计14 汪隽立 2021012957

# ANN-hw3

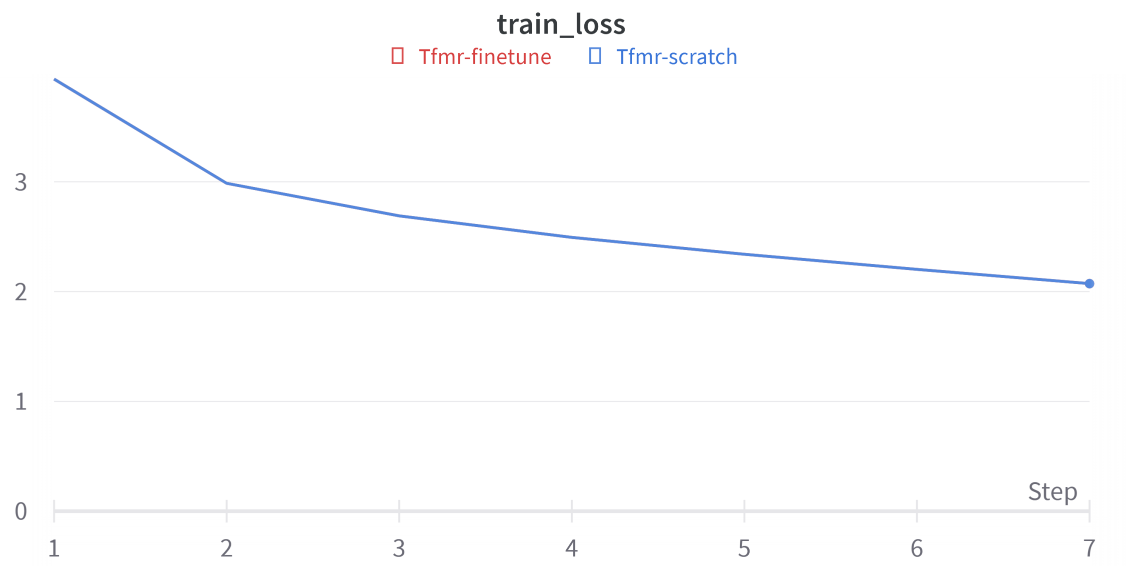
如果没有特殊声明，训练中使用的超参为：

```
{
  "attn_pdrop": 0.1,
  "bos_token_id": 50256,
  "embd_pdrop": 0.1,
  "eos_token_id": 50256,
  "initializer_range": 0.02,
  "layer_norm_epsilon": 1e-05,
  "n_ctx": 35,
  "n_embd": 768,
  "n_head": 12,
  "n_layer": 3,
  "n_positions": 1024,
  "resid_pdrop": 0.1,
  "vocab_size": 50257
}
```

## 1.1 Loss & PPL

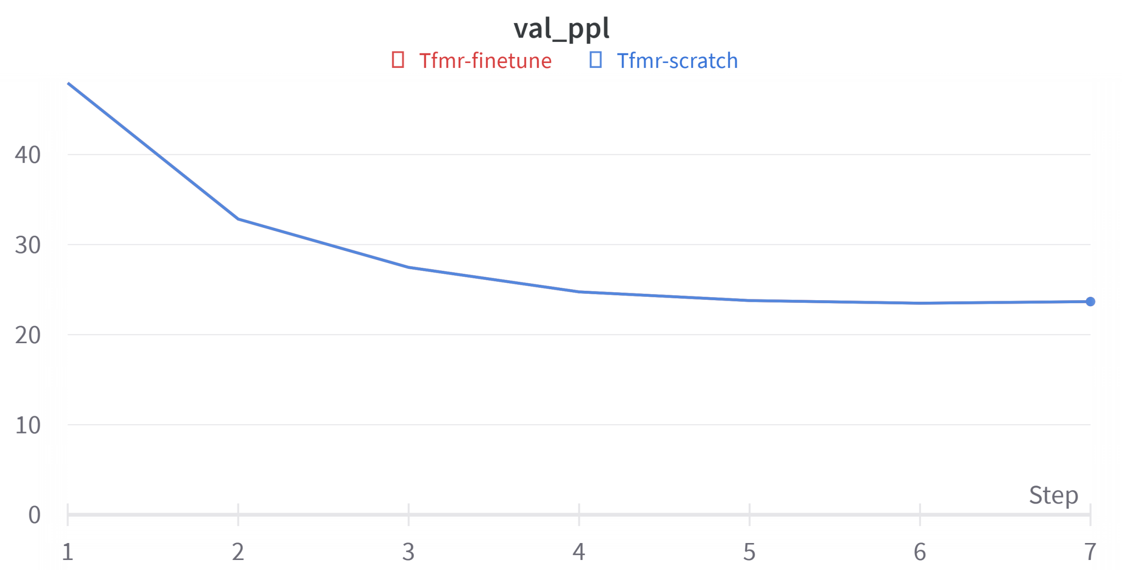
### Tfmr-scratch

- train_loss:




