

# 单词、短语和语句——如何读懂你的情绪？(Words, Phrases & Sequence—How to read your emotions?)

Project of *Introduction to Statistical Learning*

于骏浩 周子逸 李培森

## Abstract

本文针对Stanford Sentiment Treebank 数据集进行情感分类，逐步引入单词、词组和语序信息，使用以 PCA 为代表的降维方法，以 LDA 为代表的机器学习方法和以循环神经网络为代表的深度学习方法分析其对情感分类的重要性。在此过程中通过分析机器学习、深度学习模型的优劣，不断针对该数据集优化模型。最后综合单词、词组和语序信息，结合不同模型的优势，设计出一个时间复杂度较低、可解释性较强的模型，达到效率和准确率的平衡。

## Contents

|                        |           |
|------------------------|-----------|
| <b>1 研究背景与目的</b>       | <b>2</b>  |
| 1.1 研究背景               | 2         |
| 1.2 研究目的               | 2         |
| <b>2 数据收集与清洗</b>       | <b>2</b>  |
| <b>3 文本数据探索性分析</b>     | <b>3</b>  |
| 3.1 数据平衡性分析            | 3         |
| 3.2 特征工程               | 3         |
| 3.3 朴素贝叶斯与随机森林预测       | 4         |
| 3.4 词云图                | 5         |
| 3.5 N-Gram             | 6         |
| <b>4 研究内容与方法</b>       | <b>6</b>  |
| 4.1 机器学习               | 6         |
| 4.2 深度学习               | 7         |
| <b>5 研究结果</b>          | <b>8</b>  |
| 5.1 机器学习模型             | 8         |
| 5.1.1 降维处理             | 8         |
| 5.1.2 模型表现             | 10        |
| 5.1.3 讨论               | 15        |
| 5.2 深度学习模型             | 15        |
| 5.2.1 Comparison Table | 15        |
| 5.2.2 结果分析             | 15        |
| 5.2.3 讨论               | 19        |
| <b>6 总结与展望</b>         | <b>20</b> |
| 6.1 情感标注的主观性           | 20        |
| 6.2 特征工程：词与词之间的关系      | 20        |
| 6.3 恢复数据平衡性            | 21        |
| <b>7 附录</b>            | <b>21</b> |
| 7.1 小组成员分工             | 21        |
| 7.2 代码与数据集             | 21        |

# 1 研究背景与目的

## 1.1 研究背景

随着社交媒体、在线评论和用户生成内容的迅猛发展，人们越来越倾向于在公共平台上表达他们的观点、感受和情感。这种大规模生成的文本数据为我们提供了宝贵的信息资源，其中蕴含了丰富的情感和情绪表达。因此，情感分类成为了自然语言处理领域中的一个重要研究方向。

情感分类的重要性和必要性：

1. 挖掘用户情感和情绪：情感分类可以帮助我们深入了解用户在特定话题或产品上的情感倾向。通过准确分析用户的情感表达，我们可以了解他们的态度、偏好和需求，为市场营销、产品改进和用户体验提供有力的指导。

2. 品牌和舆情管理：情感分类能够帮助企业或品牌迅速了解公众对其产品、服务或活动的情感反馈。通过及时捕捉和分析用户的情感倾向，企业可以及早发现和应对负面舆情，保护品牌形象，并采取措施增强用户满意度和忠诚度。

3. 社交媒体分析：社交媒体平台上的大量用户生成内容包含了丰富的情感信息，情感分类能够帮助我们理解和分析社交媒体用户群体的情感趋势和意见动态。这对于舆情监测、事件跟踪、舆论引导和公共决策具有重要意义。

4. 智能客服和情感识别：情感分类技术在智能客服和情感识别方面具有广泛应用。通过对用户的文本输入进行情感分类，可以实现情感敏感的智能客服系统，提供更加个性化和精准的服务。此外，情感分类还可以应用于情感识别领域，如情感辅助疗法、心理健康监测等。

综上所述，情感分类在社会、商业和个人领域具有广泛的应用前景。通过准确识别和理解文本中的情感倾向，我们能够从大规模的文本数据中获取有价值的信息，为决策制定、用户体验和情感智能化提供支持。因此，情感分类的研究和应用具有重要性和必要性。

## 1.2 研究目的

基于前述原因，情感分类的应用场景极为广泛，为众多商业行为提供了关键的基础技术分析。对于企业和组织而言，准确地了解和解释用户的情感倾向对于产品改进、市场营销和品牌管理至关重要。因此，我们的研究小组选择了情感分类作为研究主题，并致力于解决一个关键问题：在情感分类中，情感词汇、短语还是句子序列化信息，哪一个提供了最为关键的特征？

情感词汇是情感分类的重要组成部分。这些词汇通过传达情感色彩和情绪表达来影响整个文本的情感倾向。然而，我们需要进一步探讨情感词汇在情感分类中的准确度和有效性，并考虑如何赋予它们更高的权重以更好地捕捉情感信息。

此外，短语和句子序列化信息也可能在情感分类中起到关键作用。短语可以提供更多上下文信息，帮助我们理解情感表达的更深层含义。句子序列化信息则允许我们考虑文本中的语义关系和逻辑结构。因此，我们需要进一步研究和探讨如何有效地提取和利用短语和句子序列化信息，以提高情感分类的准确性和稳健性。

在这一研究中，我们的目标是找到情感分类中最为关键的特征，并通过充分利用和结合这些特征来达到最佳的预测效果。我们将使用具有五分类的 Stanford Sentiment Treebank 数据集作为研究对象，并运用机器学习和自然语言处理技术来构建和评估情感分类模型。

通过深入研究情感分类中的特征选择和特征组合问题，我们期望为情感分类提供更准确和可靠的解决方案。这将有助于提升企业决策制定、市场营销和舆情分析的能力，从而更好地满足用户需求、改善用户体验，并取得商业上的成功。

# 2 数据收集与清洗

我们选择的数据集是 Stanford Sentiment Treebank，该数据集基于电影评论，以单句作为研究对象，并具有五个级别的分类，从非常负面的评论到非常正面的评论。

在数据清洗的过程中，我们首先利用预训练的情感词典 Afinn 提取了句子中的情感词，并根据其得分赋予情感词更高的权重。此外，我们还利用词性标注从句子中提取形容词、动词、名词和副词。形容词（如“brilliant”）、动词（如“love”）和名词（如“fantasy”）在情感倾向分类中具有明显的作用，而副词（如“highly”、“very”）则为我们判断评论是否极端提供了依据。然而，像“and”、“that”等词性的词不具有实际意义，对分类没有帮助，因此被去除。在进一步的词频图分析中，我们意识到“be”动词和助动词虽然属于动词，但仍然没有实际意义，因此也被排除。类似“‘ll”、“’ m”等缩写形式也被移除，因为它们不具有实际意义。词频图还显示出高频词如“movie”、“film”、“story”等

与电影相关的词汇，但这些词对情感分析没有帮助，因此也被去除。在数据清洗的过程中，我们并没有进行词形还原操作。这是因为像“most”、“best”等比较级词汇实际上对于我们判断评论是否极端是有帮助的，所以没有进行词形还原处理。

### 3 文本数据探索性分析

#### 3.1 数据平衡性分析

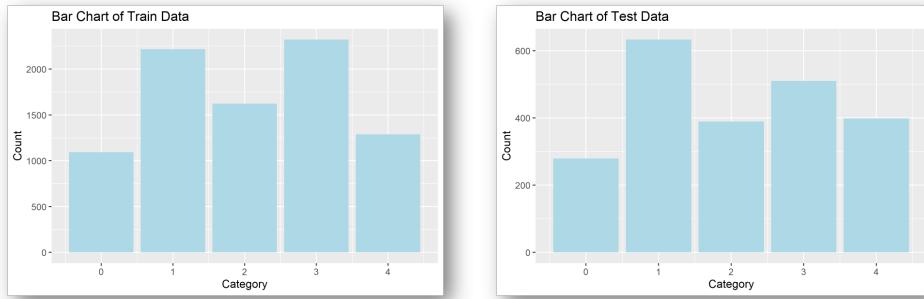


Figure 1: Bar chart of the Data Sets

Category 0 至 4 分别表示情感分类为 very negative, negative, neutral, positive 和 very positive。通过 Prevalence 分析，我们可以观察到稍显 negative 和 positive 的评论比例较高且相近，而中性和极端的评论比例较低。这符合我们对电影评论的一般认知，即极端差评和极端好评的电影占比较低。

