Report of Deep Learning for Natural Langauge Processing\_4

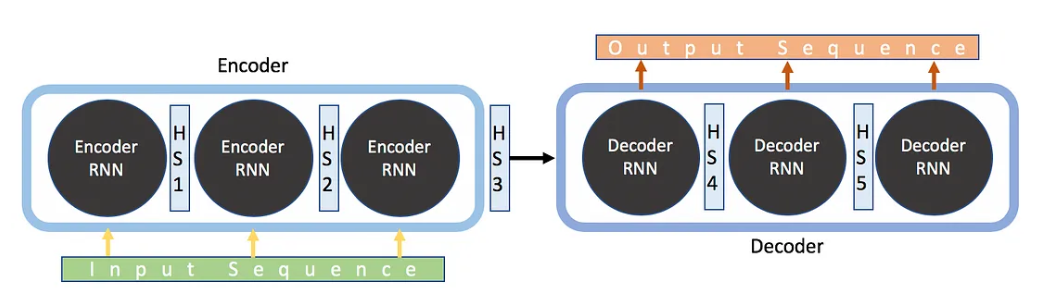
ZY2303109 金子棋

**Abstract**

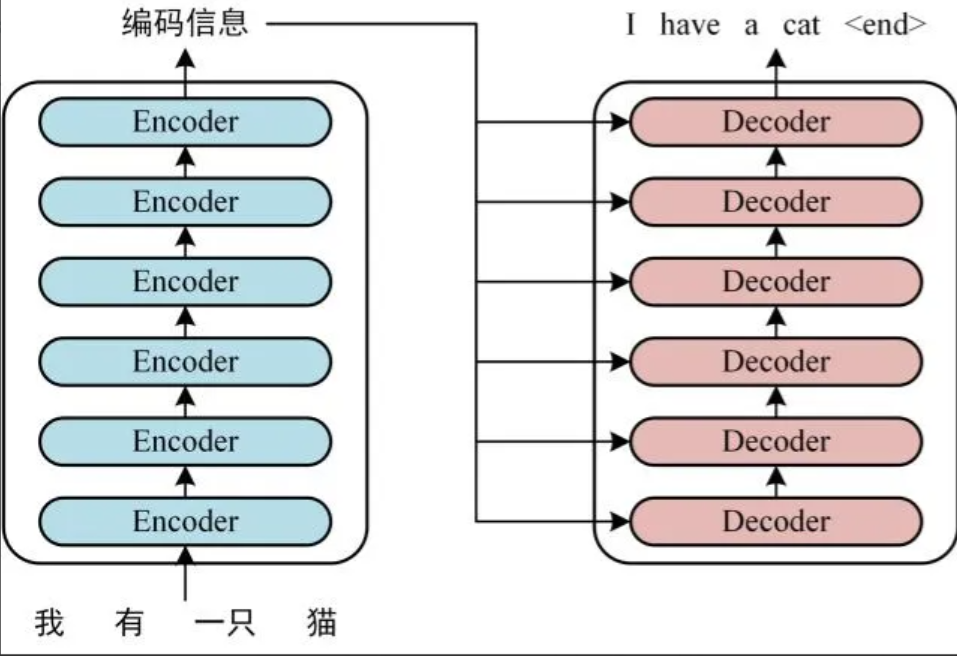
基于seq2seq及Transformer两种模型，对以金庸小说(共16篇)为内容的中文语料库进行训练，实现给定开头后生成后续文本的文本生成任务。

**Introduction**

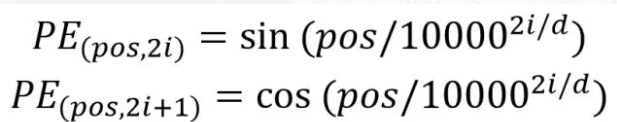
seq2seq即序列到序列的模型，其目的是根据给定的序列通过特定的生成方法生成另一个序列的方法，是RNN的一个变种，解决了RNN要求序列等长的问题，常用于机器翻译、人机对话、文本生成等任务。Seq2seq模型由一个编码器与一个解码器组成，两者都是由RNN单元组成。编码过程中，输入序列通过编码器，将输入的文本序列为压缩为固定长度语义向量C，语义向量C作为解码器的初始状态，进行第一次解码，之后将上一时刻的输出当成当前时刻的输入，由此依次生成指定长度的输出序列，完成文本生成。



