

# 语言模型实践报告

— 饶翔云,520030910366

项目地址: <https://github.com/xyrrrrrrr/languagemodel> (<https://github.com/xyrrrrrrr/languagemodel>)

## 0. 原理分析

代码主要部分在 `main.py` 文件中, 即通过不同的模型 (RNN、GRU、LSTM、Transformer等) 完成所给语句的训练, 并在测试集上测试的流程。训练集的语句输入后首先通过输入的编码层转化为词向量, 进而输入到神经网络中进行训练, 最后输出为某个词的似然概率。在测试集上输入语句后经过神经网络得到输出语句每个词的似然概率, 从而得到输出语句预测成功的概率, 并从而计算PPL值。模型在测试集上的评测指标由loss和PPL构成, 其中PPL (困惑度) 的计算公式为:

$$PPL = \exp - \frac{1}{N} \sum_{i=1}^N \log P(w_i|h)$$

测试集上PPL的值越小, 说明模型的训练效果越好。

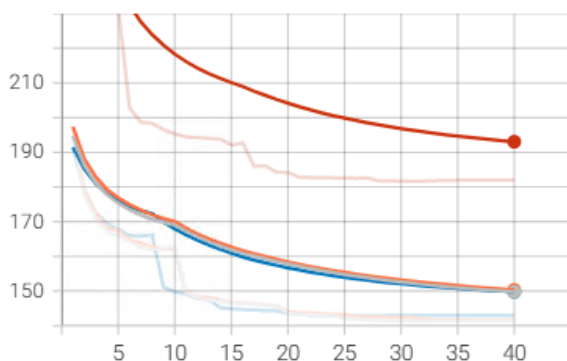
除了 `main.py` 文件, 该项目还有其他的文件: `model.py` 文件中定义了不同的模型, 在 `main.py` 的训练过程中调用; `generate.py` 文件用于从已经训练好的语言模型中采样得到一些新产生的句子; `data` 文件夹是用于训练 (train)、验证 (valid)、测试 (test) 的数据。

## 1. 模型架构

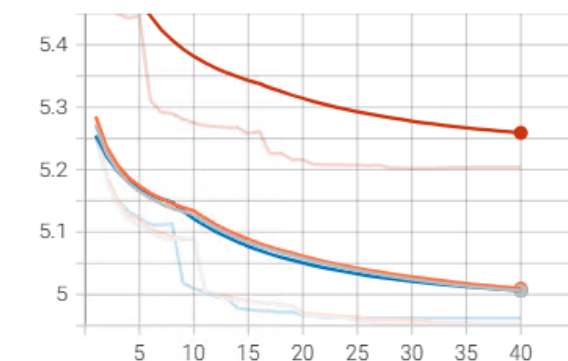
我选择使用LSTM,LSTM-tied,GRU和Transformer。性能比较如下:

test

ppl  
tag: test/ppl



total\_loss  
tag: test/total\_loss



四者的数据比较: (使用超参数均为默认值)

test	Transformer	LSTM	LSTM-tied	GRU
PPL	182	141.7	<b>141.5</b>	142.9
Total loss	5.204	4.954	<b>4.952</b>	4.962

可以看到，在默认的超参数下，LSTM-tied表现最好。

## 2. 实验过程

对nlayers进行探究，其余配置保持默认：

test	1	2	6
PPL	<b>5.05</b>	5.11	5.25
Total Loss	<b>155.43</b>	164.97	191.43

可以看到，当层数增大，PPL反而增大，这意味着网络层数过多会导致LSTM网络结构的拟合效果变差，只需要一层网络结构便能达到较好的运行结果。在这里采取默认的两层网络作为默认，原因是保证较高的模型复杂度以便于后续实验。

对序列长度进行探究，其余配置保持默认：

test	20	35	50
PPL	5.17	5.11	<b>5.07</b>
Total Loss	175.89	164.97	<b>159.38</b>

可以看到，当输入网络序列长度增加，loss和PPL降低，模型性能变好，即在训练时令网络拟合更长的序列可以使其学习更多的信息，对上下文的覆盖范围更大，即在训练时能够更好地根据上下文预测出正确的单词，即网络性能更好。

对embed\_size和hidden\_size研究：

test	100	650	850
PPL	141.5	127.66	<b>122.22</b>
Total loss	4.952	4.85	<b>4.81</b>