这一分析结果表明，在给定的数据集中，大多数评论倾向于稍显负面（category 1）或稍显正面（category 3）的情感倾向，而中性和极端情感的评论较为罕见。这可能反映了人们对电影的普遍态度，即对于绝大多数电影，人们的评论更倾向于在稍显负面或稍显正面之间。

这样的分布情况有助于我们更好地理解情感分类的挑战，以及在建立模型时需要注意的情感倾向的平衡性。

#### 3.2 特征工程

在探索性数据分析（EDA）方面，我们首先尝试了特征工程，对原始句子的单词平均长度、句子长度以及筛选后的单词平均长度和句子长度，以及词汇密度等特征进行了分析。然而，通过箱线图的观察，我们发现不同类别之间的区别并不明显。因此，我们没有尝试手动加入更多的特征，而是将注意力集中在单词本身提供的特征上。

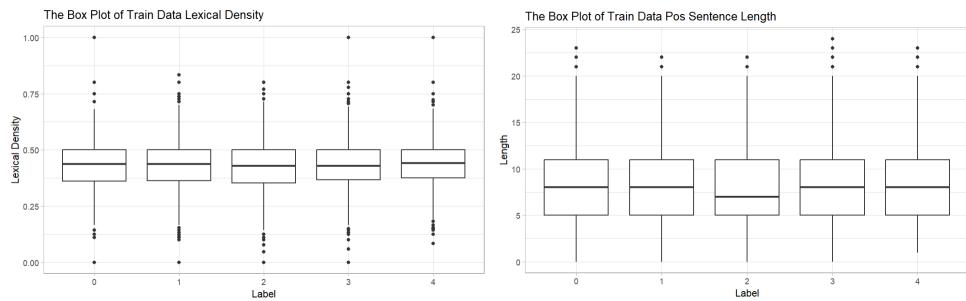


Figure 2: The Box Plot of Train Data Lexical Density & Pos Sentence Length

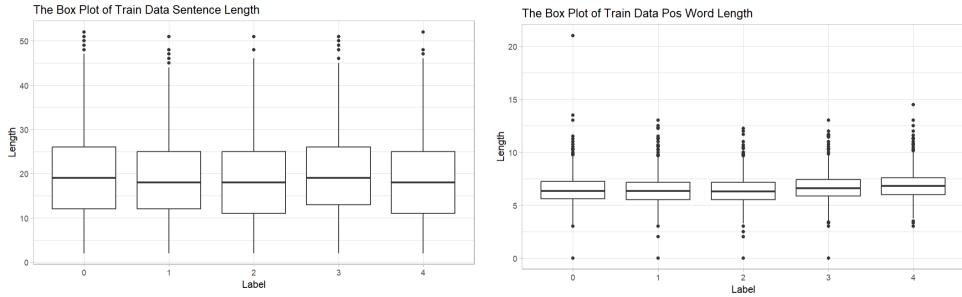


Figure 3: The Box Plot of Train Data Sentence Length & Pos Word Length

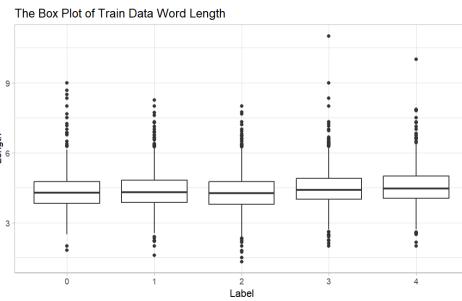


Figure 4: The Box Plot of Train Data Word Length

### 3.3 朴素贝叶斯与随机森林预测

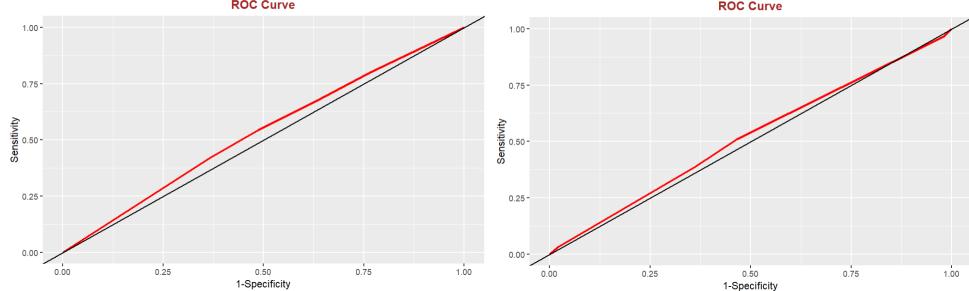


Figure 5: The ROC curve of Naive Bayes and Random Forest before data cleaning

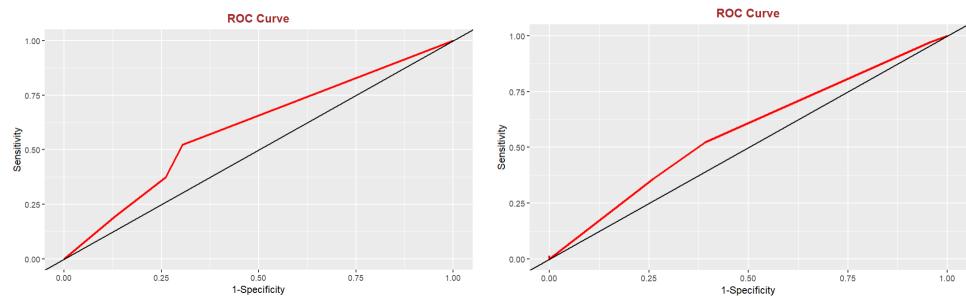


Figure 6: The ROC curve of Naive Bayes and Random Forest after data cleaning

我们首先使用了朴素贝叶斯算法和随机森林算法来验证我们进行数据清洗后的数据的有效性。通过观察结果，我们可以看到在使用清洗后的数据进行训练时，无论是 ROC 曲线还是准确率都有一定程度的提升。然而，这两个模型的表现都只是一般，并与随机猜测 (random guess) 的效果相差

不大。现在我们来从理论上进行分析，这两个模型的底层都基于词袋模型的假设。词袋模型假设词汇的顺序不重要，并且词汇是独立分布的。然而，在这个问题中，这两个假设显然不成立，让我举例说明：

假设有两个句子：

1. It's never dull and always looks good.(++)
  2. This one's weaker than most.(-)

根据词袋模型的假设，我们将这两个句子拆分为单词，并忽略词序和上下文，得到的词汇表为：[never, dull, always, look, one, good, weaker, than, most]。

然而，通过观察这两个句子，我们可以发现词汇之间的关系对于情感分类是非常重要的。比如在句子1中，“good”表达了正面的情感倾向，“dull”表达了负面的情感倾向，而“never dull”则表达了正面的情感倾向；而在句子2中，“most”可能传达了正面的情感倾向，“weaker than most”则传达了负面面对情感倾向。可见词汇的排列顺序和上下文信息在情感分类中具有重要影响。

因此，仅仅基于词袋模型的假设，这两个模型在该问题中表现一般，并且与随机猜测的效果相近。我们需要进一步改进模型，考虑词汇的顺序和上下文信息，以更好地进行情感分类。

### 3.4 词云图



Figure 7: Word Cloud Figures of 'Very Negative' and 'Negative'