- val_loss:

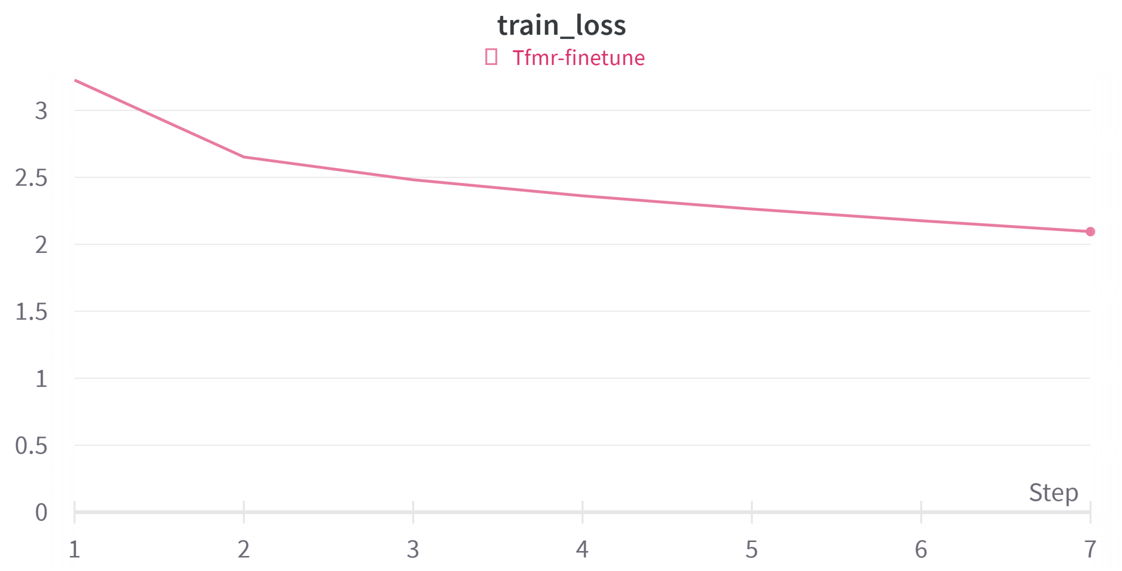

- val_ppl:



