Central University of Finance and Economics

中央财经大学

非结构化大数据分析

基于机器学习的《哈利波特》评论情感分类

吴宇翀

2021210793

统计与数学学院

15910852867

email@wuyuchong.com

指导老师：刘苗

2022 年 6 月 28 日

摘要

通过网络爬虫的方式获取《哈利波特》全系列图书的评论进行文本分析，我们使用主题模型进行对评论进行文本挖掘，之后进行文本情感分类模型的训练。在文本预处理阶段，我们尝试使用词编码和词向量的方式，在训练阶段，我们构建了 DNN、LSTM、BERT 等多个深度学习模型进行训练，并进行了模型比较，最终达到了 88% 的准确率。最后，为了进一步实现在超大文本集上进行训练，我们使用基于 Spark 的分布式算法在集群服务器上进行训练测试。[1]

模型	计算配置	用时	准确率	可拓展性
tokenize + DNN	阿里云服务器 Xeon 8 核 CPU 32G 内存	3 分钟	55%	低-单机
Word2Vec + LSTM	阿里云服务器 Xeon 8 核 CPU 32G 内存	1 小时	70%	低-单机
bert - 小型	阿里云服务器 Xeon 8 核 CPU 32G 内存	24 分钟	78%	低-单机
bert - AL	阿里云服务器 Xeon 8 核 CPU 32G 内存	1.6 小时	82%	低-单机
bert - 标准	阿里云服务器 Xeon 8 核 CPU 32G 内存	1 小时	88%	低-单机
spark - logit	中央财经大学大数据高性能分布式集群	4 分钟	73%	高-集群
spark - 决策树	中央财经大学大数据高性能分布式集群	10 分钟	80%	高-集群
spark - 梯度助推树	中央财经大学大数据高性能分布式集群	15 分钟	82%	高-集群
spark - 随机森林	中央财经大学大数据高性能分布式集群	30 分钟	87%	高-集群

[1] 分布式模型在该小型数据集上没有优势，进行此项的意义在于对大型文本数据集可拓展性的技术储备，仅有在文本量级超过单机可承载上限时，分布式计算才具备意义

目录

1 数据爬取

我们使用 python 对豆瓣上《哈利波特》全系列图书的评论进行爬取。由于豆瓣的评论机制要求读者撰写评论时必须同时填写评价级别，因此我们可以同时获取到评价文本及其对应的情感级别。豆瓣将评价级别分为三类：好、坏、中等，这有利于我们进行通过文本的内容预测评价级别的有监督机器学习训练。

在爬取过程中，我们使用已登录的 cookie 信息和浏览器信息提交请求，并在每两次请求的间隔加入随机时间的休眠，以尽可能地模拟真实人的操作，避免 IP 被官方查封。

评论	评分
每翻一页便是与你的告别。我不想同你...	good
几年后重新看了最后的几章，在这热得...	good
所有爱唱反调的胖友们，你们都没试过...	good
基本上一如预想。罗琳笔下的主要角色...	good
别的还好，双胞胎死一个活一个实在太...	medium
这本稀碎情节太多，完全都是没必要的...	medium
反正总是要看完的嘛！	medium
罗琳忽悠了全球读者	medium
对不起，我实在是无法接受这个结局。	bad
从哈一到哈七。。。水平依次下降。。...	bad

图 1: 词云图

2 文本预处理

我们使用 jieba 对中文进行分词处理。

1. 去除标签
   - 将一些网页 HTML 特有的标签进行去除，如段落标记、换行标记 `p` `br` 等
2. 去除标点符号
   - 将常用标点符号进行去除，如 ！ ； 等
3. 去除多余的空格
   - 删除无意义的连续性空格
4. 去除数字
   - 由于数字对文本情感识别作用小，我们选择将其删去
5. 去除停用词
   - 对意义较小的常用词进行删除
6. 去除过短的词汇
   - 由于英文中过短的字符一般意义较小，我们选择将其删去
7. 大小写统一
   - 大小写代表同一词汇，需要进行统一

表 3: 关键词提取

text	rating	clean_text	label
对不起，我实在是无法接…	bad	对不起	1.0
从哈一到哈七。。。水平 …	bad	从哈一到免不了	1.0
这套书根本就是在宣扬扭曲…	bad	根本就是价值观	1.0
翻译太烂。和上一本一样…	bad	比不上	1.0
糟透了	bad	糟透了	1.0

3 主题模型

3.1 主题模型的构建

主题模型是一个相对泛化的概念，从实现了文本数据的结构化的文本模型都能称为主题模型，狭义上一般代指基于隐式 Dirichlet 先验概率分布模型。

基于主题模型，每一篇文档都是一个主题的多项分布，文档集合事实上可以看作是一个天然的软聚类过程，各个主题就是聚类中心，文档在各个主题上的概率就是它与这个聚类中心的距离。同时，主题模型可以得到各主题下词汇的概率分布，主题-词汇概率矩阵旋转后就可以得到词汇在各个主题的概率分布情况，得到词汇的软聚类结果。

LDA 模型通过将生成的随机文档与真实文档进行相似度对比来决定狄利克雷先验分布的参数，从而得到最优参数的模型。

3.2 结果分析

我们构建一个 3 个主题的主题模型，并绘制解释图，图中每个气泡代表一个主题，越大的气泡代表该主题涵盖的评论数越多。蓝色柱子代表特定词汇在整个语料库中的词频，红色柱子代表特定词汇在特定话题中的词频。

