|  |
| --- |
| 电子科技大学  **计算机专业类课程** |
| **实验报告** |
| **课程名称：人工智能实验II**  **学院专业：计算机科学与工程学院**  **学生姓名：崔若晨**  **学　　号：2021060902008**  **指导教师：石小爽** |
| **日　　期： 2023 年 9 月 17 日** |
|  |
| 电子科技大学计算机学院实验中心 |

**电 子 科 技 大 学**

**实 验 报 告**

**实验三**

# 一、实验室名称：

电子科技大学清水河校区主楼A2-413

# 二、实验项目名称：

人工智能实验II-3：基于RNN的文本翻译

# 三、实验原理：

RNN作为一种递归式处理序列问题的模型，在机器翻译、文本自动摘要和语音识别中有着成功的应用。巧妙的网络结构设计使得RNN可以捕捉语言中的长距离依赖关系，例如性别一致性和语法结构，而不必事先知道它们，也不需要跨语言进行 1:1 映射。

Seq2seq 是一类特殊的 RNN，它遵循了Encoder-Decoder的设计结构，两个部分均由RNN构成；Encoder将源语句转换为表示语义的向量，然后这个向量通过Decoder可以产生对应的翻译结果。

# 四、实验目的：

通过 PyTorch 框架，搭建神经网络，首先实现基本的RNN模型，以及RNN的变体—GRU。使用GRU搭建seq2seq模型，包括Encoder和Decoder两个部分以进行文本翻译任务。在此过程中需掌握：

1. PyTorch 中文本预处理(词表构建，词嵌入)的实现；
2. RNN和GRU的基本框架、Pytorch代码实现；
3. Seq2seq的基本结构和代码实现。

扩展掌握：

1. 基于RNN的图像描述生成(image caption)基本原理；
2. 具体的实现方式。

# 五、实验内容：

基于Pytorch实现seq2seq文本翻译实验，主要实验内容具体包括四个部分：

1. 文本翻译模型的实现
2. 分析模型在对应数据集上的准确率
3. 分析使用RNN和GRU作为seq2seq模型的架构时分别对最终结果(准确率，损失函数收敛程度)有什么影响，并比较RNN和GRU的优缺点
4. 可视化、实验结果分析

# 六、实验器材（设备、元器件）：

硬件平台：PC端

开发环境：操作系统Windows 11

测试环境：Anaconda 3，Python 3.11，PyTorch 2.0.1

# 七、实验步骤：

1. 问题描述

用RNN网络实现英法文本互译

2. 算法的概要设计与分析

将数据集中的法语和英语单词进行Embedding编码，使用Seq2Seq模型，采用2个RNN网络，一个RNN作为编码器，一个RNN作解码。采用Adan优化器进行优化。

3. 核心算法的详细设计与实现

**Embedding算法**：Embedding是一个存储向量的查找表，给定一个编号，能返回该编号对应的嵌入向量，嵌入向量反映了各个编号代表的单词之间的语义关系。Embedding能够用低维向量对物体进行编码还能保留其含义。在本实验中，输入是法语或者英语的编码，然后为每一个编码生成一个256维的向量，输出为所有向量构成的Embedding矩阵。

**RNN模型：**RNN网络用于解决训练样本输入是连续的序列,且序列的长短不一的问题。RNN最大的特点在于隐层。如图 1所示，每一层的隐层将上一个隐层的输出乘上权重V，然后与当前时刻输入乘上U进行相拼接，得到一个基础值，通过函数（通常为tanh函数）激活后，更新得到新的。每一层的输出由乘上权重W，然后通过函数（通常为Sigmoid函数）激活后得到。权重U,V,W通过反向传播算法进行更新。