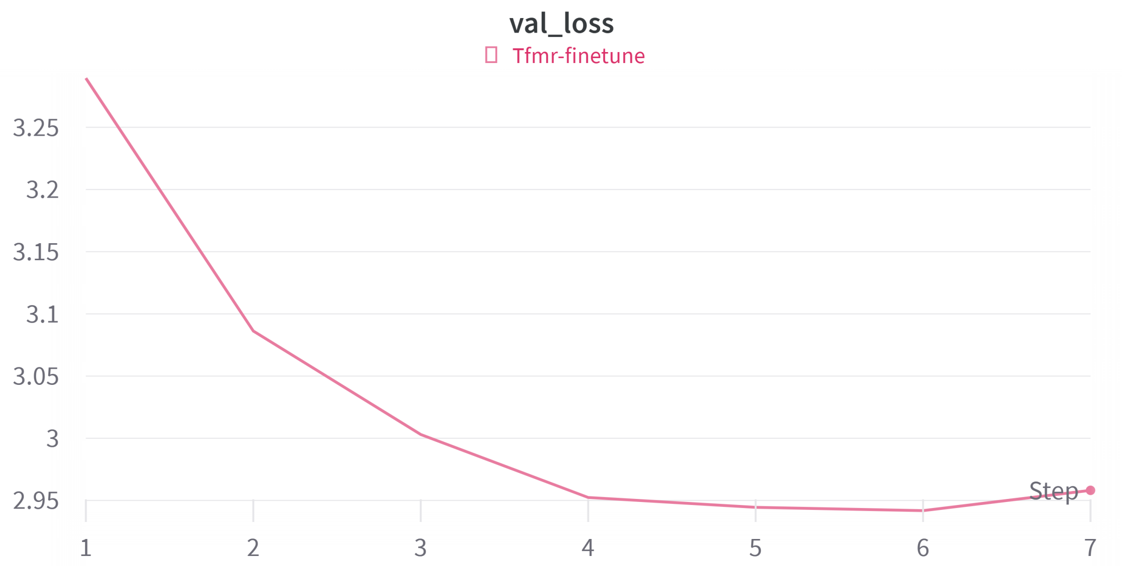

在实验中，一般进行到第7-8个epoch就会停止。

## Tfmr-finetune

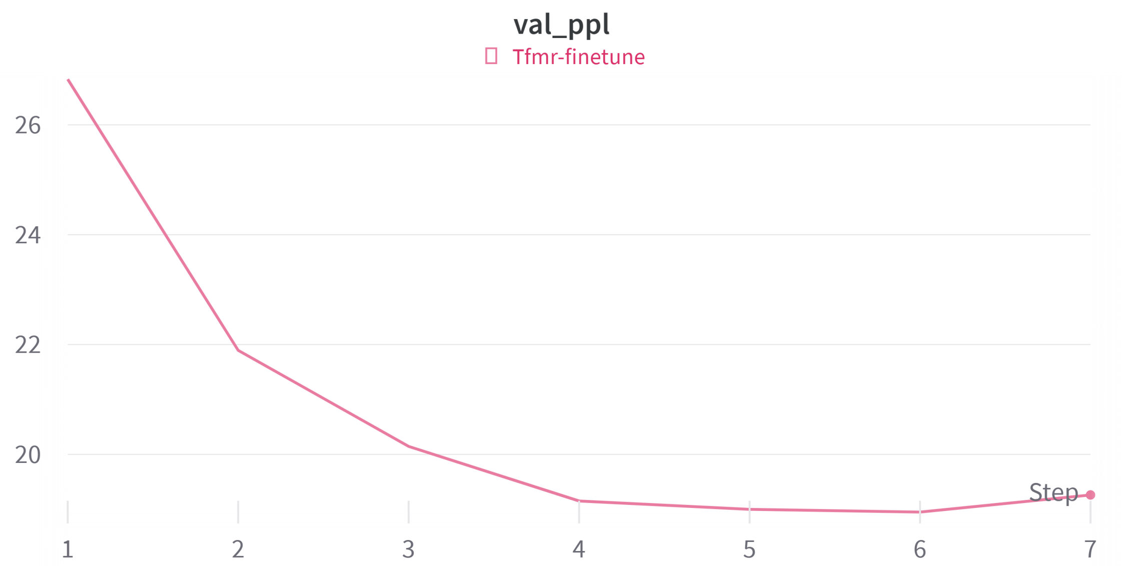
- train_loss:




- val_loss:




- val_ppl:



总的来说，scratch和finetune的模型表现如下：

| | Scratch | Finetune |
|---|---|---|
| train_loss | 2.072 | 2.095 |
| val_loss | 3.164 | 2.958 |
| val_ppl | 23.671 | 19.262 |

## 1.2 Test results

在这里，我们采用的生成策略为random

| | Scratch | Finetune |
|---|---|---|
| test_loss | 2.9299027919769287 | 2.743898868560791 |
| test_ppl | 18.72580909729004 | 15.547484397888184 |
| forward BLEU-4 | 0.584 | 0.574 |
| backward BLEU-4 | 0.430 | 0.435 |
| harmonic BLEU-4 | 0.495 | 0.495 |

## 1.3 Comparison

可以看出，Finetune模型在验证集与测试集上的表现都比Scratch的要更好，不论是ppl还是bleu metric指标。另外，Finetune模型看起来拥有更好的泛化能力：Finetune模型和Scratch模型在训练集上误差相差不大（甚至Finetune要更高），但Finetune在验证集/测试集上的表现要明显更好。

# 2. Generation with different decoding strategies

以下采用以下8种组合进行探究：

$Scratch, \tau = 1, random,$

$Scratch, \tau = 0.7, random,$

$Scratch, \tau = 1, top - p = 0.9,$

$Scratch, \tau = 0.7, top - p = 0.9,$

$Finetune, \tau = 1, random,$

$Finetune, \tau = 0.7, random,$

$Finetune, \tau = 1, top - p = 0.9,$

$Finetune, \tau = 0.7, top - p = 0.9$

| | forward BLEU-4 | backward BLEU-4 | harmonic BLEU-4 |
|---|---|---|---|
| $Scratch, \tau = 1, random$ | 0.584 | 0.430 | 0.495 |