Figure 8: Word Cloud Figures of 'Very Positive' and 'Positive'



Figure 9: Word Cloud Figure of 'Neutral'

我们通过绘制筛选后的词云图，直观展示了模型失败的原因。从词云图中可以观察到，无论评论是否极端，诸如”most”、”more”等表示程度的词汇都会频繁出现。而像”good”、”little”、”best”这类具有明显情感倾向的词汇在中性评论中也占据了相当比重。这说明仅仅依赖单个单词的特征进行分类的方法存在缺陷。

让我举例说明这一点：

- 1.The best thing that can be said of the picture is that it does have a few cute moments. (0)
- 2.The best drug addiction movies are usually depressing but rewarding.(0)

根据词袋模型的假设，我们将这两个句子拆分为单词，并忽略词序和上下文，得到的词汇列表如下：句子 1：[best, thing, said, picture, few, cute, moments] 句子 2：[best, drug, addiction, usually, depressing, but, rewarding]

如果只考虑单个单词，那么在句子 1 中，”best”、”cute”等词语传达了正面的情感倾向，但通过整体语句判断该评论传递的情感表达是中性的。然而，在句子 2 中，出现”best”、”rewarding”传达正面感情的词，”but”、”depressing”等传递负面感情词语也出现了，但是整体情感倾向是中性的。这说明单纯依赖单个单词的特征进行分类是不够准确的，因为同一个单词可以在不同上下文中传递不同的情感。

因此，我们需要更细致地考虑词汇的上下文信息，以及单词之间的关系，来改善情感分类模型的性能。

### 3.5 N-Gram

解决上述问题的关键是引入上下文信息。为了提供词组特征，我们首先引入了 n-gram 特征。为了避免引入噪声，我们在清洗后的数据上进行了 n-gram 分割。通过对比实验，我们最终选择了 2-gram 作为我们的特征。以下是我们绘制的词频图，可以观察到其中许多单个词所不具备的表义特征。

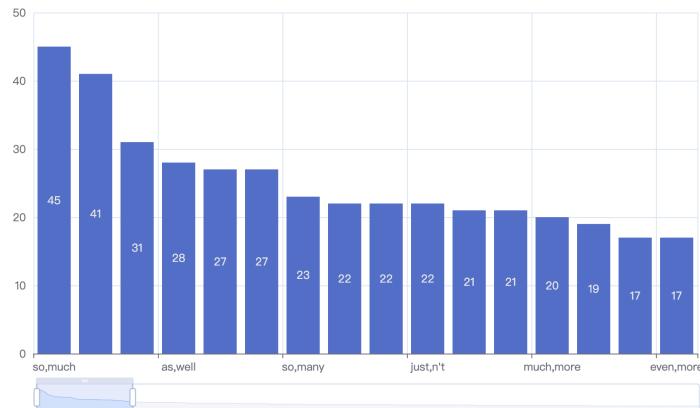


Figure 10: Bar Chart of 2-gram phrases

例如，”too much” 将原本倾向于正向情感的”much” 转变为了贬义，”not only” 赋予了”only”一个较强的递进含义，而去除了情感倾向。这些观察结果为我们提出最终模型提供了关键的思路。

通过引入 2-gram 特征，我们能够更准确地捕捉词语之间的关联和上下文信息，从而改善情感分类的性能。这一策略有助于更好地识别词组的整体情感倾向，并提供了更丰富的特征信息，进一步增强了模型的表现。

## 4 研究内容与方法

### 4.1 机器学习

首先使用一些基础的机器学习模型进行研究，我们期望能够使用的模型包括 LDA,QDA,KNN 以及逻辑回归等等，然而想要使用这些模型，首先需要将句子转化为可以表示其特征的向量。

我们采用的方法是词嵌入，利用 word2vec 预训练模型，将句子中的词语转化为在语境空间中的 300 维向量，再简单地取其平均值作为整个句子在语境空间中的向量，没有考虑词序信息。

接着，为了处理高维的输入，我们首先尝试用 PCA 对数据进行降维，并探讨选取的主成分个数对模型表现的影响。

之后，由于降维之后维度依然非常高，KNN 由于维度灾难效果不好，因此我们分别尝试了 LDA, QDA 和逻辑回归模型。

这部分研究内容与朴素贝叶斯法的主要区别在于放弃了词语之间独立的假设；仍然存在的缺点则是没有考虑语序信息，然而语序信息在情感分类中恰恰是非常重要的，因此接下来使用深度学习模型研究这一课题。

## 4.2 深度学习

在进行上述的机器学习分析过程中，我们意识到序列化信息在情感分类中是必不可少的。因此我们选用了若干深度学习模型来纳入这一信息。

我们使用了若干循环神经网络来引入序列化信息，以下是我们的训练架构：

|                               |   |
|-------------------------------|---|
| <b>Word Vector</b>            | pre-trained 300-dimensional word vector |
| <b>Hidden Layer Dimension</b> | 256                                     |
| <b>Batch Size</b>             | 64                                      |
| <b>Max Epoch</b>              | 20                                      |
| <b>Optimizer</b>              | Adam                                    |
| <b>Learning Rate</b>          | 0.0001                                  |
| <b>Loss Function</b>          | Cross Entropy                           |
| <b>Activation Function</b>    | ReLU                                    |

由于该数据集具备若干特征，因此我们在训练架构上的选取也根据这些特征做出了针对性的优化：

- 预训练词向量

1. **丰富的语义信息。**情感分类任务的一个关键步骤就是捕捉词汇间的关联性和语义相似性。由于预训练词向量是通过在大规模文本语料库上训练得到，能够利用向量之间的距离关系表征语义相似性，因此能够在该任务中提供更多的语义上下文信息。
2. **降低维度。**在前述 EDA 过程中，我们发现本数据集词汇量巨大，若采用 One-Hot 等编码方式，维度过高，会导致维度灾难，甚至导致神经网络无法收敛。但注意到类似 One-Hot 的编码方式，事实上是一个高维的稀疏矩阵，而使用预训练词向量可以将高维度的稀疏表示转化为低维度的密集表示，使得信息的表征方式更为高效，也极大缓解了维度灾难问题。
3. **迁移学习。**预训练词向量已经捕捉了通用的语义信息，使用其来初始化模型参数，可以加速神经网络的收敛。同时，在前述机器学习分析过程中，我们意识到该数据集具有很强的过拟合倾向，而由于预训练词向量捕捉的语义信息不是局限于训练集上的，因此可以保证模型具有更好的泛化能力。

- Adam 优化器

1. **规避局部最优点。**本数据集以 phrase 尺度来计数，有 215,154 条语料，因此存在大量的局部最优点，而 Adam 优化器由于结合了动量的概念，利用了过去的梯度来更新参数，有助于训练过程中跳出局部最优点。
2. **加速收敛。**在进行机器学习分析的过程中，我们意识到不同单词在预测情感时的权重是不一样的，因此不同单词对应的参数的梯度变化有很大不同，而 Adam 优化器允许每个参数有自己的自适应学习率，因此更有助于模型收敛。