3.2.1 主题一

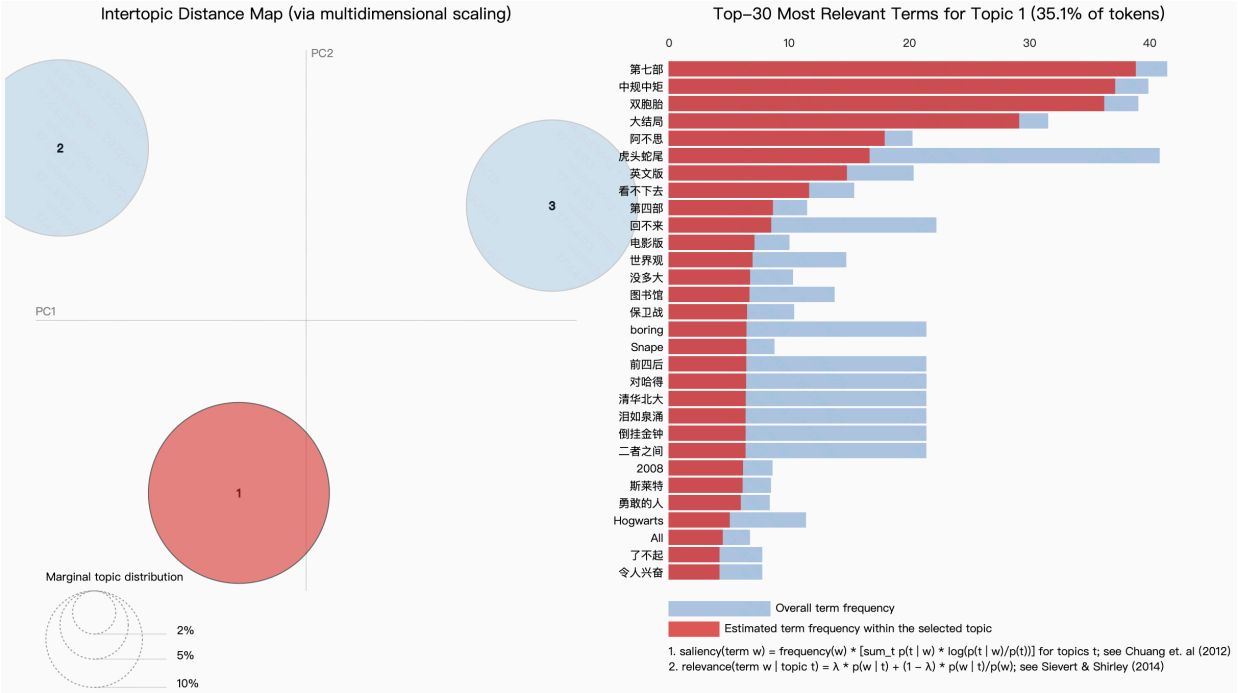

图 2: 主题一成分词汇权重图

第一个话题主要是对大结局的评论。图书馆保卫战、第七部、虎头蛇尾都是对结局的讨论。

3.2.2 主题二

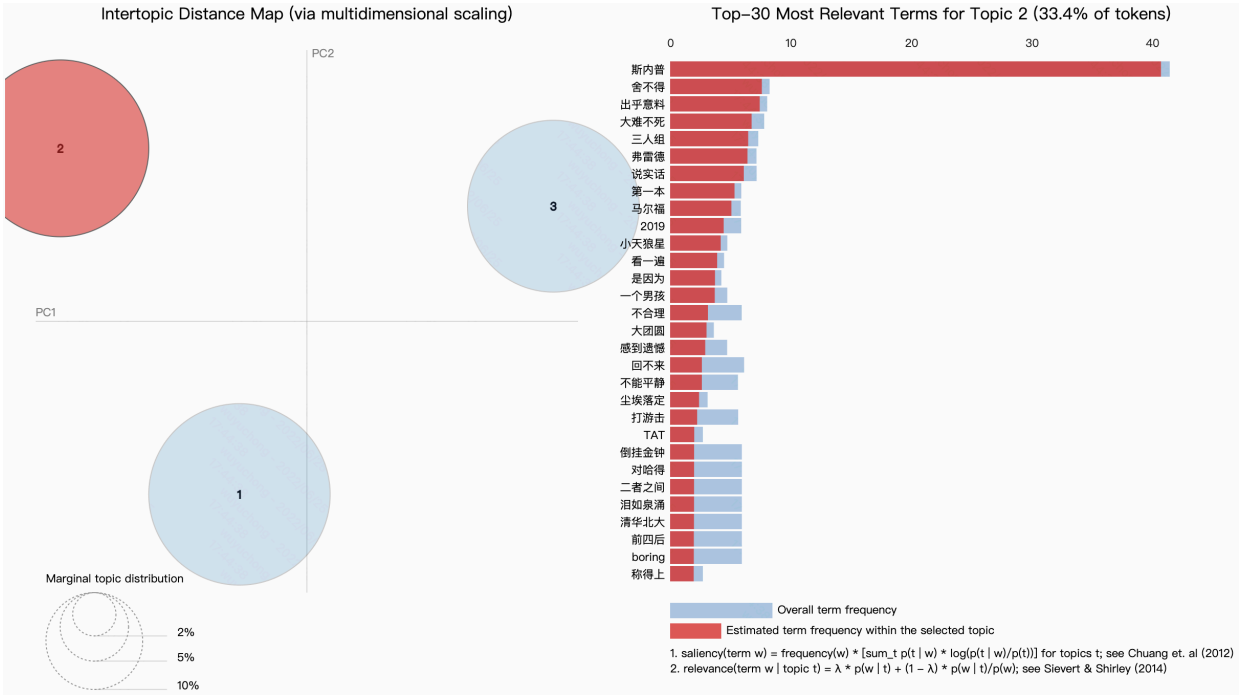

图 3: 主题二成分词汇权重图

第二个话题主是对书中各类较为主要的人物进行讨论。排名第一的词汇斯内普是书中以为颇具争议的反派人物，其性格方面的多重性给了读者非常大的讨论空间，弗雷德、小天狼星也都是书中较为典型的人物，他们与哈利波特具有密切地联系，

3.2.3 主题三

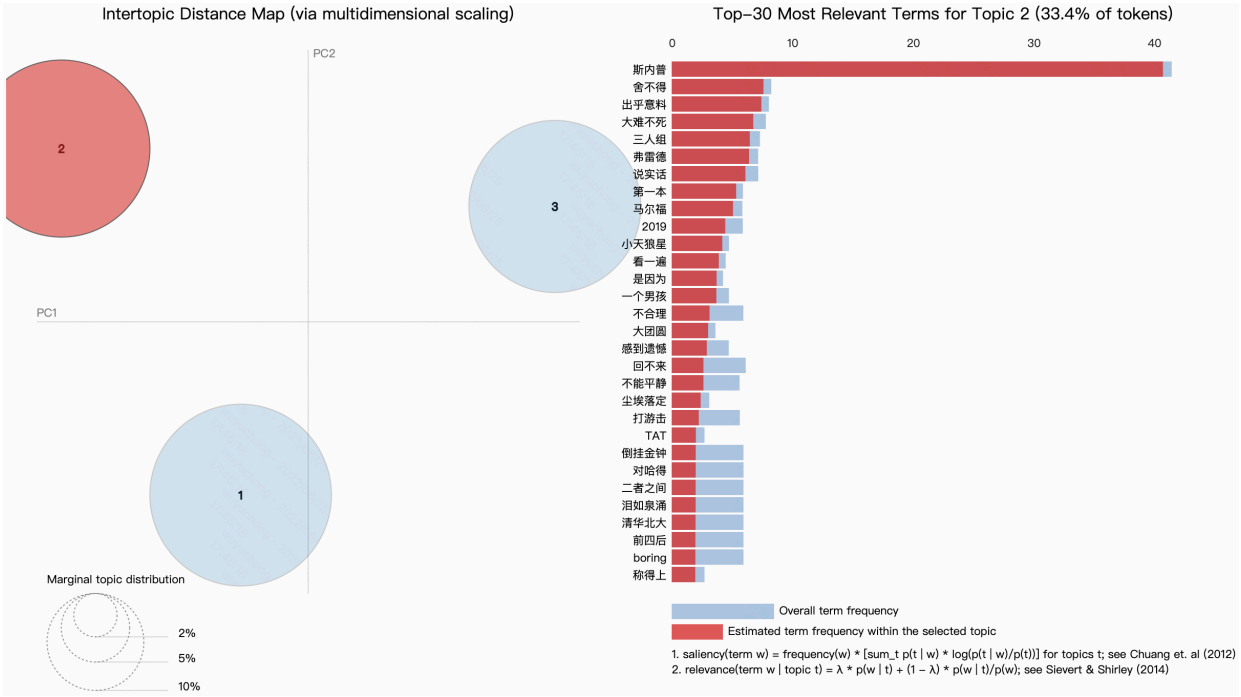

图 4: 主题三成分词汇权重图

第三个话题主要围绕最主要人物哈利波特和成长历程中的重要节点剧情进行讨论。英雄主义、火焰杯、打游击都体现了主角的主线剧情。

当然，读者的讨论并非总是局限于同一个话题，在特定的话题下有会参杂着许多对全书的感受、见解看法等等。

4 深度学习

4.1 数据处理

在进行文本情感分类的有监督训练中，我们将评级级别为差和中等的两类合并为消极一类，将好评作为积极一类。

1. 标签处理：分类标签由类别名称转为数字。
2. 数据集划分：在总体 5 万条文本中随机划分 20% 的测试集，再从训练集中划分 20% 的验证集。
   - 使用训练集的文本进行模型训练
   - 使用验证集的文本进行模型超参数的调整
   - 使用测试集的文本进行模型效果评价
3. 数据集格式转换：使用自动的缓冲区大小，使用 32 的 `batch size`。
   - batch size 为一次训练所抓取的数据样本数量
   - 分批训练相对于直接对全训练集训练的好处在于：提高了每次迭代的训练速度、利于多线程训练、使得梯度下降的方向更加准确
   - batch size 的大小与模型的收敛速度和随机梯度噪音有关
   - 当 batch size 过小时，在一定的迭代次数下，模型来不及收敛
   - 当 batch size 过大时，一方面容易出现内存紧缺，另一方面模型的泛化能力会变差