| | forward BLEU-4 | backward BLEU-4 | harmonic BLEU-4 |
|---|---|---|---|
| $Scratch, \tau = 0.7, random$ | 0.706 | 0.418 | 0.525 |
| $Scratch, \tau = 1, top - p = 0.9$ | 0.821 | 0.384 | 0.523 |
| $Scratch, \tau = 0.7, top - p = 0.9$ | 0.887 | 0.299 | 0.447 |
| $Finetune, \tau = 1, random$ | 0.574 | 0.435 | 0.495 |
| $Finetune, \tau = 0.7, random$ | 0.801 | 0.394 | 0.528 |
| $Finetune, \tau = 1, top - p = 0.9$ | 0.686 | 0.418 | 0.520 |
| $Finetune, \tau = 0.7, top - p = 0.9$ | 0.870 | 0.320 | 0.467 |

# 3. Cases from different strategy

## 1. Scratch, $\tau = 1$, random

A red passenger bus is in an indoor city at night .
A jumbo jet **jet** airplane is parked on the tarmac .
The sheep are grazing in the park with the grass .
A group of people walking past **a shop that came** .
Smalliaikes gas entertainment cap laying near a fire hydrant .
A plane parked on the runway with a very large tall plants .
A street light sitting up against a rain soaked downtown intersection .
A train tracks sitting in a grassy English blue sky .
A family with a monitor sitting on a bench and park with a green field with steps on them .
A man that is sitting in front of a fire hydrant .

## 2. Scratch, $\tau = 1$, top-p=0.9

A red passenger bus is in the street next to a forest .
A giraffe **drinking off of a zebra** in a field .
The sheep are grazing in the park with the grass .
A group of people walking past a shop that came .
A large bus driving past a brick building at the side of a street .
A plane parked on the runway with a very large tall plants .
A street light sitting outside in the rain while a bus drives down the street .
A train tracks sitting in a grassy area beside a forest .
A zebra and two giraffe standing in a park with trees .
A man that is sitting in front of a fire hydrant .

## 3. Scratch, $\tau = 0.7$, random

A red passenger bus is in the street next to a building .
A giraffe standing next to a tree in a park .
A couple of benches sitting in a park next to a forest .
A group of people walking across a street with a bus .
A large bus driving down a street near a city .
A plane parked on the runway with a very large field .

A street light sitting up against a rain soaked street .
A train tracks sitting in a field with cars on it .
A giraffe walking across a grassy dirt field .
A man and woman sitting on a bench with a dog .

## 4. Scratch, $\tau = 0.7$, top-p=0.9

A red bus parked on the side of the road .
A giraffe standing next to a tree in a park .
A couple of giraffes walking across the grassy area .
A group of people walking across a street with a bus .
A large bus driving down a street next to a forest .
A plane sitting on top of a runway with lots of snow .
A street light with a red light hanging from it .
A woman sitting on a bench with her umbrella on her cell phone .
A giraffe walking across a grassy plain with tall trees in the background .
A man and woman sitting on a bench with a dog .

## 5. Finetune, $\tau = 1$, random

A red passenger bus drives down an asphalt road past a wet street .
A jumbo jet plane sits in a lot of land at an airport .
The hipster poses over a park while crowds stares out the window .
A group of people walking past **a shop that came bell** .
Small airplane with gas pumps running across a rural area .
A plane parked on the **runway of a very large washed beach** .
A street light sitting up against a tree while a man rides .
A train travels down the highway with stairs in the background .
A family of giraffes walking through dirt and park with trees .
A man that is sitting on her bench while reading .

## 6. Finetune, $\tau = 0.7$, random

A red fire hydrant in the middle of the road .
A giraffe standing next to a zebra in a field .
A couple of benches sitting in a park next to each other .
A group of people walking along a street in a city .
A large bus driving down a street near buildings in the day .
A plane parked on the runway of a very large cliff .
A street light sitting in the middle of a downtown intersection .
A train travels down a city street lined with blue buses .
A giraffe walking across a grass covered dirt field .
A man and woman sitting on a bench while reading .

## 7. Finetune, $\tau = 1$, top-p=0.9

A red fire hydrant in the middle of the road .
A jumbo jet plane sits in a lot of land at an airport .
The sheep are grazing in the field of the grass .
A group of people walking past a shop that **came to** .
A large bus driving down a street near buildings in London .
A plane parked on the runway of a very large **washed beach** .