其次以 RNN 为例，给出我们的深度神经网络模型架构（虚线框的部分代表部分模型有，细节见后续结果中的比较表格；全连接层的维度与模型的选择有关）：

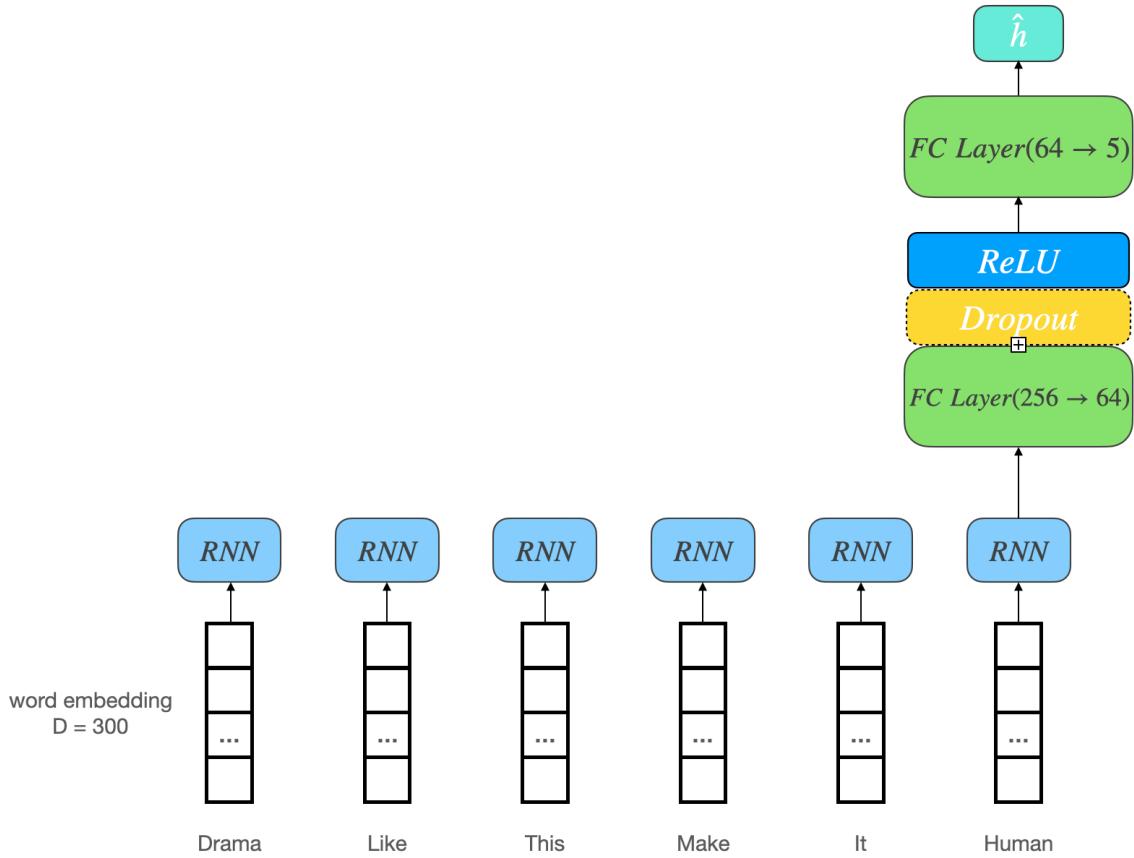


Figure 11: 模型架构（以单向 RNN 为例）

在模型架构上，除去 RNN 等循环神经网络，我们还在输出端添加了两个全连接层，该选择有如下原因：

1. **避免过大的维数差。**本训练集采用的词向量维度为 300 维，为了更好捕捉词汇语义之间的关系，隐藏层维度为 256 层，如果直接 256 维数据转换为 5 维预测输出，由于维数相差过大，实际上会导致权重更新非常缓慢。
2. **结合 Dropout 层。**本训练集存在很强的过拟合问题，因此添加全连接层也有利于融合 dropout 层以减轻过拟合问题。

## 5 研究结果

### 5.1 机器学习模型

#### 5.1.1 降维处理

对于 300 维的数据，首先考虑降维处理。我们较为熟悉的方案是因子分析和主成分分析。前者对可解释性要求较高，然而句子在语境空间中的向量解释起来比较困难，因此我们选择主成分分析。

首先对没有经过标准化的数据进行主成分分析，可以画出如下的主成分解释方差占比的 Scree plot：

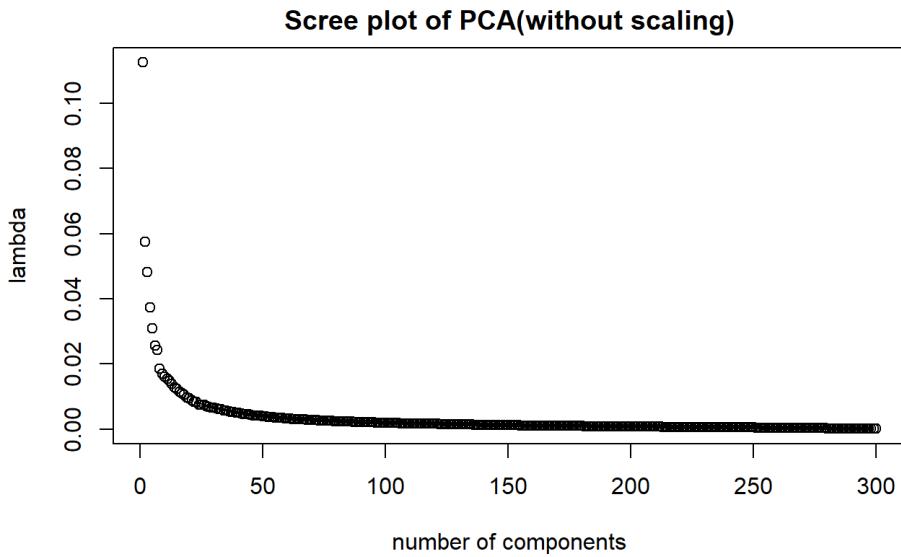


Figure 12: Scree Plot of the variance explained by each PC(without scaling)

然后再尝试对标准化(均值为0 标准差为1)后的数据进行主成分分析，同样画出 Scree Plot:

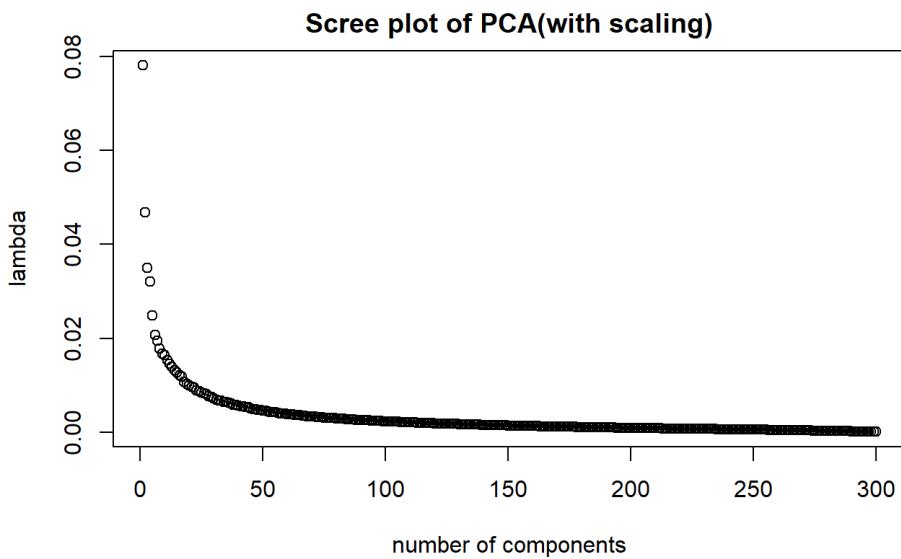


Figure 13: Scree Plot of the variance explained by each PC(with scaling)

可以发现，这两种方式得到的第一主成分解释的方差占比都约为 10%。之后解释方差占比迅速下降，但下降到 2% 之后放缓，因此很难通过肉眼直接确定应当选取的主成分个数。因此，我们采用定量的标准，计算解释方差占比大于平均值(也即  $1/300$ )的主成分个数、累计解释 80%, 90%, 95% 所需的主成分个数，如下表所示：

| Criterion                     | Data without Scaling | Data with Scaling |
|-------------------------------|----------------------|-------------------|
| <b>Average Var. Explained</b> | 60                   | 71                |
| <b>80% Explained</b>          | 108                  | 104               |
| <b>90% Explained</b>          | 183                  | 161               |
| <b>100% Explained</b>         | 253                  | 207               |