4.2 Tokenize + DNN

4.2.1 DNN 模型结构

我们搭建了一个三层神经网络用于训练。

表 4: 搭建的神经网络结构（总参数个数：775681）

神经网络层	神经元个数	参数个数
输入层	512	512512
drop out (50%)	0	0
中间层	512	262656
drop out (50%)	0	0
输出层	1	513

4.2.2 模型训练

正常情况下，随着训练迭代次数的增加，损失函数逐渐减小，对训练集的拟合越来越趋向于精细。然而过度精细的拟合容易导致模型的泛化能力变差，即当模型用于之前未曾训练过的数据时表现很差。为了观测这种情况，我们需要划分一部分数据与用于训练的数据隔开，这便是我们划分验证集的原因之一。

为了防止模型过拟合，我们设定在验证集准确率连续三次迭代不再上升时提前终止训练。

我们让学习率随着迭代次数递减：

$$\text{learn-rate} = \frac{\text{initial}}{1 + \frac{\text{decay-rate} \times \text{step}}{\text{decay-step}}}$$

其中：

- decay_rate 为衰减进行的频率：经过多次尝试调参，我们将衰减率定为 e^{-2}
- initial 为初始学习率：经过多次尝试调参，我们将初始学习率定为 e^{-5}

4.3 Word2Vec + LSTM

4.3.1 Word2Vec

在词编码的基础上，我们对文本进行 word2vec 处理。Word2Vec 模型将每个单词映射到一个唯一的固定大小向量，同时可用文档中所有单词的平均值将每个文档转换为向量；然后，此向量可用作预测、文档相似度计算等。

4.3.2 LSTM 模型结构

1. 第一层为 `Embedding` 层，我们使用 `word2vec` 方法将单词编码转换为词向量。这些词向量经过训练，对于意思相近的词，其向量夹角小。
2. 第二层使用双向的长短期记忆层。长短期记忆网络层是一种特殊的循环神经网络层，它能够减轻长序列训练过程中的梯度消失和梯度爆炸问题，适合此处词向量长度较长的情况。它遍历序列中的每个元素作为输入，按照时间顺序传递输出。由于我们使用双向结构，最终结果由输入的前向和后向传递共同决定，这使得最前端的输入不必通过漫长的处理步数才能影响到最终结果，有效的提高了训练在文本中的均匀度。
3. 第三层为全连接层，由于在多层神经网络中梯度容易在深层网络中变得极小，使得参数无法正常更新，所以我们使用 `RELU` 作为激活函数解决梯度消失问题。
4. 第四层为输出维度为 5 的输出层，为了得到多分类的概率值，使用 `softmax` 函数将输出值压缩至 0 - 1 的范围内。

表 5: 搭建的神经网络结构（总参数个数：775681）

神经网络层	神经元个数	参数个数
Embedding	64	64000
双向 LSTM	128	66048
全连接	64	8256

神经网络层	神经元个数	参数个数
输出层	1	65

4.3.3 模型训练

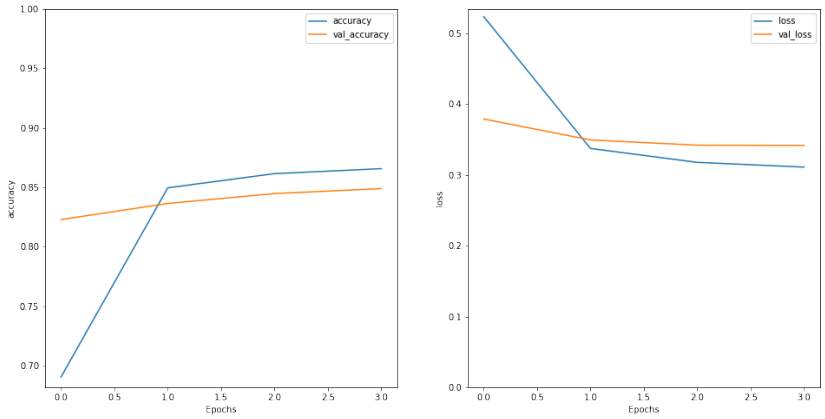

图 5: LSTM 模型训练过程损失函数和准确率趋势图

随着迭代次数的上升，模型在训练集和测试集上的损失函数下降，准确率上升。

4.4 BERT

4.4.1 BERT 介绍

BERT 是一系列双向文字编码转换模型的总称，用来结合上下文语义计算每个词的词向量，在自然语言处理中被广泛使用。

我们使用了前人在超大型语料库上训练的已有基础 BERT 模型，通过迁移学习的方式在我们的 BBC 文本数据集上进行微调。

4.4.2 预训练 BERT 模型

我们首先使用了一个参数量较少的 small-BERT 模型用于测试，在通过测试后，为了进一步提升模型的准确度，我们使用 al-BERT 和标准的 BERT 进行正式训练。

