



同濟大學
TONGJI UNIVERSITY

同济大学《深度学习》 课程实验报告

基于 Large Movie Review 数据集的
情感分类实验

任课老师：范睿

班级：10069801

2023 年 12 月

摘要

本实验旨在使用 Large Movie Review 数据集，通过手动构建几种经典的循环神经网络对电影的评论进行情感分类，将电影评论分为积极和消极两类。同时本实验的训练过程进行了可视化，记录了几种网络训练过程的 acc 和 loss；还在测试集上对于几种模型进行了 Accuracy、Recall、Precision、F1-score 等指标的评估。

关键词：深度学习，神经网络，RNN,LSTM,GRU

目 录

摘要.....	I
第一章 实验内容简述	1
1.1 问题描述	1
1.2 问题背景	1
1.3 数据集介绍	1
1.4 实验环境	1
第二章 核心算法	3
2.1 数据集的处理	3
2.1.1 构建分词器	3
2.1.2 构建词汇表	3
2.2 神经网络的搭建	3
2.2.1 RNN 网络	3
2.2.2 LSTM 网络	4
2.2.3 GRU 网络	4
2.3 优化器	5
2.4 损失函数	5
2.5 超参数的设置	5
第三章 实验结果	7
3.1 训练过程	7
3.2 评估过程	8
第四章 实验总结	9
4.1 实验心得及后续展望	9
参考文献	10

第一章 实验内容简述

1.1 问题描述

在自然语言处理领域，情感分类是一个非常重要的研究问题，本实验旨在利用 Large Movie Review 数据集，通过手动构建经典的循环神经网络 (RNN)、长短时记忆网络 (LSTM)、门控循环单元 (GRU) 等模型进行情感分类。任务的核心是对电影评论进行分类，判断评论的情感倾向。选择手动搭建模型是为了更深入地理解它们的工作原理，通过实验和比较不同模型在这一任务上的表现，深入研究它们在电影评论情感分类中的性能和效果差异，最终目标是建立一个具有良好泛化能力的模型，能够准确地判断未见过的电影评论的情感倾向。

1.2 问题背景

情感分类问题在自然语言处理领域扮演着重要的角色，情感分析在影视产业、社交媒体分析等领域有着广泛的应用，通过对电影评论进行积极、消极或中性的情感分类，有助于理解用户的意图、进行情感分析，从而提供更智能的用户体验。Large Movie Review 数据集提供了一个丰富的文本资源，其中包含大量的电影评论，并且每个评论都标注了情感倾向。

通过深入挖掘不同循环神经网络模型在这一任务上的表现，可以更好地理解它们对文本情感的建模能力。这有助于进一步改进情感分类模型，提高对用户情感的敏感度，从而更好地服务于各种应用场景。

1.3 数据集介绍

Large Movie Review 数据集，常被称为 IMDb 数据集，是一个用于情感分析的经典文本分类数据集之一。该数据集由斯坦福大学的 Andrew Maas 及其团队于 2011 年创建。^[1] 它包含来自互联网电影数据库 (IMDb) 的 50,000 条电影评论，被分为 25,000 条训练集和 25,000 条测试集，每个集合中均包含正面和负面情感的评论，无论是训练集还是测试集，其中的正/负类（即积极/消极）样本个数均相同，均为 12500 个。

这个数据集的主要用途是训练和评估自然语言处理 (NLP) 模型，特别是用于情感分析的模型。由于 IMDb 评论的真实性和多样性，这个数据集成为评估情感分析算法性能的重要基准之一。

1.4 实验环境

本次实验在本地进行代码的编写与调试，在 autodl 网站中租用 GPU 进行训练，以下为本实验的基本环境：

项目	内容
CPU	12 vCPU Intel(R) Xeon(R) Platinum 8352V CPU @ 2.10GHz
GPU	Nvidia RTX 4090
内存	90GB
OS	Linux (ubuntu20.04)
CUDA	11.3
编程语言	Python 3.8
框架	pytorch 1.10