A street light sitting below a tree filled with traffic .
A train travels down the highway with stairs in the background .
A family of giraffes walking through a field .
A man that is sitting on a bench while reading .

### 8. Finetune, $\tau = 0.7$, top-p=0.9

A red fire hydrant in the middle of the road .
A giraffe standing next to a tree on a sunny day .
A couple of giraffes walking across the grass on a hill .
A group of people walking along a street in a city .
A large bus driving down a street next to a red bus stop .
A plane parked on the runway of a very large cliff .
A street light sitting in the middle of a city intersection .
A street light sitting in a park with cars on the side of it .
A giraffe walking across a grass covered field near trees .
A man and woman sitting on a bench with her dog .

- 以上列出了一些语法错误（基本上没有）以及非常非常不能解释的逻辑错误。一个typical错误是模型很容易在shop后面生成that came...。我查阅了训练集的数据，发现并没有这样的句型，而且came只在训练集出现了一次：

  There must have been a bad storm that came through this city .

有些令人费解，可能的原因是，这些句子中的某些词向量的embedding存在不显然的相似，故在计算attention时导致了这样的结果。

- 生成效果最好的是Finetune, $\tau = 0.7$，top-p=0.9的模型。这里所说的效果好是指语法错误最少，语义最合理。再展示一些这一组的生成：

```
A couple of giraffes stand in a grassy area with trees in the background .
A giraffe standing in a field of grass in the wild .
A green double decker bus is driving down the street .
A woman is sitting on a bench with her cell phone .
A traffic light sitting in the middle of the street .
A picture of a giraffe looking into the distance .
A group of people are on a bench by the water .
A couple of giraffe standing next to a tree in the grass .
A man standing next to a woman near a bus .
A man standing next to a fire hydrant near a city street .
A red and white fire hydrant sitting on the side of a street .
```

但这也导致了一个缺点：生成的内容之间往往比较接近。比如句子开头基本上是A，经常有A man standing next to这样的句子。

- PPL不能很好地体现人类对阅读体验的判断，BLEU指标则一定程度反映了人类对句子阅读感觉的评价。例如：Scratch模型的$\tau = 0.7$，top-p=0.9模型的BLEU指标很优秀，但PPL相较finetune很远。事实上Scratch模型的$\tau = 0.7$，top-p=0.9模型生成的内容也是很不错的（语法错误较少，语义较清晰）。

## 4. Final Result

没有改动训练的超参数（即与报告开头提到的一样）。最终选用的生成模型是finetune，$\tau = 0.7$，top-p=0.9，各个metric如下：

| PPL | forward BLEU-4 | backward BLEU-4 | harmonic BLEU-4 |
|---|---|---|---|
| 15.547484397888184 | 0.870 | 0.320 | 0.467 |

## 5.1 Compare Transformer and RNN from at least two perspectives such as time/space complexity, performance, positional encoding, etc.

### 时间复杂度

记序列长度为$T$，隐含层维数为$d$，考虑Multi-head Attention的情况：假设我们有$h$个head，每个head的维数为$\frac{d}{h}$

- 将原输入转换为多个head，以及多个head转换为原维数的时间：（h个）$T \times d$与$d \times \frac{d}{h}$的矩阵相乘，复杂度为$O(Td^2)$。
- 计算attention：（h个）$T \times \frac{d}{h}$与$\frac{d}{h} \times T$的矩阵相乘，复杂度为$O(T^2d)$。

故总的时间复杂度为$O(T^2d + Td^2)$。但若认为第一项花费的时间可以省略，则复杂度为$O(T^2d)$。

RNN一次的复杂度为$O(d^2)$，共$n$次，故时间复杂度为$O(Td^2)$。

### 空间复杂度

- Transformer储存最后softmax前的输出需要$O(Td)$的空间，储存$QK^\top$需要$O(T^2)$的空间，总的需要$O(T^2 + Td)$的空间。
- RNN需要存隐藏状态$h_t$，共$O(Td)$的空间。

### 表现

- Transformer总体的表现比RNN要更好，可能是因为Transformer能够整体观察句子信息，句子中任意两个位置的信息都能通过Attention机制计算出来，而RNN中的关系只有相邻的单词是直接连接的。
- 而且，处理长序列时，RNN可能出现梯度爆炸问题。Transformer则因为注意力机制，不会出现这个问题。

### 位置编码

Transformer与RNN最大的不同是，由于attention机制本身不带关于时序信息，故在做embedding时，transformer需要手动加入位置编码信息。位置编码信息可以是固定的（例如正弦函数），也可以通过一个可学习的embedding层来实现。

