第 4 次作业报告

Siyuan Jin
20373680@buaa.edu.cn

Abstract

本文对比了 Seq2Seq 模型和 Transformer 模型在文本生成任务中的表现，并分析了它们的优缺点。

Introduction

1. Seq2Seq 模型

Seq2Seq (Sequence to Sequence) 模型是一种常见的深度学习架构，主要用于将一个序列转换为另一个序列。该模型最初是为了解决机器翻译任务而提出的，但后来被广泛应用于各种需要序列转换的任务，如文本摘要、对话系统、语音识别和图像字幕生成等。一个典型的 Seq2Seq 模型由两个主要部分组成：编码器（Encoder）和解码器（Decoder）。

(1) 编码器（Encoder）

编码器负责将输入序列转换为一个固定长度的上下文向量（context vector），也称为隐藏状态（hidden state）。编码器通常是由一系列的循环神经网络（RNN）、长短期记忆网络（LSTM）或门控循环单元（GRU）构成的。

1. 输入序列：假设输入为 $[x_1, x_2, ..., x_T]$。

2. 隐藏状态：编码器通过为每个时间步计算隐藏状态 h_t，最终将输入序列编码为一个上下文向量 h_T。

$$h_t = \text{RNN}(x_t, h_{t-1}) \tag{1}$$

(2) 解码器（Decoder）

解码器接收编码器生成的上下文向量，并生成目标序列。解码器的输入是上一个时间步的输出和编码器的上下文向量。解码器同样通常是由 RNN、LSTM 或 GRU 构成的。

1. 初始化隐藏状态：解码器的隐藏状态通常初始化为编码器的最终隐藏状态。

2. 生成输出：解码器在每个时间步生成一个输出 y_t，并将其作为下一个时间步的输入。

$$s_t = \text{RNN}(y_{t-1}, s_{t-1}) \tag{2}$$

$$y_t = \text{softmax}(W s_t) \tag{3}$$

2. Transformer

Transformer 模型是一种基于注意力机制的深度学习模型架构，由 Vaswani 等人在 2017 年提出，主要用于自然语言处理任务。与传统的 Seq2Seq 模型不同，Transformer 完全抛弃了循环神经网络（RNN），依赖自注意力机制（Self-Attention）来处理序列数据，极大地提升了并行处理能力和模型效率。Transformer 模型主要由编码器（Encoder）和解码器（Decoder）两部分组成，每部分都由多个层（Layer）堆叠而成。

(1) 编码器（Encoder）

编码器由多个相同的层（Layer）堆叠而成，每个层包括两个子层：

1. 多头自注意力机制（Multi-Head Self-Attention）
2. 前馈神经网络（Feed-Forward Neural Network）

多头自注意力机制（Multi-Head Self-Attention）自注意力机制允许每个位置的表示根据序列中所有其他位置的表示进行加权和。多头注意力机制将输入分成多个头，每个头独立地执行注意力计算，然后将结果拼接并线性变换。

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}}V) \tag{4}$$

其中，Q(Query))、K(key) 和 V(Value) 是通过输入序列线性变换得到的。

多头注意力机制可以表示为：

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_2)W^o \tag{5}$$

其中，$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$。

前馈神经网络（Feed-Forward Neural Network）前馈神经网络包括两个线性变换和一个激活函数，通常使用 ReLU：

$$\text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{6}$$

(2) 解码器（Decoder）

解码器的结构与编码器类似，但多了一个编码器-解码器注意力机制层（Encoder-Decoder Attention），用于将编码器的输出信息引入解码过程。

每个解码器层包括三个子层：

1. 多头自注意力机制（Multi-Head Self-Attention）
2. 编码器-解码器注意力机制（Encoder-Decoder Attention）
3. 前馈神经网络（Feed-Forward Neural Network）

(3) 位置编码（Positional Encoding）

由于 Transformer 模型没有循环或卷积结构，因此引入位置编码（Positional Encoding）来注入序列的位置信息。位置编码可以通过正弦和余弦函数计算得到：

$$PE_{pos,2i} = \sin(\frac{pos}{10000^{2i/d_{model}}}) \tag{7}$$

$$PE_{pos,2i+1} = \cos(\frac{pos}{10000^{2i/d_{model}}}) \tag{8}$$

其中，pos 是位置，i 是维度索引，d_{model} 是模型维度。

Transformer 模型通过自注意力机制和并行计算的特点，极大地提升了处理长序列的能力和计算效率。Transformer 不仅在机器翻译等自然语言处理任务中取得了显著的成功，还被广泛应用于图像处理、语音识别等领域。特别是其变体 BERT 和 GPT 在各种任务上的表现，使其成为现代深度学习的重要工具。