配置环境过程中出现的问题

在配置实验所需要的库时，发现在用 `pip install torchtext` 时候，会自动安装新版本的 `pytorch`，导致其他的一些包出现版本不匹配的问题，所以正确的做法应该是到 `pytorch` 官网或者是 Github 寻找对应的版本，在本实验中对应 `pytorch 1.10` 版本的 `torchtext` 是 `0.11.0`。

第二章 核心算法

2.1 数据集的处理

对于 nlp 任务来说，正确地处理数据集是非常重要的步骤，对于本实验来说，主要在于 dataset 和 dataloader 的编写。而在 dataset 部分，关键步骤是分词器和词汇表的构建。

2.1.1 构建分词器

分词是将文本分割为单词或子词的过程，是文本处理的基本步骤。通过分割文本，将连续的序列划分为模型可以理解的单元，如单词或标记。基本分词以空格或标点为基准，而进阶技术如词干提取和词形还原进一步处理单词的不规则形式。分词为文本处理奠定了基础，为后续任务如文本表示、特征提取和深度学习模型训练提供了关键支持。

2.1.2 构建词汇表

构建词汇表则是将数据集中所有唯一单词映射到唯一标识符的过程。这个词汇表为模型提供了对文本数据的数字化表示，为后续的处理和分析奠定基础。

2.2 神经网络的搭建

这部分主要是介绍了本次实验所用到的一些网络，其中包括经典的 RNN 网络，LSTM 网络与 GRU 网络共三种网络。

2.2.1 RNN 网络

循环神经网络 (Recurrent Neural Network, RNN) 是一类具有循环连接结构的神经网络，主要用于处理序列数据和时间序列数据。相较于传统的前馈神经网络，RNN 引入了循环连接，允许信息在网络内部持续传递，使其能够捕捉和处理序列中的依赖关系。^[2]

RNN 的核心思想是在网络的隐藏层中引入一个时间维度，使得网络能够接受来自上一时间步的输入信息，并将其作为当前时间步的输入，从而在处理序列数据时具有记忆能力。这种结构使得 RNN 在处理不同长度和变长序列时更加灵活，适用于自然语言处理、语音识别、时间序列分析等领域。

但是传统的 RNN 也存在梯度消失和梯度爆炸等问题，导致难以有效学习长期依赖关系，这在后面的实验结果中也有所体现下面是 RNN 网络的一个基本单元的结构图：

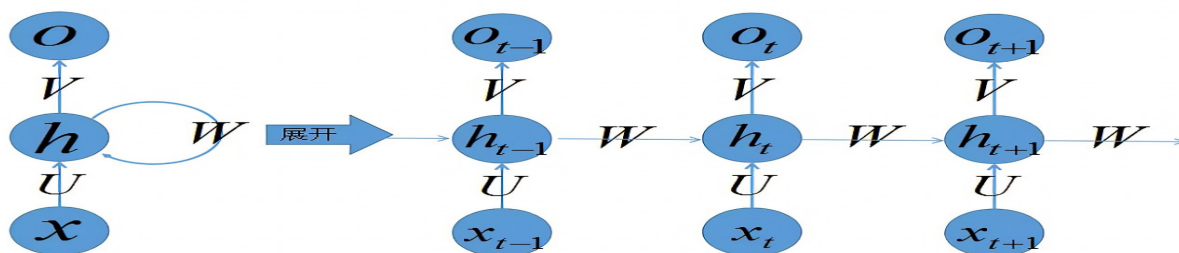


图 2.1: Structure of the RNN network

2.2.2 LSTM 网络

LSTM（长短时记忆网络）是由德国计算机科学家 Sepp Hochreiter 和 Jürgen Schmidhuber 于 1997 年提出和开发的^[3]。他们的目标是克服传统循环神经网络（RNN）在处理长序列数据时容易出现的梯度消失和梯度爆炸的问题。通过引入门控机制和记忆单元的概念，LSTM 成功地解决了这些问题，使得神经网络能够更有效地捕捉和利用长期依赖性，从而在许多序列建模任务中取得显著的成果。