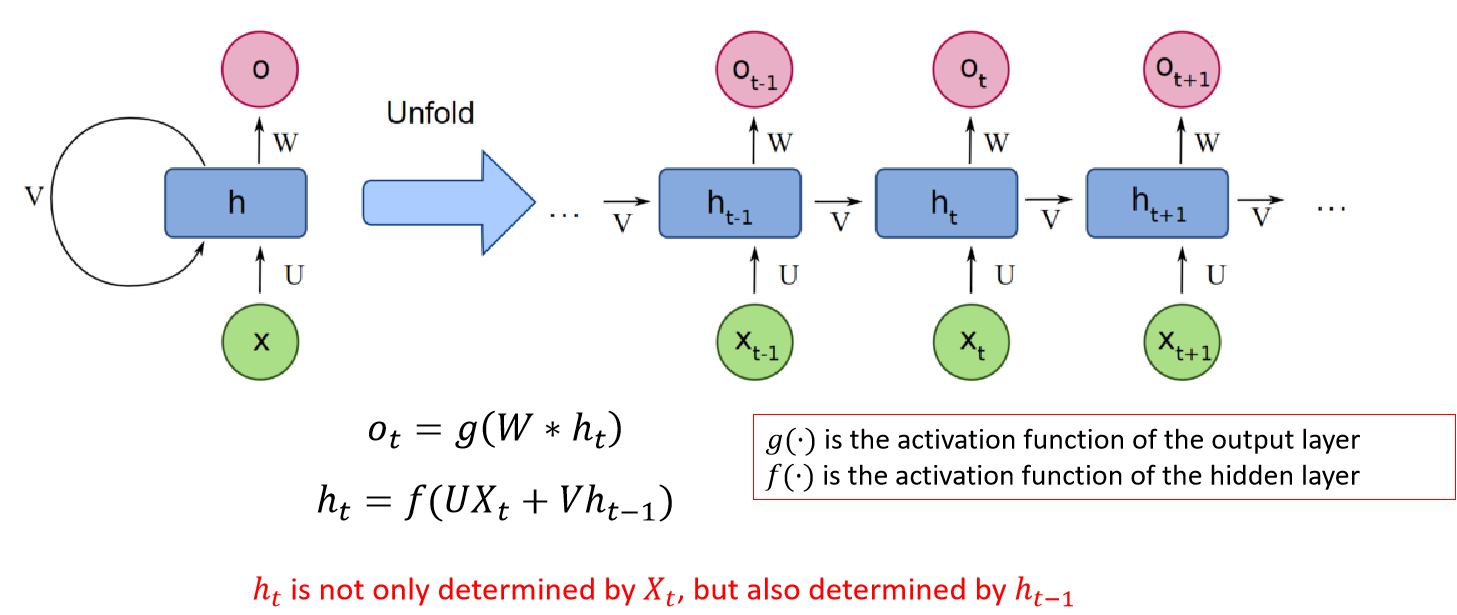
****

图 1 RNN的结构图

**Seq2Seq模型：**Seq2Seq模型是输出的长度不确定时采用的模型，广泛应用与机器翻译任务中，这是因为翻译的原文和译文长度不一定相同，所以输出的长度就不确定了。Seq2Seq模型由两个RNN构成，一个负责Encoder，另一个负责Decoder（如图 2）。Encoder负责将输入序列压缩成指定长度的向量，该向量作为序列的整体语义，这个过程称为编码，Decoder负责根据语义向量生成指定的序列，这个过程称为解码。

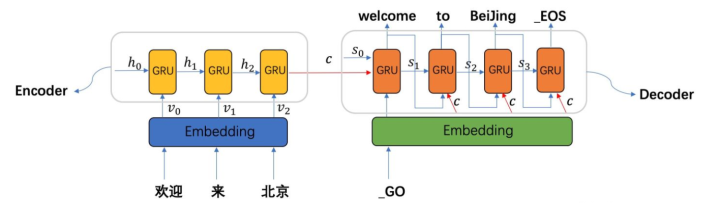


图 2 GRU网络结构

**GRU模型：**GRU在传统的RNN网路上增加了两个门。其中为控制重置的门控（reset gate），为控制更新的门控（update gate）。每一层的和更新方式如图 3所示。GRU最关键的一个步骤是“更新记忆”在这个阶段，GRU同时进行了遗忘了记忆两个步骤。我们使用了先前得到的更新门控（update gate）。更新表达式：。门控信号（这里的）的范围为0~1。门控信号越接近1，代表记忆下来的数据越多，而越接近0则代表遗忘的越多。GRU使用同一个门控同时进行遗忘和选择记忆，因而比LSTM要快，效果往往也更好。