Methodology

M1：利用 Seq2Seq 实现文本生成任务

1. 导入语料文件；
2. 对导入的文件进行预处理；
4. 将处理后的文件分为训练集和测试集;
5. 使用 Seq2Seq 模型进行训练；
6. 利用训练好的模型完成文本生成任务。

M2：利用 Transformer 实现文本生成任务

1. 导入语料文件；
2. 对导入的文件进行预处理；
4. 将处理后的文件分为训练集和测试集;
5. 使用 Transformer 模型进行训练；
6. 利用训练好的模型完成文本生成任务。

Experimental Studies

M1：利用 Seq2Seq 实现文本生成任务

实验设定：本实验 Seq2Seq 模型中的 RNN 均采用 LSTM 模型，编码器和解码器的文字编码嵌入维度均设为 150；编码器和解码器隐藏层维度均设为 100。

训练样本与测试样本的生成：本实验采用金庸小说中《神雕侠侣》的小说语料构建样本。具体地，去除停用词后，以句号“。”作为分割符划分该小说中的句子，并在所有句子中挑选出满足以下条件的句子：

1. 该句子中包含“她”这个字；
2. 该句子的字数不小于 10、不高于 40；
3. 该句子后面一句话的字数不小于 10、不高于 40。

挑选出 300 句满足上述三个条件的句子，作为训练样本；而这 300 条句子中每一个句子紧接着的后面一句话，则作为训练标签。另外，再挑选出与训练样本不重复的 10 句满足上述三个条件的句子，作为测试样本。在训练集上训练好模型后，编码器输入测试样本，解码器输出的结果即为对应的文本生成结果。

批次数据对齐处理：对于编码器的每一句输入，均在输入的文本序列开头加上开始标识符“<BOS>”，在文本序列末尾加上结束标识符“<EOS>”；文本生成时，解码器在初始时刻的输入均为开始标识符“<BOS>”。此外，为了统一同一批次数据的 one-hot 编码维度，笔者将同一批次数据输入的每一条文本序列末尾添加补齐标识符“<PAD>”，直至该文本序列的长度与该批次数据中最长文本序列的长度一致。

模型训练设置：模型迭代训练 200 代，批次大小设置为 2，学习率设置为 0.001。训练过程的 loss 曲线如图1所示。可以看到，经过约 100 轮的训练后 Seq2Seq 模型的 Loss 逐渐收敛。