LSTM 网络包含一个记忆单元（memory cell），该单元负责存储和更新信息。此外，它由三个门（gates）组成，分别是输入门（input gate）、遗忘门（forget gate）和输出门（output gate）。这些门控制着信息的流动，使得网络能够选择性地记住或遗忘先前的信息，并将当前的输入整合到记忆中。

按照自己的理解，输入门负责控制新的信息的传入，遗忘门决定要从记忆单元中删除哪些信息，而输出门则确定要输出的记忆。这种门的设计允许 LSTM 网络在处理长序列时更好地捕捉到关键的上下文信息，从而提高了其在各种任务，如语音识别、自然语言处理等方面的性能。

下面是 LSTM 网络一个基本单元的结构图：

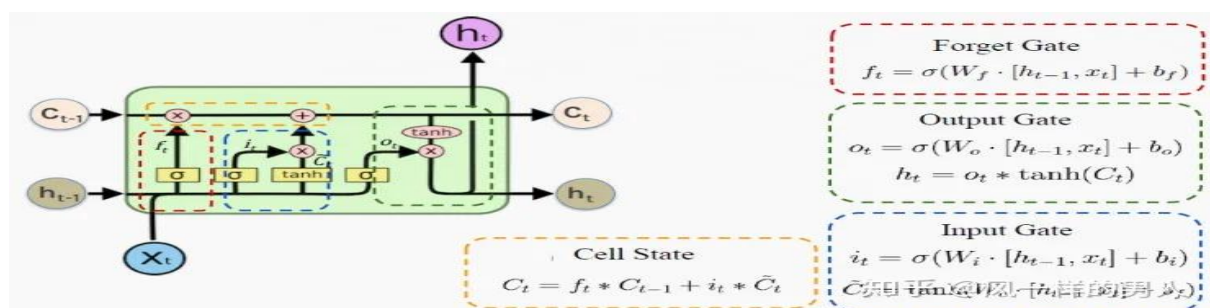


图 2.2: Structure of the LSTM network

2.2.3 GRU 网络

门控循环单元（Gated Recurrent Unit，简称 GRU）是一种类似于长短时记忆网络（LSTM）的循环神经网络（RNN）变体，是由 Cho 等人于 2014 年提出。^[4]GRU 旨在简化 LSTM 的结构，减少其参数数量，同时保持对长期

依赖性的良好建模能力。

GRU 与 LSTM 相似，也包含一个门控机制，但相较于 LSTM 的三个门（输入门、遗忘门、输出门），GRU 只有两个门，即更新门（update gate）和重置门（reset gate）。这两个门允许网络选择性地更新记忆和重置记忆的部分内容，从而实现对序列信息的灵活捕捉。

更新门决定了当前时刻的记忆状态需要保留多少先前的记忆，而重置门则控制了网络在当前时刻应该考虑多少先前的信息。这使得 GRU 在一定程度上降低了模型的复杂性，减少了参数数量，从而更容易训练，在一些任务上表现非常出色。