在 BERT 的输入层，对于原始的文字输入，我们需要将其转换成为数值编码。每一个 BERT 模型都有其严格对应的预处理模型来提升转换效果。

对于 small-BERT 模型预处理模型将输入的向量设为 128 的长度。

4.4.3 BERT 模型结构

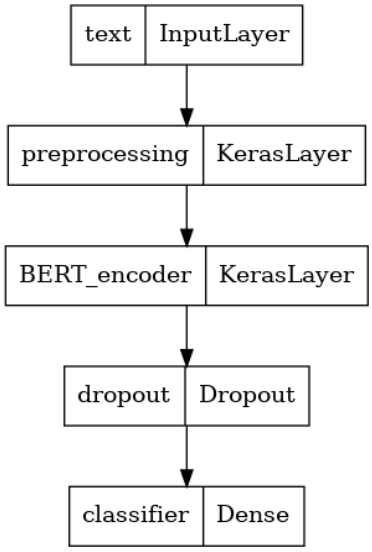

图 6: BERT 模型结构示意图

表 6: 搭建的神经网络结构（总参数个数：775681）

神经网络层	神经元个数	参数个数
Embedding	64	64000
双向 LSTM	128	66048
全连接	64	8256
输出层	1	65

4.4.4 模型训练

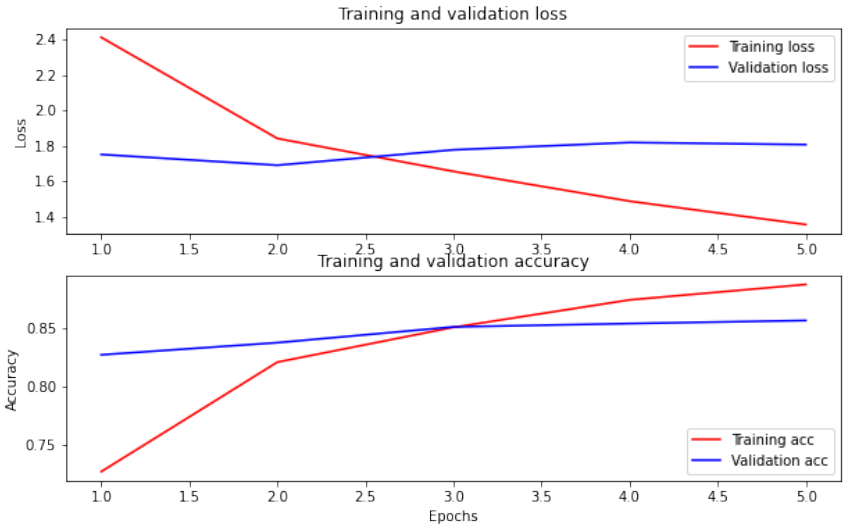

图 7: BERT 模型训练过程损失函数和准确率趋势图

随着迭代次数的上升，模型在训练集损失函数下降，但在验证集上损失函数基本保持平稳，准确率上升。

我们使用交叉熵作为我们的损失函数：

$$-\sum_{c=1}^{M} y_{o,c} \log(p_{o,c})$$

其中：

- M 是分类数
- y 是标签 c 在观测 o 下是否分类正确的 0/1 变量
- p 是预测概率

由于神经网络刚开始训练时非常不稳定，因此刚开始的学习率应当设置得很低很低，这样可以保证网络能够具有良好的收敛性。但是较低的学习率会使得训练过程变得非常缓慢，因此这里采用从较低学习率逐渐增大至较高学习率的方式实现网络训练前 10% 次迭代的“热身”阶段。一直使用较高学习率是不合适的，因为它会使得权重的梯度一直来回震荡，很难使训练的损失值达到全局最低谷。因此在 warm-up 结束后，我们使用线性减小的学习率。

在迁移学习时，我们选取的优化器与 BERT 在预训练时的 `Adamw` 优化器保持一致。

input : γ(lr), β_1, β_2(betas), θ_0(params), $f(\theta)$(objective), ϵ (epsilon)
λ(weight decay), $amsgrad$
initialize : $m_0 \leftarrow 0$ (first moment), $v_0 \leftarrow 0$ (second moment), $\widehat{v_0}^{max} \leftarrow 0$

for $t = 1$ **to** ... **do**
$g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$
$\theta_t \leftarrow \theta_{t-1} - \gamma\lambda\theta_{t-1}$
$m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t$
$v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$
$\widehat{m_t} \leftarrow m_t / \left(1 - \beta_1^t\right)$
$\widehat{v_t} \leftarrow v_t / \left(1 - \beta_2^t\right)$
if $amsgrad$
$\widehat{v_t}^{max} \leftarrow \max(\widehat{v_t}^{max}, \widehat{v_t})$
$\theta_t \leftarrow \theta_{t-1} - \gamma \widehat{m_t} / \left(\sqrt{\widehat{v_t}^{max}} + \epsilon\right)$
else
$\theta_t \leftarrow \theta_{t-1} - \gamma \widehat{m_t} / \left(\sqrt{\widehat{v_t}} + \epsilon\right)$

return θ_t

图 8: Adamw 算法示意图

Adam 的超收敛性质使其在训练学习率高的神经网络时可以达到节省迭代次数的效果。只要调整得当，Adam 在实践上都能达到 SGD+Momentum 的高准确率，而且速度更快。在几年前人们普遍认为 Adam 的泛化性能不如 SGD-Momentum，然而今年论文表明这通常是由于所选择的超参数不正确导致，通常来说 Adam 需要的正则化比 SGD 更多。

4.4.5 模型评价

我们在测试集上进行拟合，得到准确率为 78%。由于该模型仅仅为小型的 BERT，为了进一步提升模型的准确度，我们使用 al-BERT 和标准的 BERT 进行正式训练，分别达到了 82% 和 88% 的准确率。