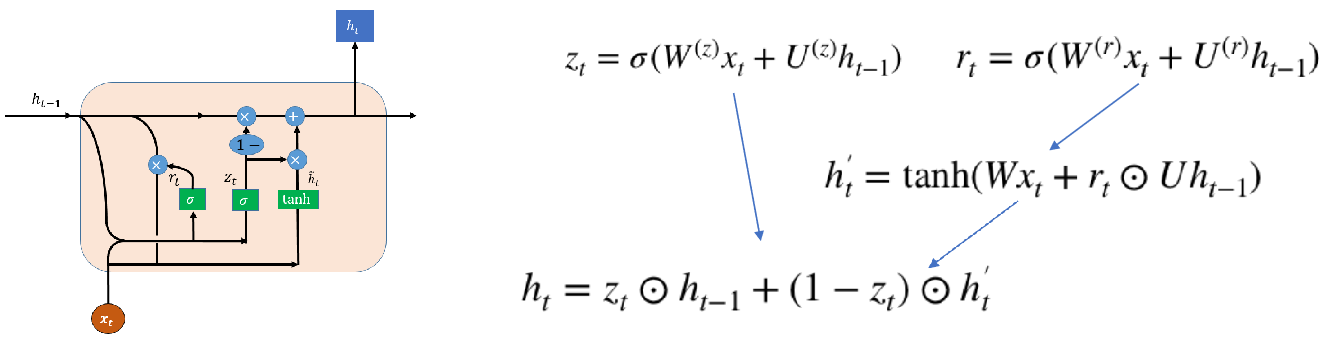


图 3 GRU的模型结构

1. 网络框架结构搭建介绍
   1. 编码器（Encoder）网络架构

如图 4所示，对于翻译句子对中的每一个法语单词，编码器为其生成一个256维的Embedding向量，将该向量作为输入传入RNN网络。隐藏层的维度也为256，因而RNN网络输出一个256维的向量。编码器将这个向量输出，同时输出一个256维的隐藏层向量。

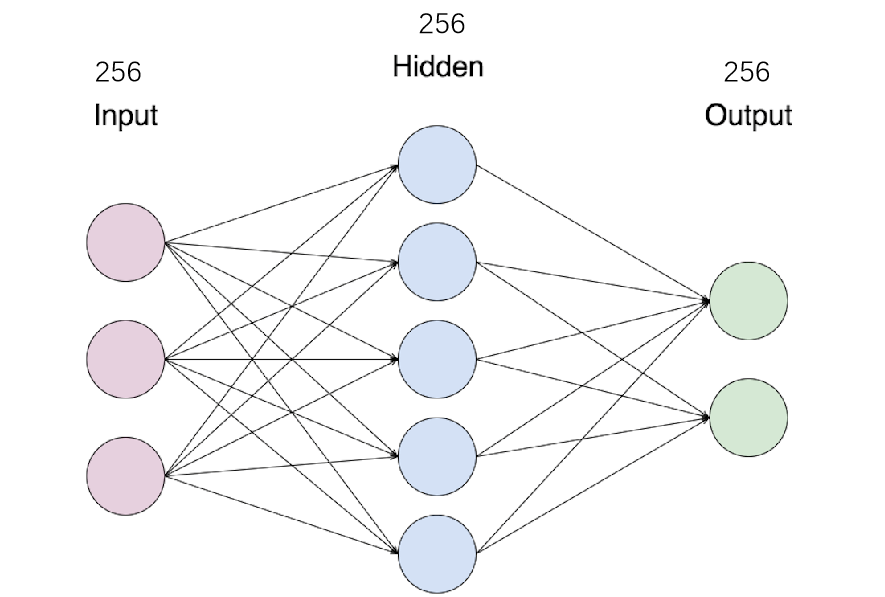


图 4 Encoder中的RNN结构

* 1. 解码器（Decoder）网络架构

如图 5，对于翻译句子对中的每一个英语单词，解码器为其生成一个256维的Embedding向量，将该向量作为输入传入RNN网络。隐藏层的维度也为256，因而RNN网络输出一个256维的向量。然后将该输出通过一个MLP映射到一个1093（数据集中英语单词的总个数）维的向量，并使用softmax函数，得到解码器的最终输出。解码器还同时输出了RNN的256维隐藏层向量。