根据上表，我们首先可以发现降维之后维度依然比较高；其次，对比未经标准化和经过标准化

的数据的表现，可以发现未经标准化的数据的前若干个主成分解释的方差的占比比未经标准化的更高（例如第一个主成分，两者的占比分别为 11% 左右和 8% 左右）但是当我们需要累计解释较高的方差占比时，经过标准化的数据表现稍好一些。我们首先保留 200 个主成分，之后再对输入数据主成分的维数进行讨论和尝试。

除了讨论主成分个数的选取之外，我们还需要讨论数据是否要进行标准化。上述讨论中，两者在降维中的表现区别不大，而且我们可以画出两者的前几个主成分的散点图，如下所示：

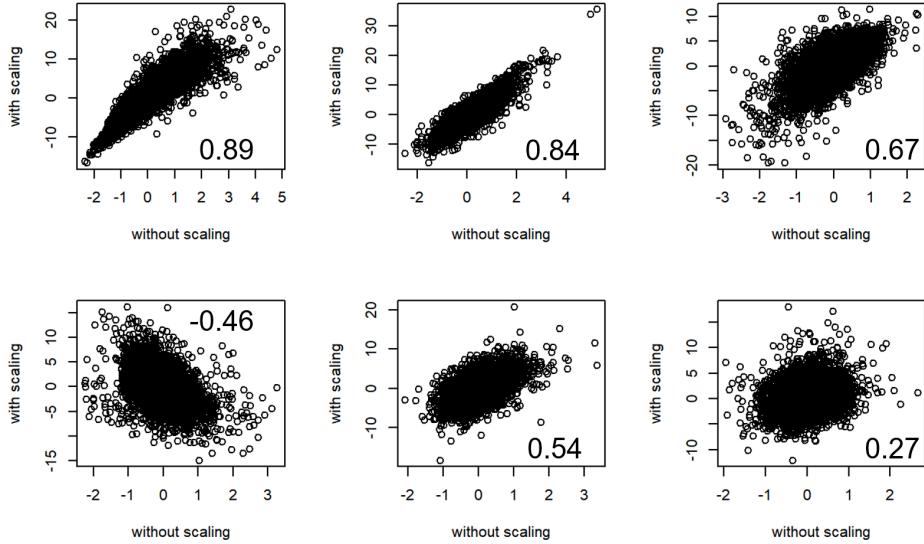


Figure 14: Scatter Plots of the first few principal components

不难发现两者的前几个主成分相关性很高，因此两种方法整体比较相似，以目前讨论的内容无法确定应该在两者中选择哪一者。我们的解决方法是在后续过程中两者都尝试一下，看看哪一个的表现更好。

### 5.1.2 模型表现

为了衡量模型的表现，我们选取的两个指标分别是 accuracy 和 F score，这里多分类 F score 采取的计算方式是 harmonic mean of macro average of precision and recall。Accuracy 计算简单，适用范围广；而对于不平衡的数据集，综合考虑 precision 和 recall 的 F score 可以提供一些更深层次的信息。

第一个模型为 Linear Discriminant Analysis，下图展现了选取一定个数的主成分之后模型在 training data 和 validation data 上的表现。

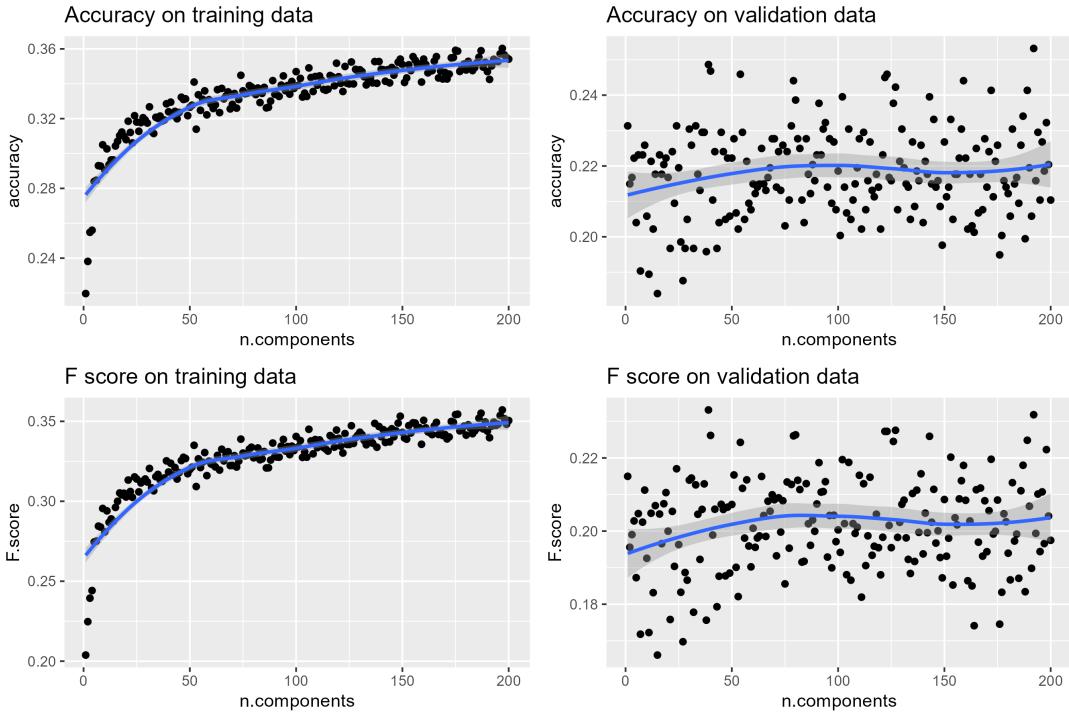


Figure 15: Accuracy and F score of LDA model (without scaling)

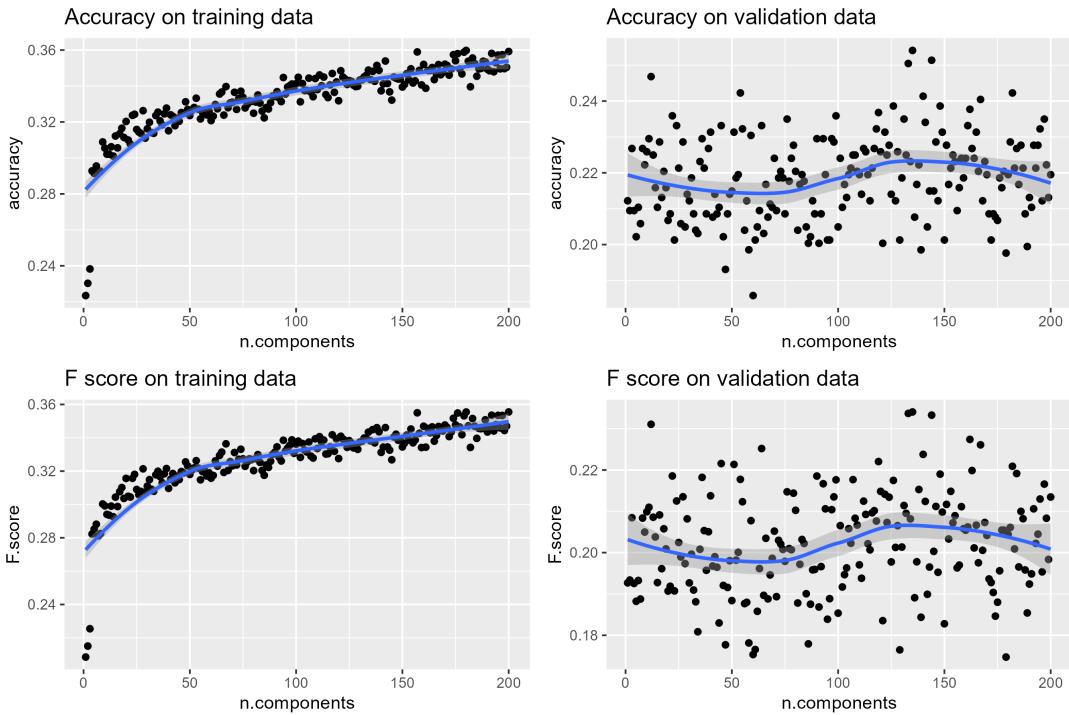


Figure 16: Accuracy and F score of LDA model (with scaling)