下面是 GRU 网络一个基本单元的结构示意图：

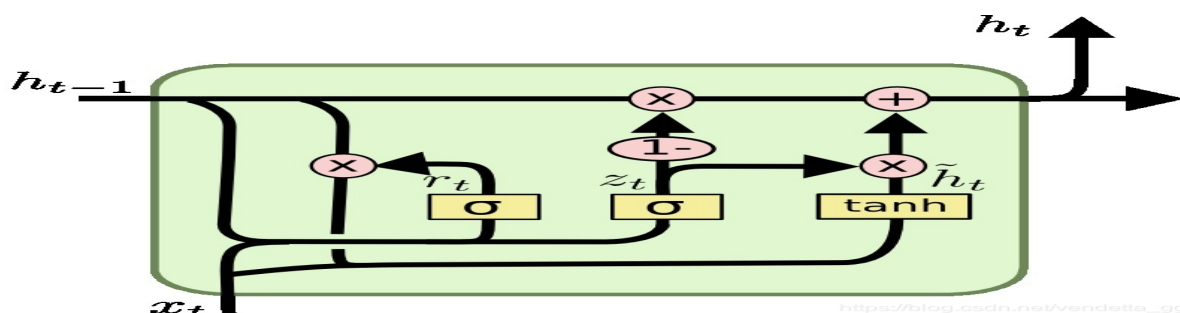


图 2.3: Structure of the GRU network

2.3 优化器

优化器的主要任务是通过根据损失函数的梯度信息调整模型参数，实现最小化损失函数的目标。本次实验采用了 SGD 优化器进行模型的训练。SGD (Stochastic Gradient Descent) 是深度学习中基础的优化算法之一。它通过随机选择样本来计算梯度并更新模型参数，逐步优化目标函数。这种简单直接的方法使得 SGD 在训练神经网络时非常高效。通过不断迭代样本和更新参数，SGD 帮助模型逐步学习并提高性能。

2.4 损失函数

在本次实验中，采用的是交叉熵损失函数。

2.5 超参数的设置

在本实验中，训练过程中涉及到的超参数包括训练轮数 epoch, batch-size, 优化器的学习率 learning rate, SGD 优化器中用到的动量 momentum 和 weight decay。对于循环神经网络来说，主要涉及到的超参数有 vocab size, embedding dim 和 hidden dim 等。在本次实验中，选定训练轮数为 50 轮，learning rate 设置为 0.005，但是后续经过实验发现，这两个参数选用的并

不是很好，对于本次实验来说 50 轮明显偏少，而学习率 0.005 明显比较小，使得除了 LSTM 网络（包括 LSTM 的初期）acc 基本不变化，loss 也不下降，希望在后续重新进行实验，探索合理的轮数和学习率以及其他的一些参数。

第三章 实验结果

3.1 训练过程

本次实验在原有的训练集上以 0.8:0.2 的比例重新划分了训练集和验证集。下面是对于三种网络（RNN 网络，LSTM 网络，GRU 网络）训练集上和验证集上 acc 与 loss 变化情况的记录。

