上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Tiny Transformer Lab Report

Mixed-Language Experiment Notes

Name Student ID
example@sjtu.edu.cn

July 31, 2026

Contents

1 Objective

This starter report records a toy experiment on a small transformer encoder. The goal is not to train a state-of-the-art model, but to show a clean place for objectives, procedures, figures, tables, equations, notes, and citations.

本实验的中文目标也很朴素：检查模板在中英混排时是否自然，顺便观察一个迷你模型在小数据集上的表现。If the numbers look too tidy, please assume the lab assistant had coffee before opening the spreadsheet.

2 Setup

The experiment uses a compact encoder with 4 attention heads and a hidden size of 128. Input samples are short bilingual sentences, for example, “the waveform is stable” and “输出曲线终于不像一根面条了”. The hardware monitor reported a board temperature between 26 °C ~ 31 °C.

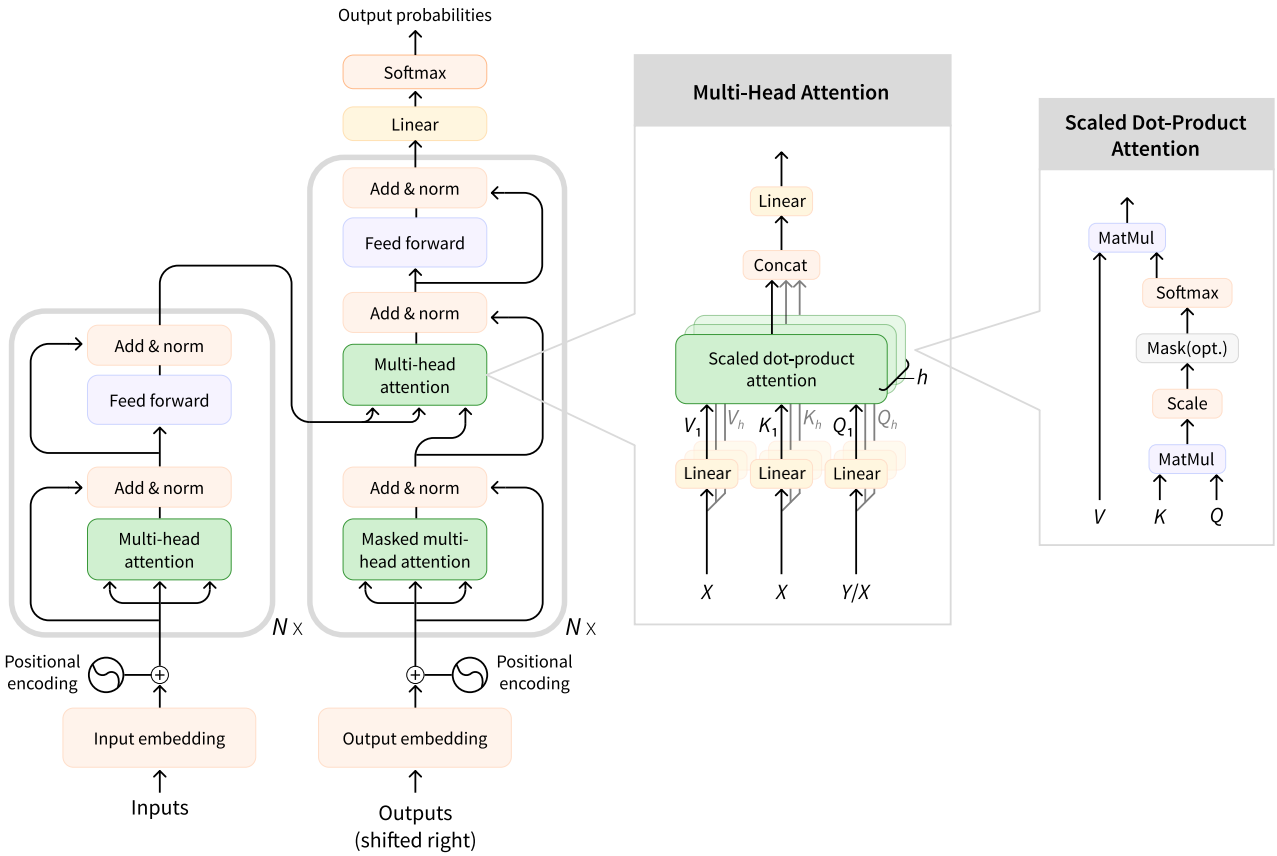

Figure 1: Transformer Architecture Overview. Image sourced from my blog.

3 Method

Figure 1 gives the overall architecture. The experiment procedure is as follows

- prepare a tiny bilingual dataset and split it into training and validation subsets;
- run a short sanity-check training pass for 12 epochs;
- record latency, validation accuracy, and any suspiciously cheerful log messages;
- write down what went wrong before future-you asks why the curve looks haunted.

For scaled dot-product attention, the unnormalized score is written as equation (1).

$$s_{ij} = \frac{Q_i K_j^{\mathsf{T}}}{\sqrt{d_k}}. \tag{1}$$

Then the attention weights are computed by applying the softmax function to the scores, as shown in equation (2).

$$a_{ij} = \frac{\exp(s_{ij})}{\sum_{k=1}^{n} \exp(s_{ik})}. \tag{2}$$

The following code snippet shows how to define a simple feed-forward layer in PyTorch.

Simple feed-forward layer in PyTorch.

```python
import torch
import torch.nn as nn

class SimpleFeedForward(nn.Module):
    def __init__(self, input_size, hidden_size, output_size):
        super(SimpleFeedForward, self).__init__()
        self.fc1 = nn.Linear(input_size, hidden_size)
        self.fc2 = nn.Linear(hidden_size, output_size)

    def forward(self, x):
        x = torch.relu(self.fc1(x))
        x = self.fc2(x)
        return x
```

A feed-forward layer consists of two `torch.nn.Linear` layers with an activation function (such as `torch.relu`) sandwiched between them.

4 Results

Table 1 summarizes one mock run. The English labels keep the table easy to scan, while the notes column shows that Chinese text fits naturally in the same layout.

> **Observation**
> The values below are illustrative. 这不是重大科学突破，只是说明示例表格、单位和中文备注可以一起工作。

Table 1: Mock training results

Epoch	Loss	Accuracy	Note
1	2.31	41.2 %	Warm start - 模型还在苏醒
6	1.08	73.5 %	Stable - 曲线开始像回事
12	0.64	86.9 %	Good enough - 别再调参了

5 Discussion

The result is consistent with the expected behavior of attention-based models on short sequence tasks [1]. In a real report, this section should discuss dataset bias, random seeds, measurement uncertainty, and whether the model merely memorized the examples.

误差来源包括样本太少、训练轮数太短、以及研究者在半夜把 “validation” 拼成 “vali-dation”。A more careful experiment would repeat the run with multiple seeds and report the mean latency, for example 7.8(4) ms.

6 Conclusion

This file is meant to be edited directly after cloning the template. Replace the fake transformer story with your own experiment, keep the structure if it helps, and delete any block that feels unnecessary.

总之，模板现在展示了标题页、目录、图、表、公式、引用、代码块、单位和中英混排。The next report can start here instead of starting from a blank page.

References

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al. *Attention Is All You Need*. 2017. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.