## 5.2.1 During inference, we usually set `use_cache` in model_tfmr.py to `True`. What is the argument used for?

在推理过程中，由于output是一个一个蹦出来的，所以在decoder子层的第一个attention块中，V和K的计算也是一个单词一个单词计算的。在计算到第$t+1$个token时，前$t$个token计算的K和V可以利用起来，即将cache里的$K_{1:t-1}$和新的$k_t$拼接起来（Value同理）。

对应了代码中的：

```
if layer_past is not None:
    past_key, past_value = layer_past
    key = torch.cat((past_key, key), dim=-2)
    value = torch.cat((past_value, value), dim=-2)

if use_cache is True:
    present = (key, value)
else:
    present = None
```

Reference: [大模型推理加速：看图学KV Cache - 知乎 (zhihu.com)]

## 5.2.2 计算在推理时的复杂度

使用cache技术，在进行第$t$步推理时，我们只需要第$t$步的query。于是在推理过程中，各个阶段耗费的时间分别为：

- 多头Attention：（n个）$1 \times \frac{d}{n}$与$\frac{d}{n} \times t$的矩阵相乘，复杂度为$O(td)$，这一步生成了一个q
- 与encoder输出的K，V进行计算，这一步也是一个多头attention，复杂度为$O(td)$。
- feed-forward：相当于1个MLP，复杂度为$O(d^2)$。
- 最后映射到Vocabulary上，复杂度为$O(dV)$

故第t步的复杂度为$O(Btd + Bd^2 + dV)$。

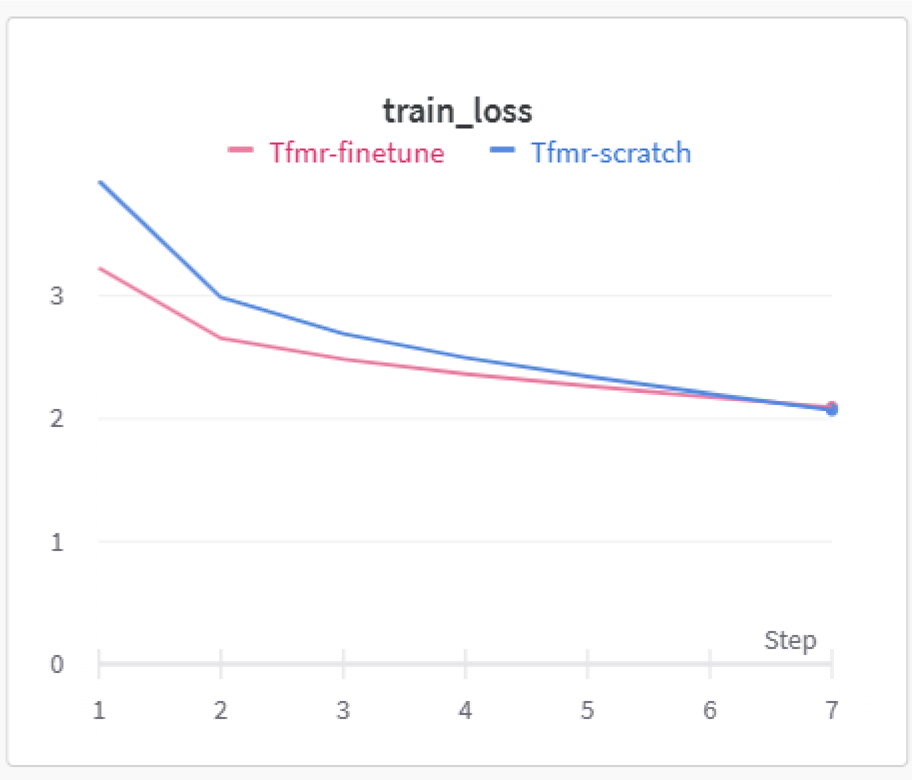
总的时间复杂度为$O(BT^2d + Bd^2T + TdV)$。

## 5.2.3 Based on your analysis of the question No 2., in which case the self-attention module dominate the time complexity? And in which case the feed-forward layer is dominant?

- self-attention的复杂度可以认为是$O(BT^2d)$。如果这一项要大于其他两项，必须有$T > d$，即句子序列足够长，长过隐含层维数。
- 相反，如果$d > T$，那么$d^2T$项占主导，也即feed-forward占据主导。

## 5.3 Discuss the influence of pre-training regarding the generation results, convergence speed, etc. Considering the experimental setup (the training task, data, pre-trained checkpoints, etc.), does the influence of pre-training meet your expectation?

