=1$, 其余为 0. 令 $\hat{P}=(\hat{p}_1,\hat{p}_2,\cdots,\hat{p}_N)$ 为预测的概率分布, 取分类损失函数为交叉熵

$$\mathcal{L}_{\text{cat}}=-\sum_{i=1}^{N}\left(p_i\log\hat{p}_i+(1-p_i)\log(1-\hat{p}_i)\right).$$

则

$$\begin{aligned}\frac{\partial\mathcal{L}_{\text{cat}}}{\partial w_{ij}}&=\frac{\partial\mathcal{L}_{\text{cat}}}{\partial\hat{p}_i}\frac{\partial\hat{p}_i}{\partial\text{score}_i}\frac{\partial\text{score}_i}{\partial w_{ij}}=\frac{\partial\mathcal{L}_{\text{cat}}}{\partial\hat{p}_i}\hat{p}_i(1-\hat{p}_i)x_j\\&=\begin{cases}(\hat{p}_i-1)x_j, & i=c\\ \hat{p}_i x_j, & i\neq c\end{cases}\end{aligned}$$

B.2.2 L_2 正则化损失

记正则化系数为 $\dfrac{\alpha}{2}$, 正则化损失为 $\mathcal{L}_{\text{reg}}=\dfrac{\alpha}{2}||W||^2=\dfrac{\alpha}{2}\displaystyle\sum_{(i,j)}w_{ij}^2$. 故

$$\frac{\partial\mathcal{L}_{\text{reg}}}{\partial w_{ij}}=\alpha w_{ij}$$

B.2.3 带动量的权重更新

初始时令动量 $v_{(0)}=0$, 用 $v_{(t)},W_{(t)}$ 表示第 t 次更新后的动量和权重, 并记 $\nabla_t=\dfrac{\partial(\mathcal{L}_{\text{cat}}+\mathcal{L}_{\text{reg}})}{\partial W_{(t)}}$.

设动量的更新权重为 β, 学习率为 γ. 则第 t 次权重更新过程为

$$v_{(t)}\leftarrow\beta v_{(t-1)}+\gamma\nabla_{t-1},\quad W_{(t)}\leftarrow W_{(t-1)}-v_{(t)}.$$

C 实验设置与结果

实验使用的环境为

```
Python=3.7, pandas=1.3.5, scikit-learn=1.0.2, regex-2022.10.31
```

实验中各个超参数的设置见表1.

超参数	特征向量维度 K	学习率 γ	正则化系数 α	动量更新权重 β	训练 Batch Size
值	60000	1×10^{-3}	1×10^{-3}	0.99	200

表 1: 超参数取值表.

训练轮数为 10 轮, 用时约 7min. 实验结果见表2, accuracy 曲线如图1.

指标	Accuracy	Macro-F1	Micro-F1
训练集	99.96%	99.96%	99.96%
测试集	78.69%	78.69%	78.28%

表 2: 超参数取值表.

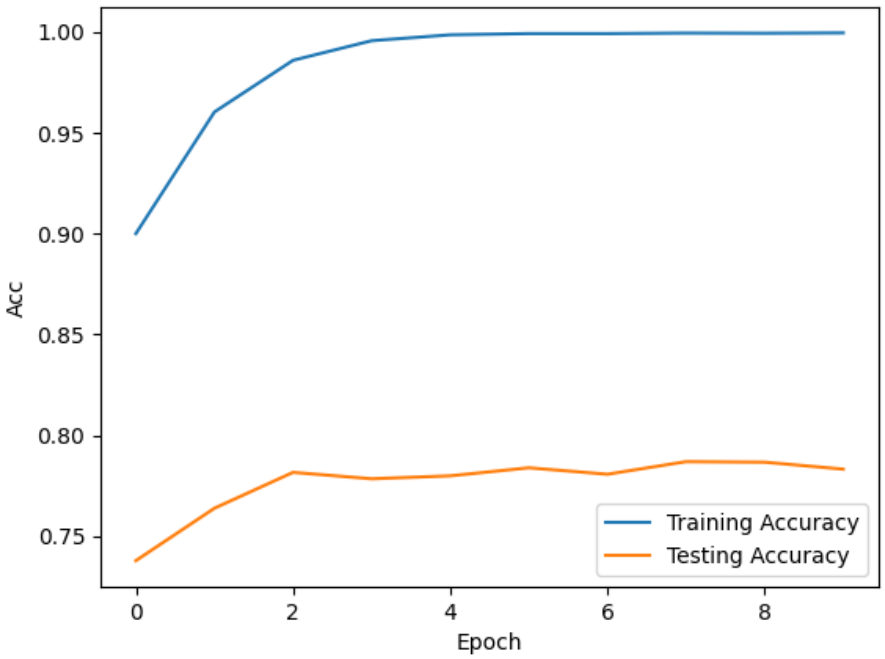

图 1: Accuracy 曲线