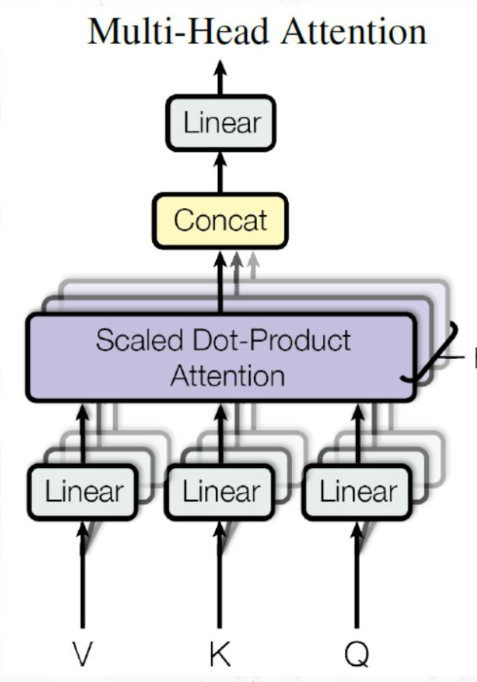
Transformer模型是一种基于自注意力机制的深度学习模型，被广泛运用于自然语言处理任务，较于传统的循环神经网络和长短时记忆网络，Transformer模型在处理长距离依赖关系时表现更好，同时具有更好的并行性能。类似于Seq2seq模型，Transformer同样是由编码器模块与解码器模块组成的，每个编码器与解码器模块都包含六个相同的子模块组成。



Transformer第一步首先要对输入进行处理，首先通过Word2Vec或Glove等方法对文本进行词向量预训练，得到词向量矩阵，并根据以下公式进行位置编码：



编码器层由多头注意力模块与前馈神经网络及两者之间的正则层组成。多头注意力模块由多个自注意力模块组成，也是Transformer模型发挥作用的关键机制，自注意力模块将每个输出向量生成三个向量，分别是查询向量、键向量和一个值向量，再根据生成向量进行分数值计算、归一化与Softmax输出，多头注意力层则是将自注意力机制转换为矩阵计算，并行计算加强了网络效果。再经过前馈神经网络的全连接层输出得到编码信息矩阵。



解码器层则由两个多头注意力模块与前馈神经网络及之间的正则层组成。第一个多头注意力模块采用了Mask机制，分为Padding Mask与sequence Mask，分别进行序列对齐与未来信息掩码，而第二个多头注意力模块则用编码器的输出编码信息矩阵进行运算， 经由前馈神经网络与Softmax层得到预测输出。