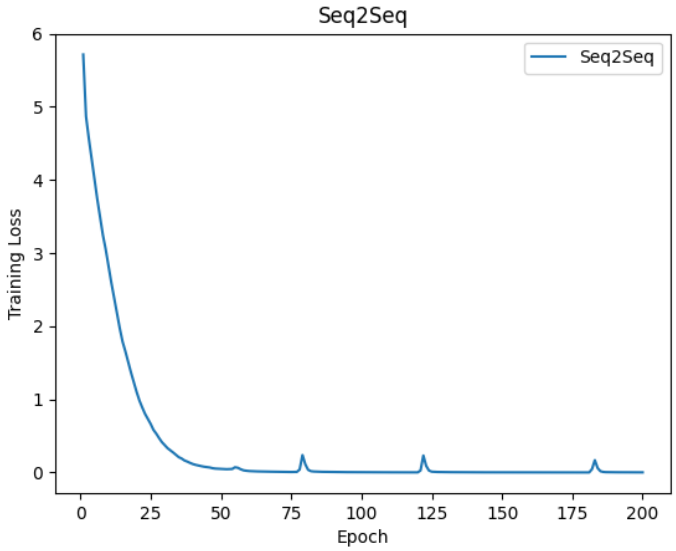

图 1: Seq2Seq 模型训练损失曲线

接着，将原文中的另外 10 句不在训练集的文本作为输入，记录训练好的 Seq2Seq 模型的输出，其结果如表2所示。

表 1: Seq2Seq 文本生成结果统计表

序号	输入（小说原文）	目标输出（小说下一句）	实际输出（预测下一句）
1	李莫愁师徒自北至南、自南回北兜截了几次，始终不见她的踪影	这一晚事有凑巧，师徒俩行至潼关附近，听得丐帮弟子传言，召只西路帮众聚会	只见她落入一座屋子的院子，推门进房
2	果然金轮法王不愿与她拚命，低头避过，只这麽一低头，手上轮子送出略缓	小龙女已乘机收回绸带，玎玎当当一阵响，圆球与轮子相碰，已将金轮的攻招解开	陆无双叹了口气，心想：这人原来真是傻的
3	起初十馀招那少女居然未落下风，她身在驴背，居高临下，弯刀挥处，五人不得不跳跃闪避	又斗十馀招，姬清虚见手中这柄断剑实在管不了用，心念一动，叫道：皮师弟，跟我来	杨过道：那倒不想起她个不瞅
4	但说也奇怪，她话声却极是柔娇清脆，令人听之醒倦忘忧	杨过道：既然如此，如何救人一凭姑娘计议	程英见到那女之已之後，我们的年纪都是活的狗身上了
5	二丐见她回到桌边坐下喝酒，背向他们，於是一步步的挨向梯边，欲待俟机逃走	李莫愁转身笑道：瞧来只有两位的腿骨也都折断了，这才能屈留大驾	她在石上所写的字，就是这一首诗的前半截八句
6	杨过将她紧紧抱住，在她嘴上亲去	小龙女在他一吻之下，心魂俱醉，双手伸出去搂住他头颈	这一首诗的前半截八句

续下页

表 1 – 续前页

序号	输入（小说原文）	目标输出（小说下一句）	实际输出（预测下一句）
7	杨过抢步上前，将她搂在怀里，柔声道：龙儿，你不好，我也不好，咱们何必理会以后	今天你不会死的，我也不会死	愁怀中了一行人般张臂将他抛得影踪不见
8	金轮法王当她从左侧掠过时回肘反打，竟然一击不中，心下也佩服她身法轻捷	杨过又拾起武修文掉下的长剑交在她手里，说道：姑姑，这和尚无礼，咱们打他	郭靖此时齐喝道：喂，这位姑娘，你把我的兵刃踏在地下干么？侧身长臂，来抓玉箫
9	她却不知蜂翅上的细字被周伯通发见，而给黄蓉隐约猜到了其中含义	两人说了半天话，小龙女回进屋去烧了一大盆鱼，佐以水果蜂蜜	黄蓉一声斜见了，你哥儿俩同时尽力巴结
10	李莫愁见了她这副可怜巴巴的模样，胡乱打骂一番，出了心中之气，也就不为已甚	陆无双如此委曲求全，也亏她一个小小女孩，居然在这大魔头门下挨了下来	英雄如此，晚辈告辞

M2：利用 Transformer 实现文本生成任务

实验设定：本实验 Transformer 模型的编码器和解码器的嵌入层维度均设置为 256，而各自的隐藏层维度均设置为 512；8 头 6 层。

训练样本与测试样本的生成：本实验采用金庸小说中《神雕侠侣》的小说语料构建样本。具体地，去除停用词后，以句号“。”作为分割符划分该小说中的句子，并在所有句子中挑选出满足以下条件的句子：

1. 该句子中包含“她”这个字；
2. 该句子的字数不小于 10、不高于 40；
3. 该句子后面一句话的字数不小于 10、不高于 40。

挑选出 300 句满足上述三个条件的句子，作为训练样本；而这 300 条句子中每一个句子紧接着的后面一句话，则作为训练标签。另外，再挑选出与训练样本不重复的 10 句满足上述三个条件的句子，作为测试样本。在训练集上训练好模型后，编码器输入测试样本，解码器输出的结果即为对应的文本生成结果。

批次数据对齐处理：对于编码器的每一句输入，均在输入的文本序列开头加上开始标识符“<BOS>”，在文本序列末尾加上结束标识符“<EOS>”；文本生成时，解码器在初始时刻的输入均为开始标识符“<BOS>”。此外，为了统一同一批次数据的 one-hot 编码维度，笔者将同一批次数据输入的每一条文本序列末尾添加补齐标识符“<PAD>”，直至该文本序列的长度与该批次数据中最长文本序列的长度一致。

模型训练设置：模型迭代训练 100 代，批次大小设置为 2，学习率设置为 0.001。文本生成与多样性调节：在文本生成过程中，使用了温度调节（temperature）技术来控制生成文本的多样性。温度值较低（如 0.6）会使模型倾向于选择较高概率的词汇，从而生成更确定性的输出。温度值较高则增加输出的随机性和多样性。

