单位代码_____
学　　号 ZY2303803
分 类 号_____

北京航空航天大學
BEIHANG UNIVERSITY

深度学习与自然语言处理（NLP）第四次课后作业

院（系）名称　自动化科学与电气工程学院

专 业 名 称　自动化

学 生 姓 名　董晨辉

2024 年 06 月

深度学习与自然语言处理（NLP）第四次课后作业

董晨辉
1127666815@qq.com

Abstract

本文利用金庸小说语料库，用 Seg2Seq 与 Transformer 两种不同的模型来实现文本生成的任务(给定开头后生成武侠小说的片段或者章节)，针对这两种模型本报告分别使用 Seq2Seq 模型以及基于 Transformer 的(GPT-2 模型实现文本生成，并对比与讨论两种方法的优缺点。

Introduction

Seq2Seq 模型通常由两个主要部分组成:编码器(Encoder)和解码器(Decoder)。编码器处理输入序列，将其编码为固定长度的上下文向量(conlcxtvcclor)。1 解码器根据上下文向量生成输出序列。它的工作流程是输入序列通过编码器，每个时间步(时间步 t)都会更新隐藏状态;编码器的最后一个隐藏状态作为上下文向量，传递给解码器。解码器根据上下文向量和之前生成的词(在训练时为真实词，在推理时为上一步生成的词)，逐步生成输出序列。

Transformer 模型由编码器堆栈(Encoder Stack)和解码器堆栈(Decoder Stack)组成每个堆栈包含多个相同的层(Layers)。每个编码器层包含自注意力机制(Self-Attention)和前馈神经网络(Feed-Forward Neura Network)。每个解码器层包含自注意力机制。

码器-解码器注意力机制(Encoder-Decoder Attention)和前馈神经网络。Transformer 模型执行流程是输入序列通过嵌入层(EmbeddingLayer)并添加位置编码(Positional Encoding)以保留顺序信息。然后编码器堆栈对输入序列进行编码，生成一系列隐藏状态。最后解码器堆栈根据编码器的输出和前一步的输出序列(在训练时为真实词在推理时为上一步生成的词)，逐步生成输出序列。

Methodology

本次实验利用 Seq2Seq 和 Transformer 模型进得到文本生成模型，在 Seq2Seq 模型中的 Encoder 和 Decoder 模块都利用 LSTM 模型进行训练。训练时利用金庸先生的 16 本小说作为实验数据集，进行文本生成模型的训练和测试实验。

1 数据预处理

由于数据库里存在各种标点符号以及网页信息，所以首先需要对数据进行预处理操作。

删除 txt 文件中关于网址描述的与金庸武侠小说内容无关的字符"本书来自 www.cr173.com 免费 txt 小说下载站\n 更多更新免费电子书请关注 www.cr173.com","本书来自 www.cr173.com 免费 txt 小说下载站。

删除非中文字符，根据中文字符的 utf-8 编码的字节长度为 3 来判断;

删除标点符号，并且根据带有分割意义的标点符号['\n','。','？','！','，','；','：',' . ']对文本进行按句换行分割。

2 分词

本文选择"结巴（jieba）"中文分词模块，该模块可以支持三种分词模式：精确模式，试图将句子最精确地切开，适合文本分析；全模式，把句子中所有的可以成词的词语都扫描出来,速度非常快，但是不能解决歧义；搜索引擎模式，在精确模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词。同时，由于金庸小说中包括部分繁体字，该模块可以支持繁体分词、支持自定义词典。

本文使用 jieba.cut()进行分词，例如对以下一句话：

"武林至尊宝刀屠龙号令天下莫敢不从倚天不出谁与争锋"

进行断句后得到：

['武林','至尊','宝刀','屠龙','号令','天下','莫敢','不','从','倚天','不出','谁','与','争锋']

可以看出 jieba 可以对中文句子很好地进行分词操作。之后进行数据集制作。

3 训练模型

（1）字典生成

将文本语料库 corpus_chars 的字符不重复统计，可以得到一个字典，并且给字典的每个字符对应一个索引，本来语料库是由中文字符组成的，可以通过字典来将字符转换成索引，得到索引组成的语料库