4.4.6 模型应用

使用模型对输入的文本进行分类。我们输入一则测试新闻文本：“这部书很差劲”，该文本被模型分类为消极，符合预期。

5 分布式训练

我们使用 `pyspark` 进行分布式训练。分布式不同于单机训练，而是通过集群上许多的计算机节点同时进行训练。对于文本量很大的数据集而言，单机可能不具备足够的内存和 CPU 资源进行训练，借助于分布式系统，我们能调度集群计算资源进行计算。`pyspark` 是 `spark` 在 `python` 下的实现，它使用 Zookeeper、hadoop 作为底层，通过 MapReduce 的方式将大的计算任务拆解成为一个个小的任务，分发到每个计算机节点上进行计算。

5.1 环境启动

- 通过 YARN 资源调度系统提交到作业队列：`spark-submit --master yarn`
- 由于在 UDF（用户自定义）函数中使用了第三方包，需要将其发送至集群中的每个计算节点 `--py-files gensim.zip`
- 队列计算完成后将结果重定向输出 `> output.txt`

5.2 数据读取

由于数据为逗号分隔的 csv 格式，在文本列出现混淆。我们使用 pandas 进行读取后再转换为 spark DataFrame 格式

5.3 文本特征工程

词频-逆文档频率（TF-IDF）是一种广泛用于文本挖掘的特征向量化方法，它反映了单个词汇相对于语料库中文档的重要性。我们用表示 t 代表词汇，用 d 代表表示文档，用 D 表示语料库。词频 $TF(t,d)$ 是该词在文档 d 中出现的次数，而文档频率 $DF(t,D)$ 是包含该词的文档的数量。如果我们只使用词频来衡量重要性，很容易过分强调那些出现频率很高但几乎没有关于文档的信息的词，例如 “这”“的” 等词汇。如果一个术语在语料库中经常出现，则意味着它不包含有关特定文档的特殊信息。逆文档频率是一个术语提供多少信息的数值度量：

$$IDF(t,D) = \log \frac{|D|+1}{DF(t,D)+1}$$

其中 $|D|$ 是语料库中的文档总数。

由于使用对数，如果一个词出现在所有文档中，它的 IDF 值变为 0，因此使用平滑词以避免对语料库之外的词除以零。TF-IDF 是 TF 和 IDF 的乘积：

$$TFIDF(t,d,D) = TF(t,d) \cdot IDF(t,D)$$

在 TF 的基础上，我们使用改进版的 HashingTF 进行处理。HashingTF 将词汇转换为固定长度的特征向量。HashingTF 利用散列表应用哈希函数映射词汇到索引，之后通过映射的函数计算词频，能有效降低 TF 在大型语料库所需的时间。

我们从一组句子开始，将每个句子分成单词，构建词袋，使用 HashingTF 将句子散列成特征向量，使用 IDF 重新缩放特征向量，然后将我们的特征向量传递给学习算法。

5.4 训练模型

我们首先使用简单的 logistic 模型进行拟合，在训练集上进行拟合，之后在测试集上验证模型的效果。

表 7: 真是值与预测值示例

真实值	预测值
0.0	0.0
0.0	0.0
1.0	1.0
1.0	1.0
0.0	0.0
1.0	1.0
0.0	1.0
0.0	0.0

我们准备了两个测试用例来验证模型是否有效。

1. 我喜欢这部书
2. 它很差劲

模型对前一个句子的分类结果为积极，对一个句子的分类结果为消极。

5.5 模型比较

在 logistic 模型的基础上，我们还搭建了决策树模型、梯度助推树模型、随机森林模型。

决策树的特点是它总是在沿着特征做切分。随着层层递进，这个划分会越来越细。

梯度助推树模型使用 Boosting 的方式把基础模型组合起来。既然决策树基础模型可以做出不完美的预测，那么用第二的基础模型，把“不完美的部分”补上，不断地对“不完美的部分”进行完善，就可以得到效果足够好的集成模型。Boosting 的策略非常多，以 GBDT 为例，它会用第 K 个 CART 拟合前 k-1 个 CART 留下的残差，从而不断的缩小整个模型的误差。

相比于决策树模型，随机森林其实是一种集成算法。它首先随机选取不同的特征 (feature) 和训练样本 (training sample)，生成大量的决策树，然后综合这些决策树的结果来进行最终的分类。所以理论上随机森林相比单一的决策树模型一般来说准确性上有很大的提升，同时一定程度上改善了决策树容易被攻击的特点。

5.6 模型调参

我们使用网格搜索的方式对几个模型的超参数进行调整，选取最优的模型。

6 结论

在本研究中，我们通过网络爬虫的方式获取《哈利波特》全系列图书的评论进行文本分析，在文本预处理阶段，我们首先进行文本清洗，之后对中文文本进行分词，再进行文本特征工程。

6.1 主题模型

使用主题模型，我们对评论进行主题提取，提取出三个最主要的主题：

1. 第一个话题主要是对大结局的评论。图书馆保卫战、第七部、虎头蛇尾都是对结局的讨论。
2. 第二个话题主是对书中各类较为主要的人物进行讨论。排名第一的词汇斯内普是书中以为颇具争议的反派人物，其性格方面的多重性给了读者非常大的讨论空间，弗雷德、小天狼星也都是书中较为典型的人物，他们与哈利波特具有密切地联系，
3. 第三个话题主要围绕最主要人物哈利波特和成长历程中的重要节点剧情进行讨论。英雄主义、火焰杯、打游击都体现了主角的主线剧情。

