LabelSmoothing原理

2021 年 6 月 26 日

Huacheng Li

1 背景介绍: OneHot 编码

神经网络的输出成为Logits，间记为z，经过Softmax后转化为和为1的形式，记为$\hat{y}$，真实的target记为y，K为总的分类的类别的数量。

Logits

logits是指一件事情发生与不发生的比值的对数。假定时间发生的概率为p，那么该事件的logits为

$$\text{logits}(p) = \log \frac{p}{1-p}$$

在深度学习中，softmax会对输入进行归一化处理，则第i个类的概率

$$p(i) = \frac{\exp(a_i)}{\sum_{j \in K} \exp(a_j)}$$

$\{a_0, a_1, \ldots, a_K\}$就是logits。因此logits可以理解为为归一化的概率

当损失函数为交叉熵，且target的编码和为1时，导数为$\hat{y}_i - y_i$。

求导过程

根据softmax公式和交叉熵公式，p_i为预测概率，q_i是真实概率

$$p_i = \text{softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j\in K}\exp(z_j)}$$

$$Loss = -\sum_{i\in K} = q_i \log(p_i)$$

求导：1. 当分类$i == k$时：

$$\begin{aligned}\frac{\partial p_i}{\partial z_k} = \frac{\partial p_i}{\partial z_i} = \frac{\partial \frac{\exp(z_i)}{\sum_{j\in K}\exp(z_j)}}{\partial z_i} &= \frac{\exp(z_i)\sum_{j\in K}\exp(z_j) - \exp(z_i)\frac{\partial \sum_{j\in K}\exp(z_j)}{\partial z_i}}{[\sum_{j\in K}\exp(z_j)]^2} \\ &= \frac{\exp(z_i)}{\sum_{j\in K}\exp(z_j)} - \frac{[\exp(z_i)]^2}{[\sum_{j\in K}\exp(z_j)]^2} = p_i - p_i^2\end{aligned}$$

2. 当$i \neq k$时：

$$\begin{aligned}\frac{\partial p_i}{\partial z_k} = \frac{\partial \frac{\exp(z_i)}{\sum_{j\in K}\exp(z_j)}}{\partial z_k} &= \frac{0\times\sum_{j\in K}\exp(z_j) - \exp(z_i)\frac{\partial \sum_{j\in K}\exp(j)}{\partial z_k}}{[\sum_{j\in K}\exp(z_j)]^2} \\ &= -\frac{\exp(z_i)\exp(z_k)}{[\sum_{j\in K}\exp(z_j)]^2} = -p_i \times p_k\end{aligned}$$

3. 交叉熵求导

$$\frac{\partial Loss}{\partial p_i} = -q_i\partial\sum_{j\in K}\log(p_i) = -q_i\partial\log(p_i) = -q_i \times \frac{1}{p_i} = -\frac{q_i}{p_i}$$

4. 对logits求导

$$\begin{aligned}\frac{\partial Loss}{\partial z_i} &= \frac{\partial Loss}{\partial p_i}\times\frac{\partial p_i}{\partial z_i} + \sum_{j\neq i}(\frac{\partial Loss}{\partial p_j}\times\frac{\partial p_j}{\partial z_i}) = -\frac{q_i}{p_i}\times(p_i - p_i^2) + \sum_{j\neq i}(\frac{q_j}{p_j}\times p_j \times p_i) \\ &= -q_i\times(1-p_i) + \sum_{j\neq i}(q_j\times p_i) = \sum_{j\in K} = -q_i + q_i\times p_i + \sum_{j\neq i}(q_j\times p_i) \\ &= \sum_{j\in K}(q_j\times p_i) - q_i = p_i\sum_{j\in K}(q_j) - q_i = p_i - q_i\end{aligned}$$

假定总共有K个类，则可以如下表示：

$$\begin{cases}\hat{y}_i = \frac{\exp(z_i)}{\sum_{j\in K}\exp(z_j)} \\ \frac{\partial l}{\partial z_i} = \hat{y}_i - y_i\end{cases} \tag{1}$$