- 收敛速度




将finetune模型和scratch模型的训练误差在一张图中显示，可以看出finetune模型可以使得收敛速度更快（需要更少的epoch收敛）。

- 泛化能力

正如前文提到，一个有意思的点是，虽然finetune和scratch在训练集上的误差相差不大，但finetune在验证集上的误差明显要更小。这说明finetune模型有更好的泛化能力，更好地避免了过拟合。

- PPL & BLEU

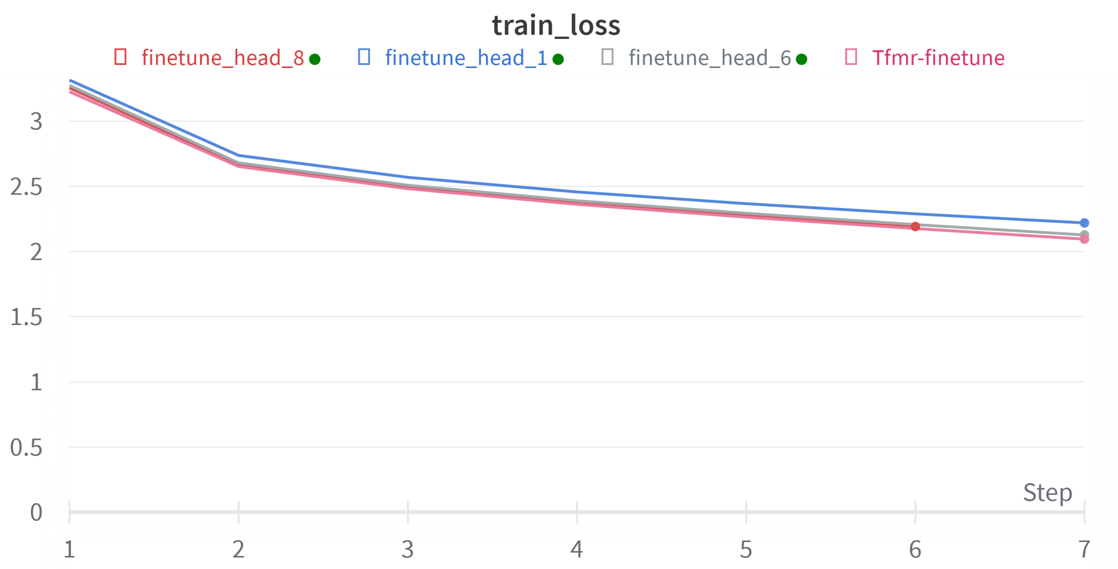
finetune模型能明显降低模型，但对于BLEU指标提升实际效果不大。

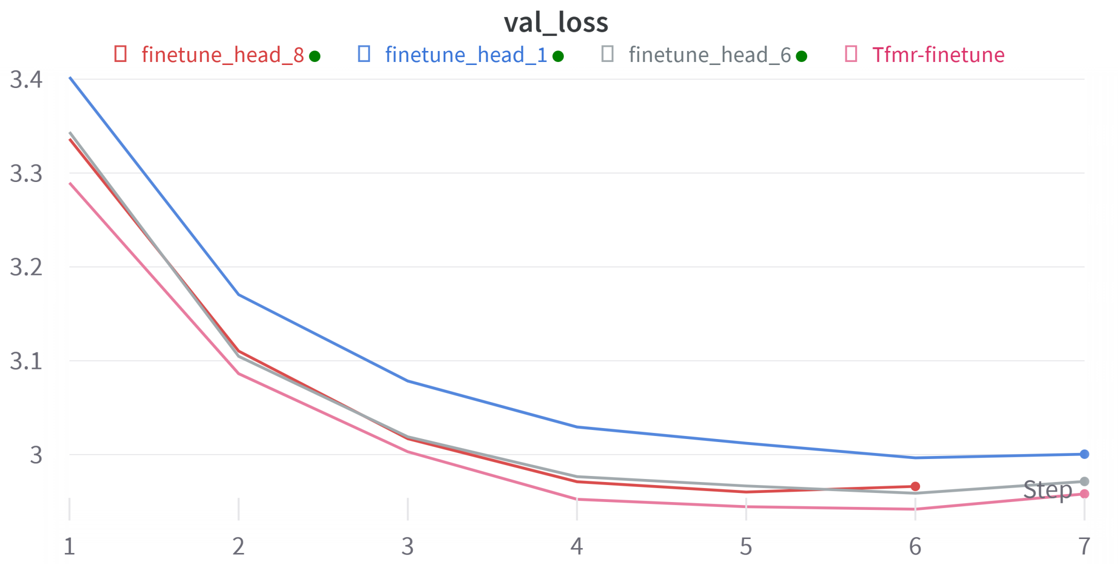
从task和data来说，finetune模型和scratch实际差距不大，但预训练的模型已经学到了一些数据中的特征，故在接下来的学习过程中，收敛速度更快，造成的过拟合现象更轻（更注重已经学到的一些数据特征，或者从另一个简单的角度来解释，训练集变得更大了）。