（2）WordEmbedding

建立字典可以将字符变成索引，还需将索引变成词向量，这一步叫做词嵌入，即 WordEmbedding，词向量可以是不用训练的，比如 one-hot，也可以是需要训练的，比如使用 torch.nn.Embedding()。本次实验使用 one-hot 向量。

（3）数据集生成

num_steps,batch_size 两个参数分别代表训练集的文本序列长度和批样本数量。输入进网络的文本可以表示成[batch_size,num_steps]的一个索引 tensor。这一步通过对 corpus_indices 切片分块来实现，前 num_steps 个 token 作为输入，后 num_steps 个 token 作为输出。

（4）seq2seq 模型

在本次的 seq2seq 模型中，编码器和解码器都是采用 LSTM 网络，直接使用 pytorch 的 torch.nn.LSTM(input_size,hidden_size,num_layers)模块。

input_size 代表输入 sequence 的特征维度；

hidden_size 代表 hiddenstate 的特征维度；

num_layers 代表 LSTM 网络层数。

由于输入的是 one-hot 向量，维度为字典长度 len(char_to_idx)=1186，hidden_size 可以设置为 128，256，512，1024。num_layers 可以设置为 1，2 等。

loss=nn.CrossEntropyLoss()

optimizer=torch.optim.Adam(model.parameters(),lr=lr)

反向传播过程中使用了梯度裁剪 grad_clipping()

Experimental Studies

Experimental 1

使用 seq2seq 传统的方法，利用 LSTM 作为编码器和解码器进行训练，实验结果如下：

网络参数如下：ENC_EMB_DIM = 200，DEC_EMB_DIM = 200，HID_DIM = 512，N_LAYERS = 3，ENC_DROPOUT = 0.5，DEC_DROPOUT = 0.5，N_EPOCHS = 10。

输入文本为：start_text = "马背上伏的是个高瘦的汉子，汉子手里拿了一把长剑，剑长三

尺"。

输出结果为:“马背上伏的是个高瘦的汉子，汉子手里拿了一把长剑，剑长三尺上兜的是悍高小小汉子，汉手手里拿铲一把长剑之剑长「对上歌的是般高小小汉子，你手手里拿布一把长剑之剑长「对上歌的是般高小小汉，，你手手里拿布一把长剑之剑长「对上歌的是般高小小汉，，你手手里拿布一把长剑之“。

生成的效果不好。

Experimental 2

使用 Transformer 的方法，进行训练，实验结果如下：设置 gpt2 的 epoch = 10，batch_size = 4。

输入文本为：start_text = "马背上伏的是个高瘦的汉子，汉子手里拿了一把长剑，剑长三尺"。

输出结果为:“马背上伏的是个高瘦的汉子，汉子手里拿了一把长剑，剑长三尺，上身是一个巨大的粗壮汉子。他在剑上伸了个长枪，长剑是一个很大的长剑。他不但不砍了长剑，还杀了一个巨汉，最后还杀了一个巨汉，他用身躯扛下一把“

生成的结果相较于 seq2seq 较好。

Conclusions

本文基于 Seq2Seq 和 Transformer 模型来实现文本生成的模型，输入可以为一段已知的金庸 小说段落， 来生成新的段落并做分析。但是经过实验分析发现，无论是哪种方法，生成的文字效果并不太理想，实际并不可读，但是相比 seq2seq，transformer 的效果相对较好。

这是因为 Transformer 模型在机器翻译、文本生成、文本分类、命名实体识别等多个 NLP 任务中表现出了出色的性能。由于其自注意力机制，Transformer 模型可以更好地捕捉长距离依赖关系，从而在处理长序列时具有优势。在需要处理长序列或捕捉复杂依赖关系的任务中，Transformer 模型可能更有优势；而在某些特定的序列到序列转换任务中，Seq2Seq 模型可能更合适。

References

https://blog.csdn.net/shuihupo/article/details/85162237

https://blog.csdn.net/weixin_44966965/article/details/124732760