不难发现，LDA 在 training data 上的表现随着主成分个数的增加而上升（后续几个模型也类似，因此后面不会再对训练集上的表现进行讨论。然而，accuracy 和 F score 在 validation data 上的表现基本平稳，当主成分增加时，没有非常明显的上涨或者下跌，着说明 LDA 模型比较保守。对于未经标准化的数据，主成分个数由 1 增加至 100 的过程中，模型表现有非常轻微的上升趋势，在达到 100 之后则基本保持平稳；对于经过标准化的数据，其峰值大约出现在 130 个主成分处。因此，对于

未经标准化的数据选取 100 个主成分，对于经过标准化的数据选取 130 个主成分，最后在 test data 上面得出的 accuracy 分别为 21.6% 和 20.3%.

第二个模型为 Quadratic Discriminant Analysis，如下图所示；

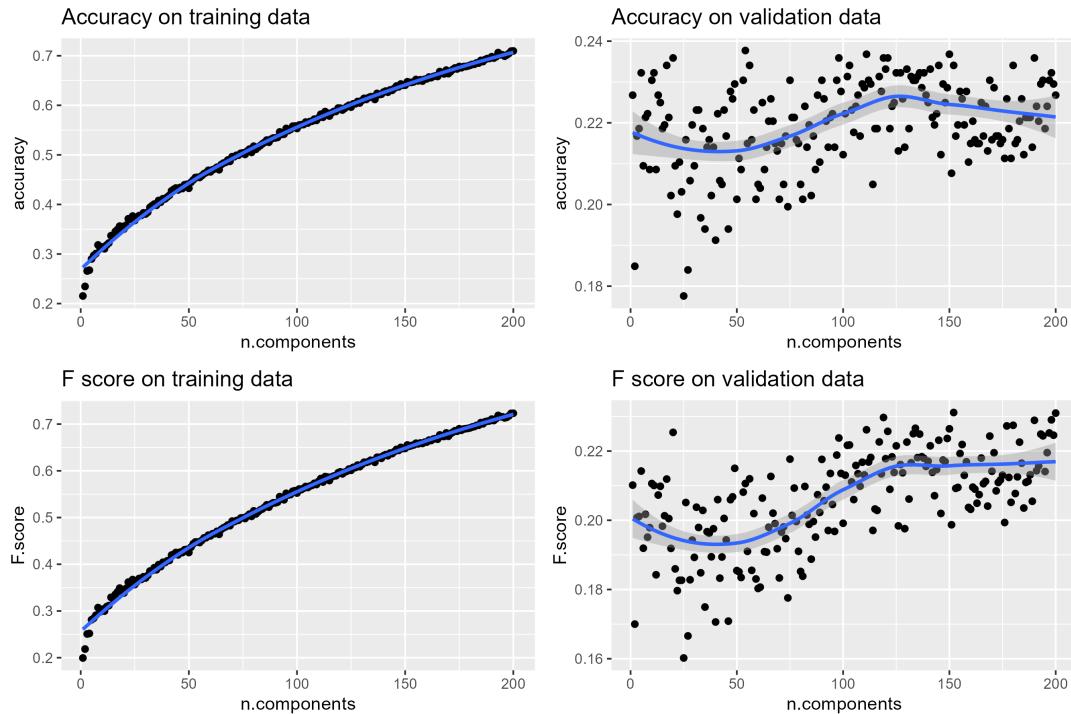


Figure 17: Accuracy and F score of QDA model (without scaling)

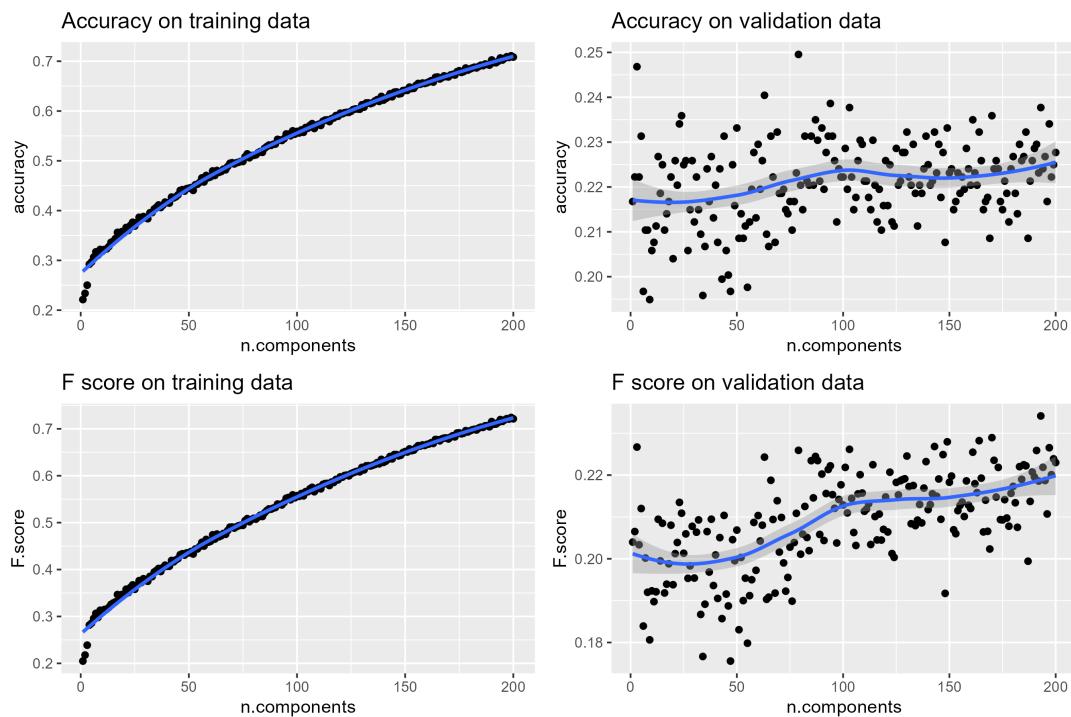


Figure 18: Accuracy and F score of QDA model (with scaling)

QDA 模型在 validation data 上面的表现与 LDA 模型有所不同。对于未经标准化的数据，当主成分个数从 1 增加到 125 时，QDA 在 validation data 上的表现的上升趋势比 LDA 明显得多；对于经过

标准化的数据，QDA 在 validation data 上的 accuracy 在 1 至 100 之间有上升趋势，而 F score 在 0 至 100 上上升得非常明显。

QDA 在主成分增加时表现的提升比 LDA 明显，其原因可以从模型特征和数据量两方面解释。一方面，QDA 模型比 LDA 更加灵活；另一方面，训练数据的样本较充分（8208 个样本），因此在样本量的支持下，更加灵活的 QDA 在 validation data 上的表现的上升趋势更加明显。

对于未经标准化的数据，选取 125 个主成分，在 test data 上面得到 accuracy 为 24.6%；经过标准化的数据则选取 100 个主成分，在 test data 上面得到 accuracy 为 22.7%。QDA 模型的表现比 LDA 稍好一些。