**Methodology**

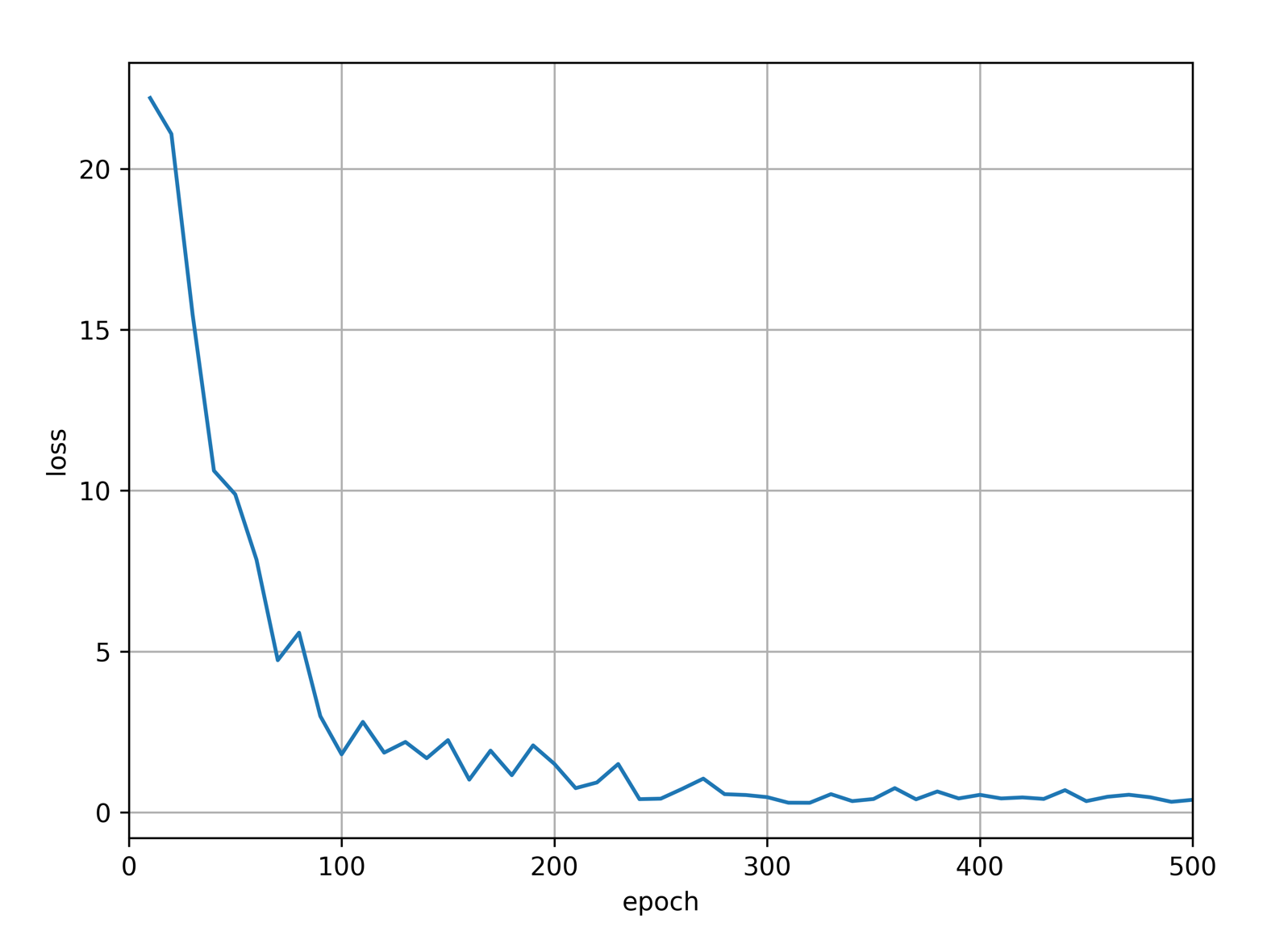
1. 文本预处理

本次使用的中文语料库为金庸小说集，有效正文文本文件共16个，此外还有包含所有文件的zip包、文本文件来源网站url链接以及包含所有章节标题的文本文件inf.txt，直接在文件夹中删去。

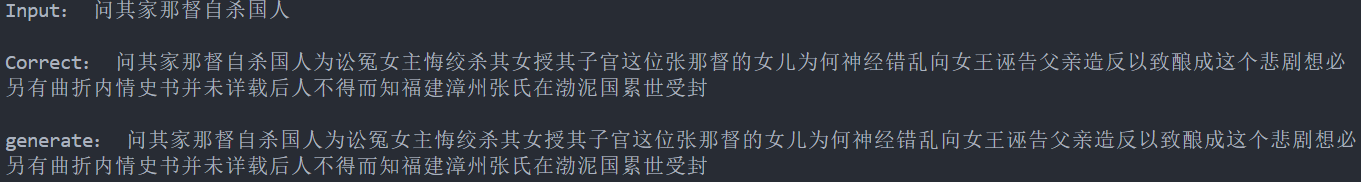
对于正文文本，一需要删除开头结尾网站附上的广告信息，二需要去除文本中非中文字符、空格换行等无效信息，三需要去除中文文本中的标点符号，最后仅剩中文文字作为后续处理的信息源。

1. Seq2Seq模型训练与效果验证

本次报告中seq2seq模型基于Tensorflow实现，RNN单元由tf.keras.layers.Embedding、tf.keras.layers.LSTM、tf.keras.layers.Dense三层接续构成，将输入以Time\_step等间距分开输入网络进行训练，用model.fit()训练后保存模型，然后再重新加载模型进行样例测试。训练过程中采用隐藏层设置为2\*64，采用Adam优化器，批次大小为32，学习率为0.005，训练500轮，结果如下：