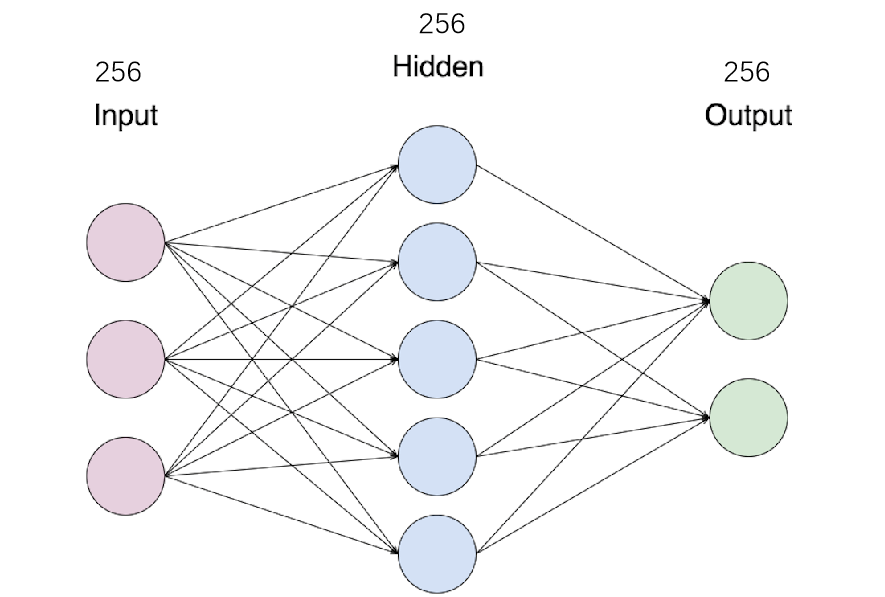


图 5 Decoder中的RNN结构

* 1. 整体网络框架

如图 6，对于翻译句子对中的每一个法语单词，输入Encoder，将Encoder的隐藏层输入到Decoder的隐藏层中，同时将翻译句子对中的每一个英语单词输入的Decoder中，Decoder的输出作为预测。

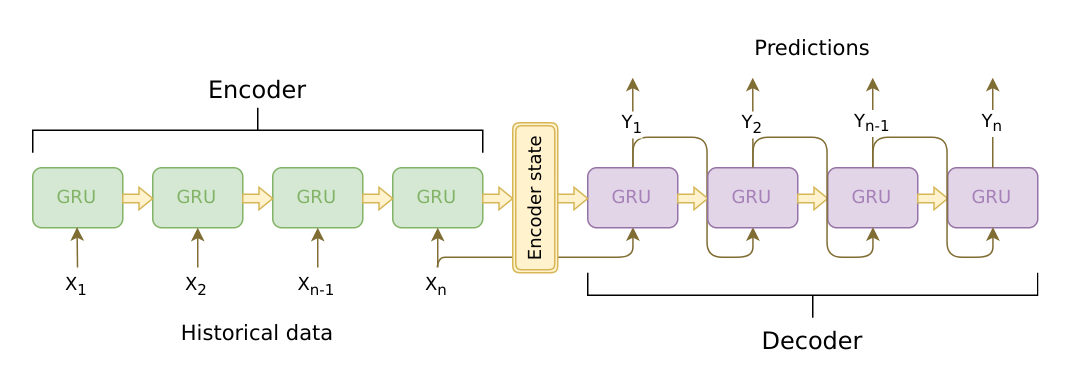


图 6 整体网络框架结构

1. 核心代码介绍

|  |
| --- |
| **输入：**训练集eng-fra  神经网络Encoder  神经网络Decoder  Adam优化器optim  NLLloss损失函数criterion  **过程：**   1. 为数据集中的每一个法语单词和英语单词设置索引 2. for n\_iters 1 to 55000 3. 从数据集中随机选取一个法语-英语句子对 4. 将句子对中每一个法语单词传入Encoder,得到e\_output和e\_hidden 5. 将句子对中每一个英语单词和e\_hidden传入Decoder,得到d\_output和d\_hidden 6. losscriterion(decoder\_output,target) 7. BP算法更新Encoder和Decoder的权重w和偏置b 8. end for 9. 从数据集中随机选取10个句子对进行测试   **输出：**loss,折线图,测试结果 |

1. 训练执行流程介绍

总共训练55000次。每次训练都计算训练集上的损失，每训练5000次打印一次训练结果（包括损失值，完成进度，时长等），每100次记录一次loss,在训练完毕后绘制对应的loss曲线图。

# 八、实验数据及结果分析：

测试需要给出测试用例（从正常，边界，错误等各方面给出测试用例，建议用3个表格的形式给出3种不同类型的测试用例），贴图（运行结果截屏），分析。做看图说话。每一图都要给出图名。有表则给出表名。请根据实验实际，用大量语言进行描述讨论。

1. 数据介绍：

本次实验使用eng-fra数据集，共有39365对英语-法语句子对，在限定句子最大长度为10后，共整理得到1271个句子对，共有1418个法语单词，1093个英语单词。每次训练或者测试都从1271个句子对中，随机选择一对，并将这对句子按照每个单词的编码进行向量化处理。

1. 结果分析：

测试的时候，从pairs中随机选择10对，通过比较网络翻译和真实翻译来评估网络的效果。

* 1. 使用RNN训练结果：

> tu n es pas si interessant .