第三个模型为 Multinomial Logistic Regression。其表现如下图所示：

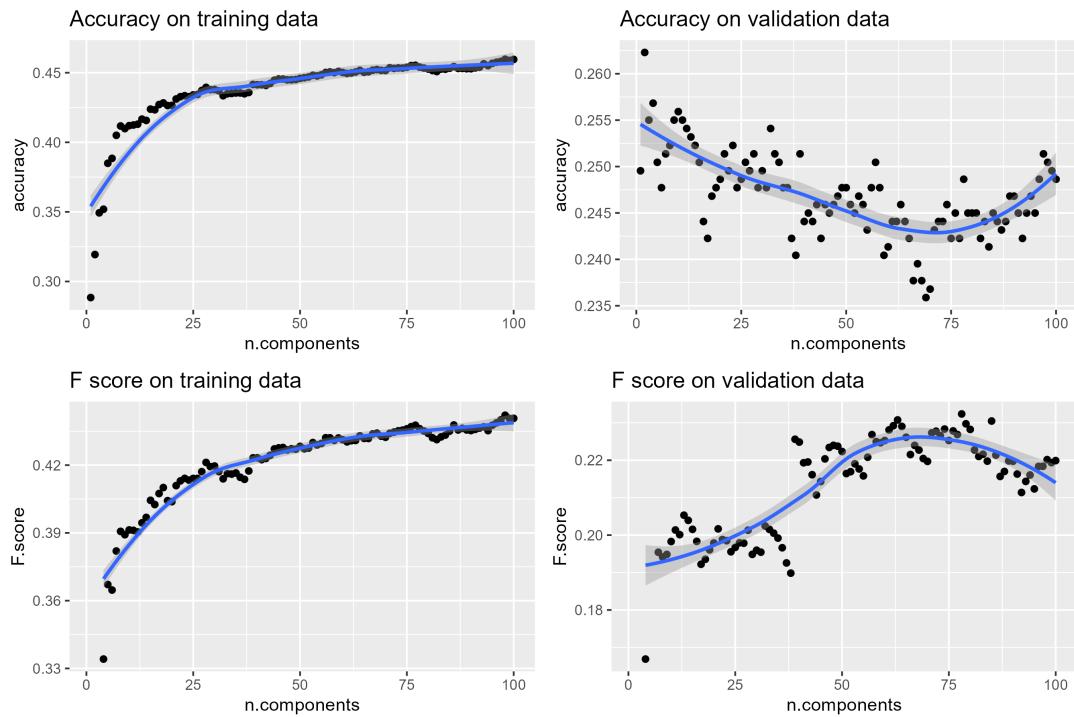


Figure 19: Accuracy and F score of Softmax Regression model (without scaling)

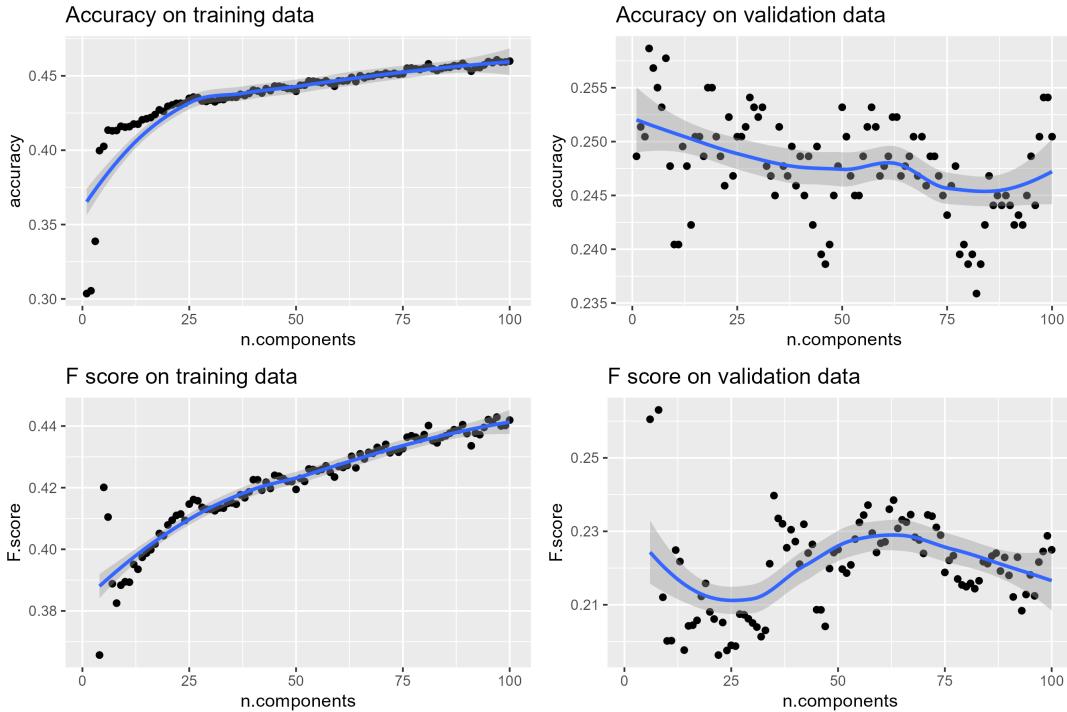


Figure 20: Accuracy and F score of Softmax Regression model (with scaling)

对于 multinomial logistic regression 的参数需要倍增 (对于五分类，需要 4 倍)，因此选取的主成分个数不宜过大出于训练难度的考虑，如果选取的主成分过多，也不容易收敛。因此限制选取的主成分个数不超过 100。

从图中可以发现，两个指标在 validation data 上面的表现比较特殊，对于未经标准化的数据，accuracy 先下降再上升、F score 却是先上升再下降 (峰值在 65 处)。对于经过标准化的数据，accuracy 几乎一直下降，F score 在主成分个数很少的时候较高 (可能是离群值)，随后也是上升再下降 (峰值在 63 处)。对于未标准化的数据，如果我们根据 accuracy 的表现选取维数  $k=1$ ，在测试集上的表现为 accuracy 28.6%，如果根据 F score 的表现选取位数  $k=65$ ，在测试集上 accuracy 为 23.1%。对于标准化后的数据，以类似的方式，选取  $k=1$  和  $k=63$ ，在测试集上 accuracy 分别为 26.8% 和 22.8%。

这种特殊的现象可以从数据的不平衡性上解释。训练数据集中，label 为 1 和 3 (也就是 negative 和 positive) 的较多，而 label 为 0, 1 和 4 的比较少 (即 very negative, neutral 和 very positive) 的比较少。这或许是因为判断情绪是积极和消极较为容易，然而认定一句话不带情绪则比较难。同时，句子中含有强烈情绪的频率也不会太高。这是一种普遍的规律，而不是某个数据集的特点，因此 test data 的 label 也有相似的不平衡特征。当预测变量比较少的时候，multinomial logistic regression 倾向于根据先验概率预测 label，而测试集的 label 和训练集分布类似，因此预测到正确 label 的概率就较高。例如未经标准化的数据，只取一个主成分训练模型，在 test data 上得到的 confusion matrix 如下所示：

|          | <b>0</b> | <b>1</b> | <b>2</b> | <b>3</b> | <b>4</b> |
|----------|----------|----------|----------|----------|----------|
| <b>0</b> | 0        | 0        | 0        | 0        | 0        |
| <b>1</b> | 9        | 31       | 61       | 178      | 151      |
| <b>2</b> | 0        | 0        | 0        | 0        | 0        |
| <b>3</b> | 14       | 35       | 99       | 282      | 233      |
| <b>4</b> | 0        | 0        | 0        | 0        | 0        |

可以发现模型将所有的预测都压在了 1 和 3 两个 label 上，而 test data 的 label 也有很多是 1 和 3，因此 accuracy 比较高。

然而，这种选择方式保证了较高的 recall，但 precision 则不好，F score 综合考虑 recall 和 precision，因此这种预测方式并不会得到较高的 F score。