根据真实情况，我们应当令正确类$\hat{y}_{true} = 1$，错误类$\hat{y}_{false} = 0$，因此可以写为以下两式：

$$\begin{cases}\hat{y}_{true} = \frac{\exp(z_{true})}{\sum_{j\in K}\exp(z_j)} = 1 \\ \hat{y}_{false} = \frac{\exp(z_{false})}{\sum_{j\in K}\exp(z_j)} = 0\end{cases} \tag{2}$$

根据公式(2) 可以得到下式：

$$\begin{aligned}\exp(z_{true}) = \exp(z_{true}) + \sum_{j\neq true}\exp(z_j) \\ \rightarrow \sum_{j\neq true}\exp(z_j) = 0 \\ \rightarrow \exp(z_j)_{j\neq true} = 0 \\ \rightarrow z_{false} = -\infty\end{aligned} \tag{3}$$

Conclusion

因此在target为one-hot编码，损失函数为交叉熵的情况下，$z_{true} \to C, z_{false} \to -\infty$。这表示错误类的logits为负无穷，正确类的logits为常数。这种最有情况一般是不能达到的，且$z_{true} \gg z_{false}$。根据[Szegedy et al., 2016]观点，这种情况下会出现两个非常不好的性质：

- 导致过拟合，将所有概率都赋值给了真值，泛化能力下降
- 要求真值对应的logits要远远大于其他值的logits，但导数$\frac{\partial l}{\partial z_i}$是有界的，也就是数值不会很大。这意味着要更新很多次

2 LabelSmoothing

LabelSmoothing是[Szegedy et al., 2016]提出的。作者应该是认为：蒸馏改变了学习的真值，为了能够获得更好的结果，但是需要准确率更高的教师网络；如果现在想要训练出一个准确率最高的模型，要是没有网络能给我知识，所以就通过LabelSmoothing学习一种简单的知识。

LabelSmoothing 的编码形式如下式所示，其中ϵ是超参数，一般取值为0.1

$$y_i = \begin{cases} 1-\epsilon & if \quad i == true \\ \frac{\epsilon}{K-1} & otherwise \end{cases} \tag{4}$$

对公式(4)求导，类似公式(2)我们可以得到下式

$$\begin{cases} \frac{\exp(z_{ture})}{\exp(z_{true}+\sum_{j\neq true}\exp(z_j))} = 1-\epsilon \\ \frac{\exp(z_{false})}{\sum_{j\in K}\exp(z_j)} = \frac{\epsilon}{K-1} \end{cases} \tag{5}$$

因为正确的类只有1个，错误的类有$K-1$个，且在解析解的情况下，错误类的概率近乎相等。因此可以得到下式：

$$\begin{aligned} \exp(z_{true}) &= (1-\epsilon)\exp(z_{true}) + (1-\epsilon)(K-1)\exp(z_{false}) \\ &\to \epsilon\exp(z_{true}) = (1-\epsilon)(K-1)\exp(z_{false}) \\ &\to z_{true} = \log(\frac{(K-1)(1-\epsilon)}{\epsilon}) + z_{false} \end{aligned} \tag{6}$$

可以令z_{false}为α，那么在导数等于0的情况下，logits的取值为：

$$z_i^* = \begin{cases} \log(\frac{(K-1)(1-\epsilon)}{\epsilon}) + \alpha & if \quad i = y \\ \alpha & otherwise \end{cases} \tag{7}$$

Conclusion

One-Hot编码需要错误类的logits趋向于负无穷，这样会导致正确类和错误类的输出误差很大，网络泛化能力不强。并且因为网络训练时一些正则化的存在，logits的输出很难是负无穷的。LabelSmoothing编码方式只要正确类和错误类有一定的数值误差即可。

References

[Szegedy et al., 2016] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society.