6.2 评价分类训练

我们使用用户的评论评价进行文本情感分类模型的训练。在文本预处理阶段，我们尝试使用词编码和词向量的方式，在训练阶段，我们构建了 DNN、LSTM、BERT 等多个深度学习模型进行训练，并进行了模型比较，最终达到了 88% 的准确率。最后，为了进一步实现在超大文本集上进行训练，我们使用基于 Spark 的分布式算法在集群服务器上进行训练测试。[2]

模型	计算配置	用时	准确率	可拓展性
tokenize + DNN	阿里云服务器 Xeon 8 核 CPU 32G 内存	3 分钟	55%	低-单机
Word2Vec + LSTM	阿里云服务器 Xeon 8 核 CPU 32G 内存	1 小时	70%	低-单机
bert - 小型	阿里云服务器 Xeon 8 核 CPU 32G 内存	24 分钟	78%	低-单机
bert - AL	阿里云服务器 Xeon 8 核 CPU 32G 内存	1.6 小时	82%	低-单机
bert - 标准	阿里云服务器 Xeon 8 核 CPU 32G 内存	1 小时	88%	低-单机
spark - logit	中央财经大学大数据高性能分布式集群	4 分钟	73%	高-集群
spark - 决策树	中央财经大学大数据高性能分布式集群	10 分钟	80%	高-集群
spark - 梯度助推树	中央财经大学大数据高性能分布式集群	15 分钟	82%	高-集群
spark - 随机森林	中央财经大学大数据高性能分布式集群	30 分钟	87%	高-集群

[2]分布式模型在该小型数据集上没有优势，进行此项的意义在于对大型文本数据集可拓展性的技术储备，仅有在文本量级超过单机可承载上限时，分布式计算才具备意义

7 参考文献

[1] 张征杰, 王自强. 文本分类及算法综述 [J]. 电脑知识与技术,2012,8(04):825-828+841.

[2] 汪岿, 刘柏嵩. 文本分类研究综述 [J]. 数据通信,2019(03):37-47.

[3] 贾澎涛, 孙炜. 基于深度学习的文本分类综述 [J]. 计算机与现代化,2021(07):29-37.

[4] 王博, 刘盛博, 丁堃, 刘则渊. 基于 LDA 主题模型的专利内容分析方法 [J]. 科研管理,2015,36(03):111-117.DOI:10.19571/j.cnki.1000-2995.2015.03.014.

[5] 关鹏, 王曰芬, 傅柱. 不同语料下基于 LDA 主题模型的科学文献主题抽取效果分析 [J]. 图书情报工作,2016,60(02):112-121.DOI:10.13266/j.issn.0252-3116.2016.02.018.

[6] 黄佳佳, 李鹏伟, 彭敏, 谢倩倩, 徐超. 基于深度学习的主题模型研究 [J]. 计算机学报,2020,43(05):827-855.

[7] 胡吉明, 陈果. 基于动态 LDA 主题模型的内容主题挖掘与演化 [J]. 图书情报工作,2014,58(02):138-142.DOI:10.13266/j.issn.0252-3116.2014.02.023.

[8] 陈晓美, 高铖, 关心惠. 网络舆情观点提取的 LDA 主题模型方法 [J]. 图书情报工作,2015,59(21):21-26.DOI:10.13266/j.issn.0252-3116.2015.21.003.

8 爬取文本展示

---------- 好评 ----------

“阿不思•西弗勒斯，你的名字中含有霍格沃茨两位校长的名字。其中一个就是斯莱特林的，而他可能是我见过的最勇敢的人。
哈利波特原本就是一套适合从11，12岁读到20出头的系列，我们有幸赶上这个时代，和小主人公们一起沉淀，一起成长；我们
这本书是我高三的时候在Amazon买的预售，和一枚小小的金色飞贼一起到来，用了3天读完，就再也没有勇气看第二遍，当时
邓布利多死得值了，结尾够曲折。邓布利多与格林德沃……一个世纪的爱与恨，这么有料的CP，发展个番外吧。
休憩33rd.当初刚出七，我买来英文版艰辛啃完，泪流满面，今天我第一次读中文版，于是在魔都旅馆床上，放声大哭
所有爱唱反调的胖友们，你们都没试过给疯姑娘卢娜的《唱唱反调》投稿吗?
基本上一如预想。罗琳笔下的主要角色——除了铁三角是标准成长故事的逐步成长外——其他的都经历了一个从认识表象到探
//阿布思·西弗勒斯，你的名字中含有霍格沃茨两位校长的名字，其中一位就是斯莱特林的，而他可能是我见过的最勇敢的人
走吧走吧 人总要学着自己长大
最感动的情节是多比的死亡和哈利的面对死亡的表现.
无论何时何地，爱与勇气都不可忘却。
多年以后才想起来看结局，马上就沦陷了。和初中时候一样，一开始看就满眼幻觉，四点爬起来看书这种事情只有在这时候才
all as well，一套看了近十年的书，JK罗琳用十年的时间让我们忘记了自己永远只是一个麻瓜的事实。
每翻一页便是与你的告别。我不想同你告别。那瑰丽堂皇的永不言弃的世界。我不想同你们分开。| 消失的东西去了哪里。化
几年后重新看了最后的几章，在这热得要死的梅雨终焉，在东瀛的小小房子中依旧看得热泪盈眶，哈利是真正的领袖，真正的
再也沒有哈利波特了.多讓人傷心.
我觉得2018年1月，我打开豆瓣读书的年终总结页面，可能会发现17年我一共读过七本书，哈利波特1-7。。
哈利还是那个哈利，纳威已经不是那个纳威了。
赶在去大阪的前一天重温完了这套书。为了把大布局收尾，罗琳硬生生编出两个“N大”设定，这是让线索合理化的最简单技
我用六个小时读完了《哈利波特与死圣》。中文版的。中间只站起来喝了一口水。合上书时人都要虚脱了。
我所看过的最杰出的一部通俗文学。
卢平，教授，一路走好
一个时代的完结
2020年重读08:故事结束了，难过的时候我会拿出来读读，因为还相信某些就像书里所说早已被耻笑的东西，是因为这些才想
想写的太多来不及交代完的感觉，还有还我的双胞胎!
终于看完了，斯内普是最大的剧情了，总体来说，越来越烦哈利波特本人。。
呜呜呜哭了好几次。
在哈利三人组开始毫无头绪的逃亡之后，作者又试图不断用圣器魂器这类名词混淆读者的思路，但最终，还是以哈利在决斗中