首先是训练集上的变化趋势：

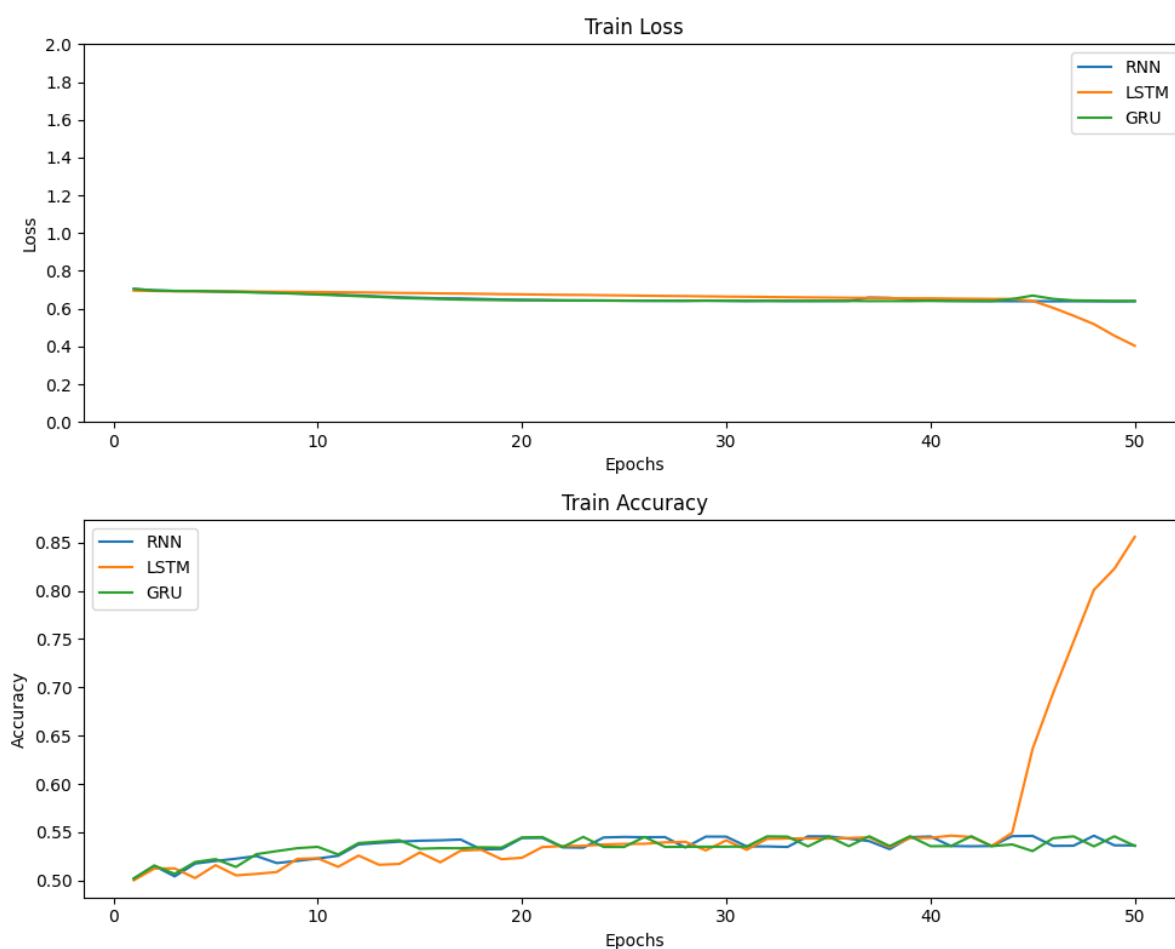


图 3.1: training loss and training accuracy

从两幅图中可以明显看出，当训练轮次（epoch）为 50 的时候，三种网络在前期表现都比较一般，acc 上升速度很慢或基本不上升，loss 不下降，而当快训练结束时,LSTM 突然发生了 acc 的上升和 loss 的下降，最终在训练集上达到了 85 的正确率，在验证集上也达到了 80 的正确率。