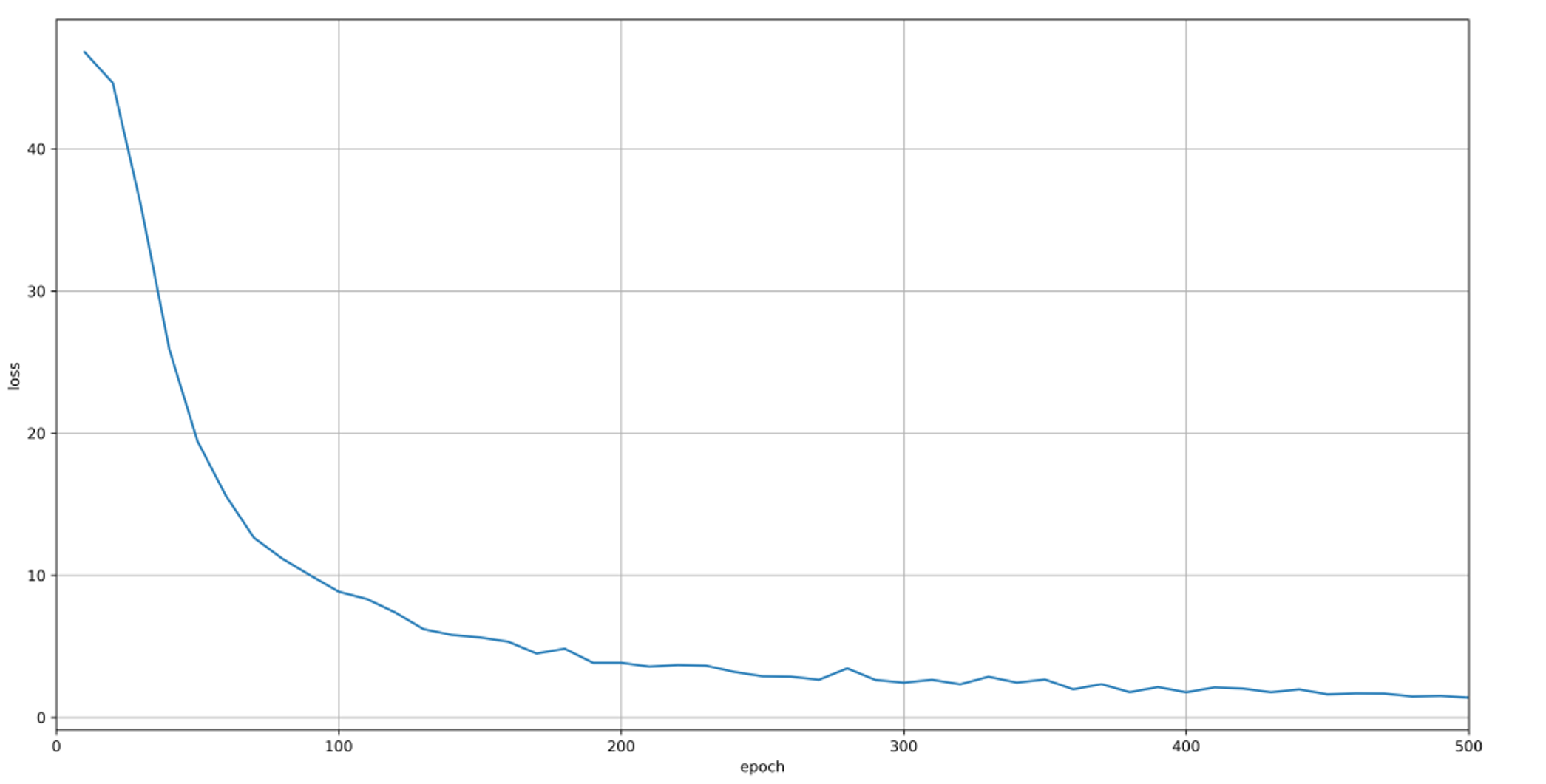
最终Loss为0.239，训练效果较好，测试文本生成任务如下：



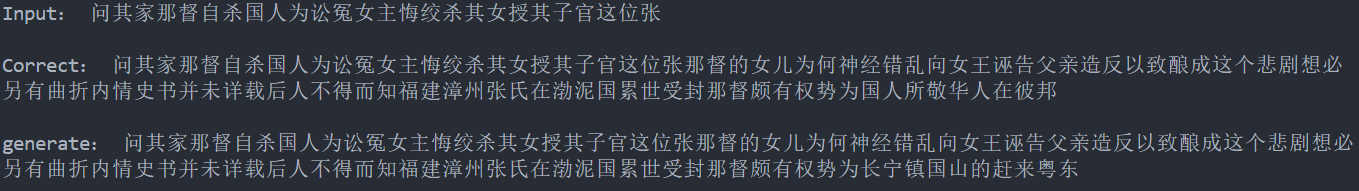
生成文本成功。

1. Transformer模型训练与效果验证

本次报告中Transformer模型依赖Tensorflow实现，结构比较复杂在这里不再赘述。设置隐藏层2 \* 64，dropout率为0.1防止过拟合，使用Adam优化器，设置学习率为0.002，训练500轮，结果如下：



最终Loss为1.236，相较于Seq2seq模型Loss较高，可能还需要增加训练轮数。测试文本生成任务如下：



可见大致上生成文本正确，对于更长的文本生成准确率不高。

**Conclusion**

Seq2seq模型作为常用的文本生成模型，结构较为简单，训练容易，具有比较好的生成效果，在简单的文本生成任务上表现良好，但是测试将文本拉到预测后100个词元效果就变差了，对于复杂的文本生成任务效果较差。同时RNN是顺序计算的，在计算上难以利用GPU资源，在数据量较大时不好加速。Transformer模型则结构较为复杂，训练调参比较麻烦，但是具有并行计算与复杂文本生成能力，整体上上限比Seq2Seq模型高。