训练过程的 loss 曲线如图2所示。可以看到，经过约 100 轮的训练后 Seq2Seq 模型的 Loss 没

图 2: Transformer100 轮次模型训练损失曲线

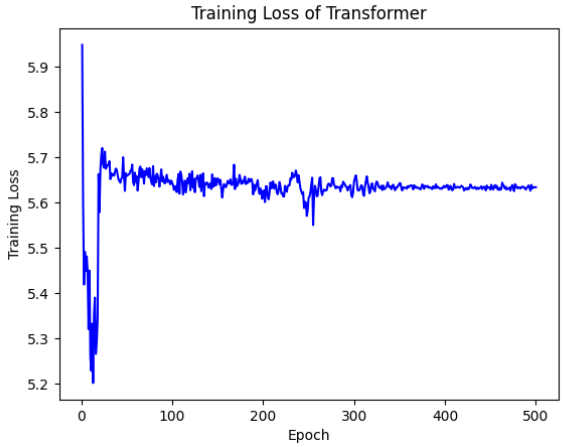

图 3: Transformer500 轮次模型训练损失曲线

有明显的收敛趋势，增大训练轮数后，500 轮次的损失曲线如图3所示，损失仍未有明显下降趋势。

接着，将原文中的另外 10 句不在训练集的文本作为输入，记录训练好的 Transformer 模型的输出，其结果如表2所示。

表 2: Transformer 模型文本生成结果统计表

序号	输入（小说原文）	目标输出（小说下一句）	实际输出（预测下一句）
1	她这些年来武功大进，内力强劲，出掌更是变化奥妙，十馀招中，欧阳锋竟丝毫占不到便宜	郭靖叫道：欧阳先生，别来无恙啊	
2	她曾与李莫愁交过手，平时听武氏兄弟说起杀母之仇，心中早当她是世上最恶毒之人	黄蓉道：李道长帮咱们去找你妹子	，

续下页

表 2 – 续前页

序号	输入（小说原文）	目标输出（小说下一句）	实际输出（预测下一句）
3	杨过叹道: 我想念她, 倒也不是为了她美貌，就算她是天下第一丑人，我也一般想念	不过……不过要是你见了她，定会更加称赞	
4	你们斩我一千刀、一万刀, 我还是要她做妻子	这番话当真是语惊四座，骇人听闻	了白不, , , : 是在道小，了
5	她父母生前将女儿托付於他抚养	他受人重托, 责任未尽, 此时大难临头，便将这块救命的锦帕给了她	过, 道杨道下, 咱, , 若行道, 一
6	她随身带同一双白雕，若有紧急事，便可令双雕传递信息	程英、陆无双姊妹坚要陪她同去	女上不後此
7	日后见了我妻子，我也会告诉她	说到这里，语音已然哽咽	也, , 了, 道了, 不半: 郭人, 便
8	郭襄道: 妈, 她也是没有法子啊	她既欢喜了杨叔叔，杨叔叔便有千般不是，她也要欢喜到底	，头过不後，
9	小龙女轻轻的道: 我师姊呢? 她也来了麽? 洪凌波道: 我师父命弟子先来，请问师叔安好	小龙女道: 你出去罢, 这个地方莫说是你，连你师父也是不许来的	，道，女, , 不了
10	陆姑娘咬定那部秘本给丐帮拿了去，赤练魔头便押著她去追讨	谅来她性命一时无妨，折磨自然是免不了	, , , 杨道带, 不她, 无你她到似,

根据文本生成结果可见，与 Seq2Seq 模型相比，Transformer 模型几乎很难生成完整的句子，其模型收敛性也不如 Seq2Seq，这是因为训练的数据量太小，Transformer 模型达到较好的性能表现通常需要经过较大的数据。

Conclusions

基于本实验的研究内容，总结 Seq2Seq 模型与 Transformer 模型在文本生成任务中的优缺点如下：

Seq2Seq 模型：优点：

1. 模型结构简单：Seq2Seq 模型由编码器和解码器组成，结构相对简单，易于理解和实现。

2. 训练收敛较快：相比于 Transformer 模型，Seq2Seq 模型通常收敛速度更快，需要的训练数据量也相对较少。

3. 生成句子完整性较好：在本次实验中，Seq2Seq 模型生成的句子相对完整，能够生成连贯的文本片段。

缺点：

1. 并行处理能力有限：Seq2Seq 模型依赖于循环神经网络，并行处理能力有限，难以处理长序列数据。

2. 长距离依赖问题：Seq2Seq 模型难以捕捉长距离依赖关系，导致生成文本的连贯性和流畅性不如 Transformer 模型。

Transformer 模型：

优点：

1. 并行处理能力强：Transformer 模型基于注意力机制，能够并行处理序列数据，计算效率高，能够处理更长的序列。2. 长距离依赖能力强：Transformer 模型能够有效捕捉长距离依赖关系，生成文本的连贯性和流畅性较好。3. 生成文本多样性高：通过温度调节等技术，可以控制 Transformer 模型生成文本的多样性，使其更加丰富和灵活。

缺点：

1. 模型结构复杂：Transformer 模型结构复杂，参数量较大，训练难度较高。2. 需要大量数据：Transformer 模型通常需要大量数据才能达到较好的性能，训练成本较高。3. 生成句子完整性较差：在本次实验中，Transformer 模型生成的句子相对不完整，难以生成连贯的文本片段。