---------- 中评 ----------
别的还好，双胞胎死一个活一个实在太不能忍了。
这本稀碎情节太多，完全都是没必要的。比如哈利怀疑邓布利多的过去、罗恩的出走、哈利让罗恩摧毁魂器、赫敏再见到罗恩
反正总是要看完的嘛!
罗琳忽悠了全球读者
惯性让人读完。

早就读过了啊……
情节峰回路转，让人摸不着头脑，总体感觉罗琳的逻辑开始秀逗了
比前六部烂多了
无声地夸大了一些东西的影响。
2008-4-14 终于下决心去完结这个系列，其实已经没多大兴趣了，意义多点
2008-5-8 第十二章。看不下去了，等电影吧
十年光阴
糟糕无比的翻译，真可谓每况愈下
比混血王子让我满意多了 赫赫 还是值得读的
这本书，它可以更好；但作者是有心无力还是心血都舍不得投了呢；除了斯内普教授谜底揭晓，没有亮点，结构和逻辑水准..
唯独这个结尾觉得很莫名……
为电影而创作...
一切回归平静。
我对它有执念，严重的执念。
任何结局对于哈利fan来说都不可能完美……
有点别扭的结局……
混血王子以后一直都没敢看最后这一部。现在总算看完了。唉。
不忍心去过多批评什么，从第一部到第七部，从初一到大一……
二十九章开始的败笔
拖沓
算是个不错的结局……
怎么没有前6部好呢？ 2019.11.22维持原判哈哈~
哈利没死真的是大败笔
最讨厌的一部
出于对哈的感情才打的3颗，不然。。。
这是七本里我认为最狗血的一本……但是，身体里的一部分也就跟着这个系列的终结而死掉了
一般，看起来没有前几本那么引人入胜，难道是我的年纪大了？
很垃圾的最后一本,不喜欢!
花了两年大学的假期断断续续才把此书读完，这本大结局让我想起了那些苦背雅思范文的学生，复出巨大的精力与时间，就是
斯内普
越来越差了。
有些地方拖沓。。。
算是7本里最差的了。
结局接受不了。
减一星，为了小天狼星....
回不去の心境
不喜欢结尾。
竟然是反转剧，我晕
有一点失望。

英国起点小说

---------- 差评 ----------

对不起，我实在是无法接受这个结局。
从哈一到哈七。。。水平依次下降。。越写越不如前，作家都免不了这样的杯具么?
这套书根本就是在宣扬扭曲的价值观吧
翻译太烂。和上一本一样，还比不上网译版本。
糟透了
什么破结局!
2星指翻译！翻译得还不如网上的
我很后悔买了这书
书越来越厚，却没有第一次拿起的心情，后面的都不怎么好看了，最后一步挣扎了很久还是决定有始有终
情节不连贯，罗琳依旧采用在最后几章揭谜底的方法来为小说自圆其说，但是无法掩饰哈利在情节上的苍白。哈利终究只是一
吐都吐不出……
伤心除了斯内普把亲爱的校长大人说成这样
我希望哈五六七是三部同人，真诚地。
讨厌这种结尾...那么多人凭什么不明不白死了...
毕业了
为毛让教授当个情圣还悲惨的挂掉。tnnd
罗琳在还是无法胜任最后一部的写作，是她水平还不到还是没有认真写?
少兒讀物
这个和之前的完全没法比，感觉越来越有糊弄小孩的趋势，不过可能是因为我年龄的问题。看哈1才上小学四年级，看哈7已经
狗血剧情。。。
斯内普死得时候很感动。
系列里，最差的一本，能看出来罗琳对波特烦透了
说实在的，没看懂 -_-b
咬牙看完的，为的是看电影接的上。毫无快感，罗琳大妈般的磨磨唧唧，故作煽情的描写，真是终于结束了。
莫名其妙的哈金恋 恨
狗血
KJ让我失望了
仓促的结局，令人无法满意呢
有始有终，撑着看完滴
我是有偏见的，对于匆忙的结尾，特别是卢平他们的死
骗我钱。。。虽然心甘情愿。。。。
才发现八年前的我还是挺直白，没有因为感伤和怀念乱打高分。
还好啦~尽管结局都是意料之中的！~
明显黔驴技穷
傻逼翻译