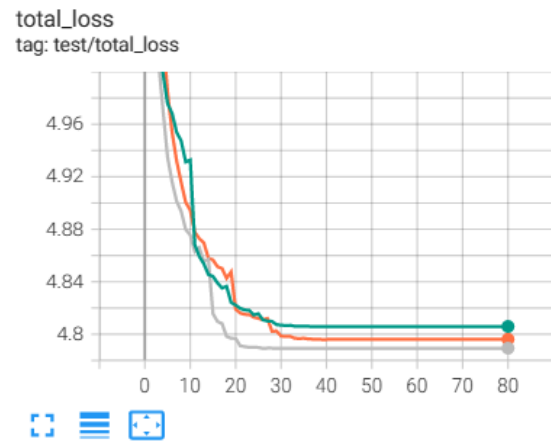
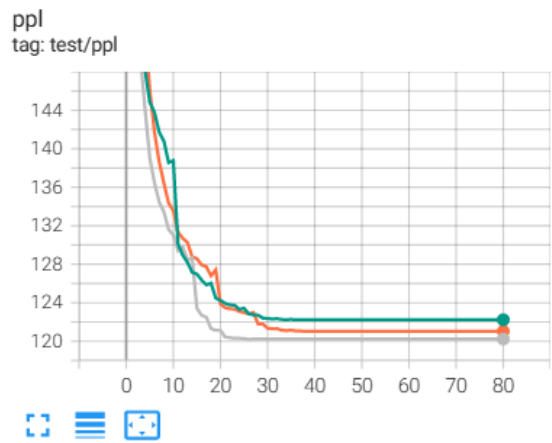
通过测试可以发现，在满足模型参数不超过60m的情况下，这类参数越大越好。我认为是因为这类参数增大增加了模型的复杂度使得模型有更强的识别能力。

确定了embed\_size和hidden\_size后，对学习率进行探究：

test	20	10	5
PPL	122.2	<b>120.2</b>	121
Total loss	4.806	<b>4.789</b>	4.796

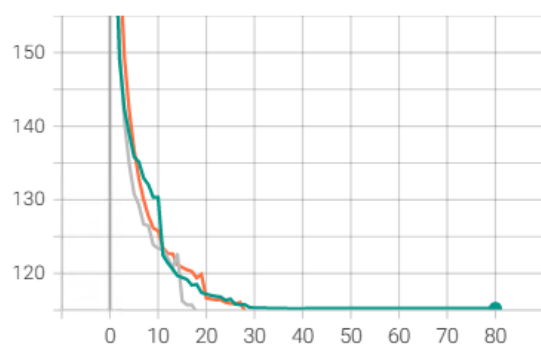
可以看到，初始学习率为10时效果最好。

test

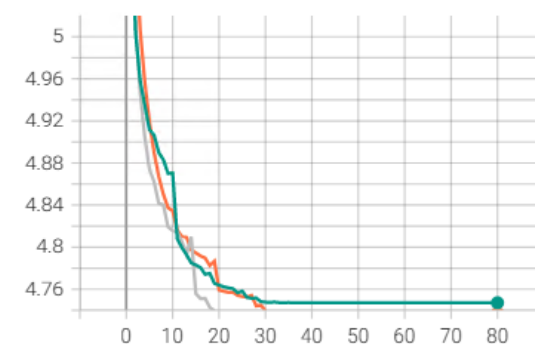


val

ppl  
tag: val/ppl

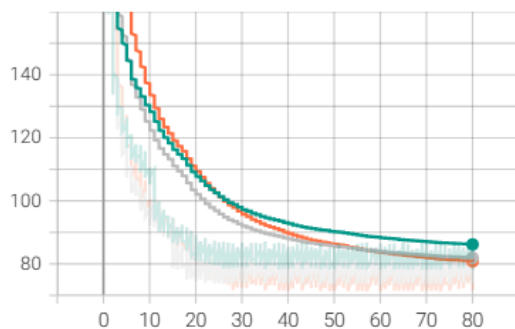


total\_loss  
tag: val/total\_loss

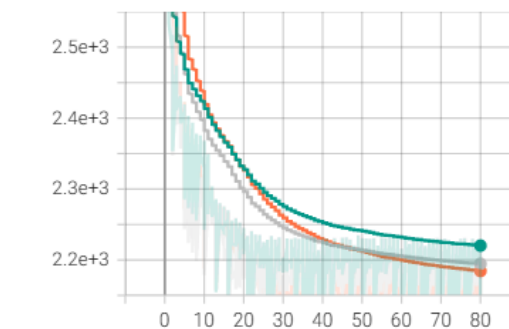


train

ppl  
tag: train/ppl



total\_loss  
tag: train/total\_loss



### 3. 最佳配置

调节epoch为80，embed\_size为850和hidden\_size为850，lr为10，bptt为50，其余参数不变，模型参数大小为58.77M。可以跑run.sh来复现。

### 4. 实验总结

本次实验通过一个设计好的训练语言模型的框架，让我们重新复习巩固了上智能语音识别课时学习的对语言模型的设计，训练，测试，调参过程，让我们对超参数调参有了更直观的认识，在之后的实践项目中调参更有方向性。