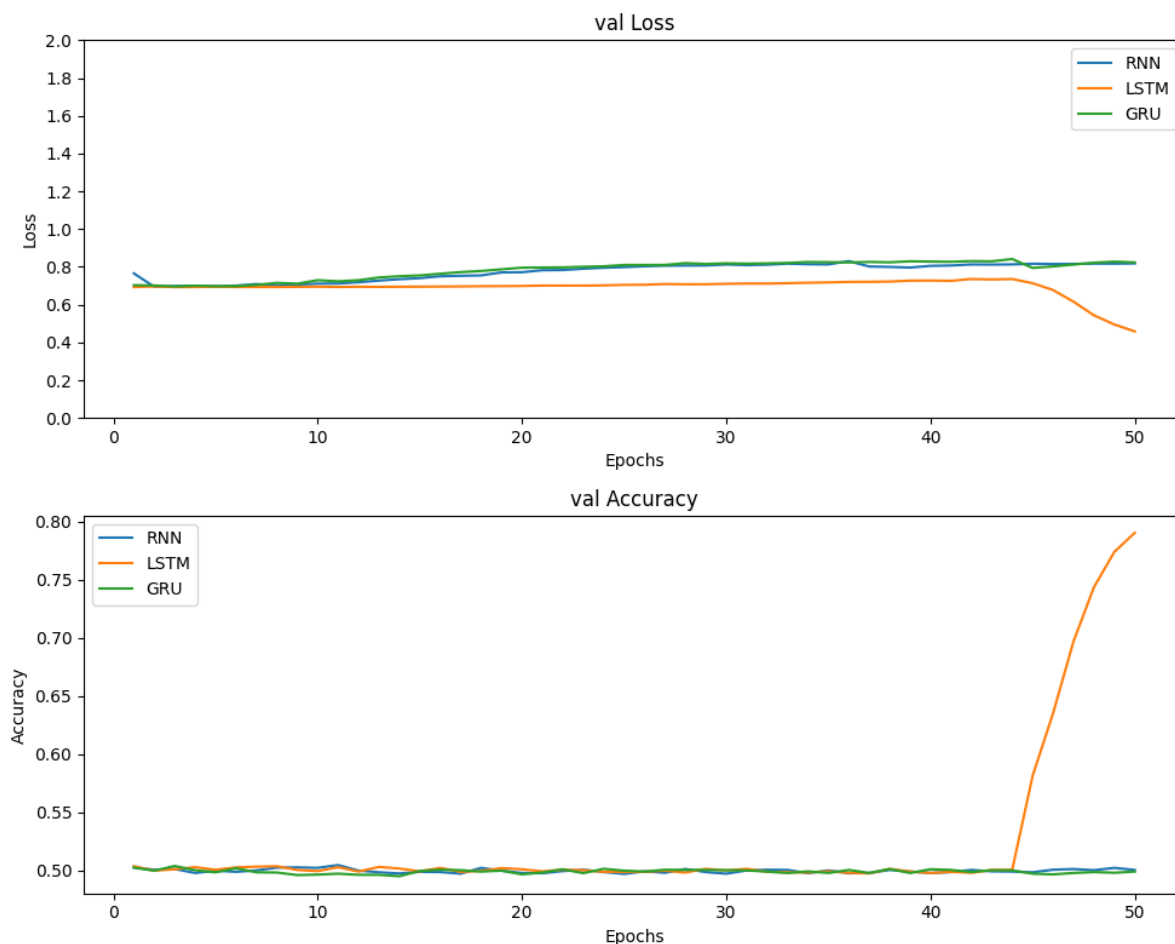


图 3.2: val loss and val accuracy

3.2 评估过程

该部分展示了对三种网（RNN 网络，LSTM 网络，GRU 网络）在测试集上进行评估，通过 Accuracy、Recall、Precision、F1-score 这四种指标进行测评。这里是通过调用 `sklearn.metrics` 中内置的函数来计算这几个指标。

网络 \ 指标	Accuracy	Recall	Precision	F1-score
RNN	0.5012	0.5012	0.5036	0.5029
LSTM	0.8043	0.8043	0.8046	0.8044
GRU	0.5058	0.5058	0.5071	0.5074

第四章 实验总结

4.1 实验心得及后续展望

在这个实验中,我尝试了手动搭建了几个经典的循环神经网络模型,包括 RNN 网络、LSTM 网络以及 GRU 网络,并利用这三种网络进行基于电影评论的情感分类实验。

但是本次实验质量比较差,原因主要是没有选择合适的轮数与学习率,没有进行多次对比,导致了 RNN 和 GRU 两种网络 acc 都不高, LSTM 虽然在训练集上达到了 85 的 acc,在验证集和测试集上基本达到了 80 的 acc,但是由训练过程中的 acc 和 loss 变化分析可以说明,如果增加轮数, LSTM 肯定会有更好的表现。

后续希望可以多进行实验,寻找合适的轮数和学习率,以及进行学习率的动态调整等策略;同时除了这三种模型之外,尝试搭建 transformer 和 bert 等更为复杂的神经网络,进行更为复杂的 nlp 任务。

参考文献

- [1] A L Maas, R E Daly, P T Pham, et al. Learning Word Vectors for Sentiment Analysis[C/OL]//Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, Oregon, USA: Association for Computational Linguistics, 2011: 142–150. <http://www.aclweb.org/anthology/P11-1015>.
- [2] Z C Lipton, J Berkowitz, C Elkan. A critical review of recurrent neural networks for sequence learning[J]. ArXiv preprint arXiv:1506.00019, 2015.
- [3] S Hochreiter, J Schmidhuber. Long Short-Term Memory[J]. Neural Computation, 1997, 9(8): 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.
- [4] J Chung, C Gulcehre, K Cho, et al. Empirical evaluation of gated recurrent neural networks on sequence modeling[J]. ArXiv preprint arXiv:1412.3555, 2014.