= you re not that interesting .

< you re not that interesting . <EOS>

> je vais preparer a diner .

= i m going to fix some dinner .

< i m going to buy some bread . <EOS>

> elle est l editeur en chef .

= she is the editor in chief .

< she is the executive editor . <EOS>

> nous n allons pas le jeter .

= we re not throwing it away .

< we re going to do everything we can . <EOS>

> je me trouve dans une situation desesperee .

= i m in a desperate situation .

< i m confused enough as it is . <EOS>

> je suis desolee a propos de hier soir .

= i m sorry about last night .

< i m sorry about last night . <EOS>

> tu n es pas aussi maligne que moi .

= you re not as smart as me .

< you re not as smart as me . <EOS>

> j oublie toujours les noms des gens .

= i m always forgetting people s names .

< i m always forgetting people s names . <EOS>

> vous ne trompez personne .

= you re not fooling anybody .

< you re very sophisticated . <EOS>

> ceci m est indifferent .

= i am not concerned with this .

< i am not concerned with this affair . <EOS>

这10个测试样本中只有4个可以翻译正确，但是对于一些简答的单词（比如I am,he s,we re）等还是能够翻译正确。图 7为RNN训练时的loss曲线图，可以看到最后的loss在0.7左右，因而测试时翻译的准确度不高。

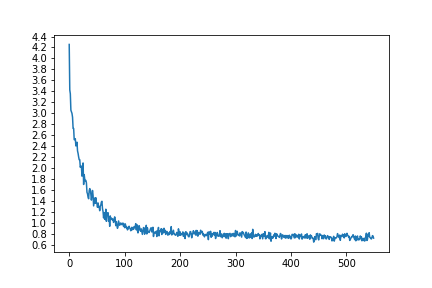


图 7 RNN训练时的loss

* 1. 使用pytorch提供的GRU模型的训练结果

> on te reclame au telephone .

= you are wanted on the phone .

< you are wanted on the phone . <EOS>

> tu n es pas comme les autres .

= you aren t like the others .

< you aren t like the others . <EOS>

> elle aime beaucoup les fleurs .

= she is very fond of flowers .

< she is very fond of flowers . <EOS>

> il est plus intelligent que moi .

= he is more clever than i am .

< he is more clever than i am . <EOS>

> il n est pas aussi grand que toi .

= he s not as tall as you are .

< he s not as tall as you are . <EOS>

> je ne suis pas inquiet pour l argent .

= i m not worried about money .

< i m not worried about money . <EOS>

> nous allons travailler ce soir .

= we re going to work tonight .

< we re going to work tonight . <EOS>

> c est un bon mari pour moi .

= he is a good husband to me .

< he is a good husband to me . <EOS>

> je ne suis pas d humeur a blaguer .

= i am in no mood for joking .

< i am in no mood for joking . <EOS>

> je ne suis pas concerne par cette affaire .

= i am not concerned with this affair .

< i am not concerned with this affair . <EOS>

可以看到pytorch提供的GRU网络在测试的10个样本中全部翻译正确，该网络在训练过程中的loss如图 8所示，该网络在训练最终的loss非常接近0。相比于自己搭建的GRU网络，pytorch内置的GRU模型有更多的优化模块，能够快速的让loss下降，达到一个较好的效果。查询pytorch内置GRU网络结构，发现pytorch中GRU的r和z的权重为经过MLP得到的维度为隐藏层维度的向量，而自己设计的GRU得到的维度为1。这导致权重丢失大量的信息，因而GRU的记忆和遗忘功能不能很好的体现。

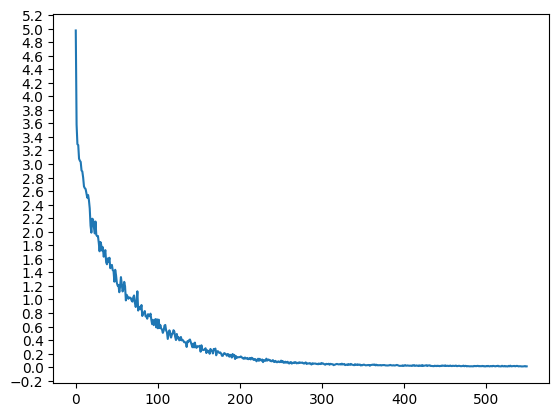


图 8使用pytorch提供的GRU模型的训练loss

# 九、总结及心得体会：

# 十、对本实验过程及方法、手段的改进建议及展望：

**报告评分：**

**指导教师签字：**