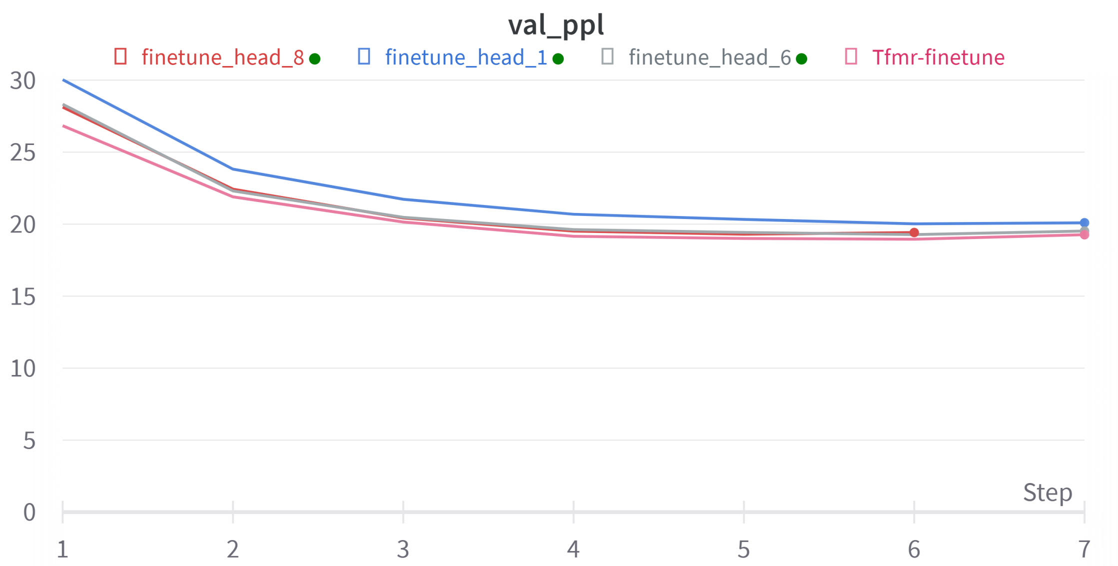
# Bouns

## Discuss the effect of the number of heads used in multi-head attention

选取3层的预训练模型，n_heads选取1, 6, 8, 12，分别进行训练和生成：




随后，分别对他们使用$\tau = 0.7$，top-p=0.9进行生成：

| head | PPL | forward BLEU-4 | backward BLEU-4 | harmonic BLEU-4 |
|---|---|---|---|---|
| 1 | 22.82 | 0.773 | 0.332 | 0.465 |
| 6 | 16.10 | 0.873 | 0.324 | 0.472 |
| 8 | 15.63 | 0.885 | 0.316 | 0.465 |
| 12 | 15.55 | 0.870 | 0.320 | 0.467 |

从结果来说，在训练过程中，除了单头的训练误差与验证集PPL都明显大于剩下三者以外，头数为6，8，12对训练过程并没有太大影响。而且，从并行计算的角度来说，更多头可能能更好加速计算过程。

首先，目前我们模型中的做法是：**将隐含层（768）分为头数乘以每一个头的隐含维度**。直接来说，这样的attention只能学到相邻（一定范围）内特征的相关性了。这样的好处可能是：过远特征的相似性没有在attention中体现，可能能解决一些过拟合的问题。坏处也由此诞生，一些相关性被丢弃了。

实验结果也与这个直觉有些吻合，更大的头数似乎导致了更好的泛化能力，不过差距实际上并非特别大。

虽然结果有点模棱两可，但原论文中对多头机制的解释是：不同的头能提取不同的特征关系。那么一个可能性便是，在不同的层间，不同的头可能发挥的作用不同。我觉得可能可以提出某种头的dropout机制，探索这种情况下模型的不同效果。