### 5.1.3 讨论

下面对基础机器学习模型进行一些讨论：

**第一点，为什么本文中不使用 KNN 来解决分类问题？**这是因为即使经过了降维，我们得到的维度依然比较高，由于高维空间的稀疏性，KNN 的表现不好。

**第二点，为什么这些模型表现比朴素贝叶斯好？**这是因为基于词嵌入的这些模型打破了朴素贝叶斯的独立假设。嵌入向量将词语转化为语境的信息，然后通过平均操作将整个句子整合成一个向量，综合性更强从而更好地表达数据的特征。

**第三点，这些模型依然存在怎样的缺点？**首先，我们再将句子转化为向量的时候采取的是平均操作，没有设计权重，实际上模糊了一些很重要的信息，例如，形容词、副词也许更加重要；其次，没有考虑词语之间的联系，例如 lack 后加上含积极含义的词语时，这些词语的作用应该是使句子向更加负面的方向倾斜，而平均操作下，这些词语反而将句子整体的情感往积极的方向回拉；此外，这些模型仍然没有考虑词序的信息，而在情感分析中，语序恰恰是非常重要的，例如将 but 前后的内容互换，句子意思将截然不同，或者例如出现在 lack 后面的词语对句子的情感贡献应该与其本意相反。

**第四点，关于词嵌入向量的降维问题。**在 Word2Vec、GloVe 和 fastText 的原论文中，研究者们都选取 300 作为词向量的维度。这些颇有影响力的文章导致随后的研究者纷纷选择 300 维的词向量。由于深度学习的样本量较大，300 维的输入是可以接受的。本文在研究机器学习模型时，出于惯性进行了降维。那么降维对于模型的表现有何影响呢？不降维直接使用原数据在 test data 上进行预测，LDA 的 accuracy 为 18.6%（对比而言，降维之后的 accuracy 为 21.6%），QDA 的 accuracy 为 25.52%（对比而言，降维之后的 accuracy 为 24.6%），Softmax regression 的 accuracy 为 22.2%（对比而言，降维之后的 accuracy 为 23.1%）由此可见，在这个问题中进行降维帮助并不是很大。因此，应该在进行词嵌入的时候就对维度进行选择，而不是在获得嵌入向量之后再进行降维。如何选择词嵌入的维度是一个非常复杂的话题，涉及到优化 PIP 损失的 bias-variance trade-off，[3] 在本文中不再深入探讨。

**第五点，情感分类 label 所含有的 ordinal 的信息是否有助于提升模型表现** cumulative logit model 可以利用 ordinal 的信息，由于上面一点讨论得出的结论是降维对性能的影响不太明显，我们直接使用 300 维的嵌入向量作为输入，不再讨论维度数目。拟合模型，在测试集中得到的 accuracy 为 22.6%，相较于 softmax regression 的 23.1% 并没有什么增长。原因可能是 accuracy 这个评价指标不够全面，考虑 ordinal information 时，有可能只在程度上犯错，例如将 negative 判为 very negative，而不考虑 ordinal information 时则有可能犯更严重的错误，例如将 positive 判为 negative，然而在 accuracy 的评价体系下，这两者受到的惩罚是一样的。

## 5.2 深度学习模型

### 5.2.1 Comparison Table

| Model | Bidirectional | #Layers | Dropout | Training Time | Test Accuracy   |
|-------|---------------|---------|---------|---------------|-----------------|
| RNN   | No            | 1       | No      | 54.1s         | 23.8462%        |
| RNN   | Yes           | 1       | No      | 2m 6.3s       | 37.9638%        |
| LSTM  | No            | 1       | No      | 4m 5.9s       | 33.6652%        |
| LSTM  | Yes           | 1       | No      | 8m 53.1s      | 41.5385%        |
| LSTM  | Yes           | 1       | Yes     | 8m 13.1s      | <b>45.5204%</b> |
| LSTM  | Yes           | 2       | Yes     | 19m 31.6s     | 45.3394%        |
| GRU   | Yes           | 1       | Yes     | 6m 30.1s      | 44.3439%        |

### 5.2.2 结果分析

针对训练结果和数据集，我们分析得到了如下结果：

- 单向 RNN 和机器学习模型表现相仿。**本数据集句子长度中位数为 20 左右，且具有较长的右尾，因此在长句子中会出现明显的梯度消失问题，导致句子序列信息的权重没有得到更新，因此真正起作用的转变为全连接层，某种意义上退化为机器学习。（这里的图有问题，我需要的是没清洗过的句子的长度）

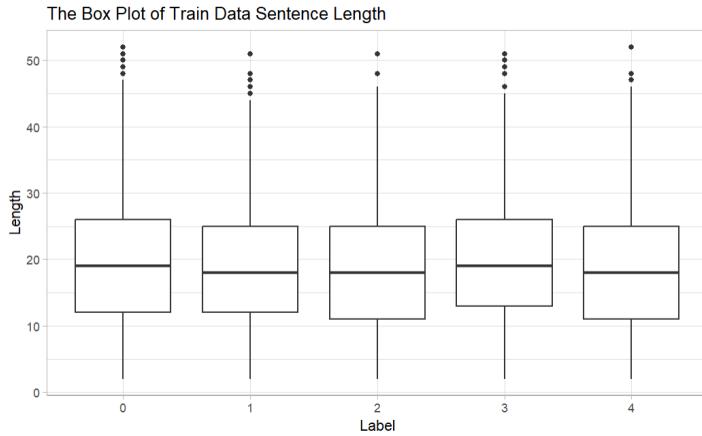


Figure 21: The Bar PLOT of Sentence Length

2. **双向信息提升明显。** 经过对比单向模型和双向模型的情感分类效果，我们发现双向信息主要针对一类具有明显特征的句子，即存在转折的句子，进行了优化。具体来说，如果只利用单向信息，即使存在门控单元等捕捉长序列信息的机制，主要起作用的仍集中于后半句话，即后半句话的权重被过分增大了，因此利用到的序列信息实际上是不完善的；而双向则保证了前半句和后半句的信息都被充分利用，因此分类效果就会有一个质的提升。
3. **Dropout 提升明显。** 根据对数据集和预训练词向量的分析，我们意识到本数据集具备的过拟合倾向事实上是内生的。我们采用了 300 维的词向量，特征空间维数非常高，而训练样本数仅有 8000 余条，因此很难进行非常有效的参数学习，导致很容易拟合数据集里的噪声，泛化能力较弱。再加上本数据集存在一定的数据不平衡的问题，即 positive 和 negative 的数据量比例较大，导致模型很容易关注这两类标签，进而导致过拟合。而 dropout 通过随机关闭某些神经元，一定程度上缓解了这一问题。
4. **时间复杂度较高。** 从表中可以看出，训练一个有 dropout 的 LSTM 模型需要长达 8 分钟的时间，这也是由本数据集和词向量的特征决定的。注意到 LSTM 模型的时间复杂度约为  $O(ND^2)$  (事实上更精确的估计应为  $O(NDH)$ )，其中  $N$  为序列长度， $D$  为词向量维度， $H$  为隐藏层维度，但由于需要避免过大的维度差距，因此隐藏层维度和词向量维度基本相仿，故将模型复杂度估计为  $O(ND^2)$ ，而本训练集采用的词向量为 300 维，因此时间较长。

结合训练时的损失曲线和准确率曲线，我们可以进一步证实和补充上述论述：

1. **双向信息的有效性。** 在这里我们以 RNN 和双向 RNN 为例，从图35和图25可以看出，仅仅使用单向信息时，RNN 模型的损失会在前两个 epoch 迅速下降，之后则基本不再下降；而双向 RNN 也表现出在前两个 epoch 迅速下降的特征，但之后依然存在平缓下降的趋势。这可以从侧面证明双向模型的确相比单向模型利用了更多的信息。与此同时，从图23和图26中可以看出，单向 RNN 的准确率在前两个 epoch 迅速上升，之后就不断震荡，说明其模型复杂度较低，因此捕捉的信息更少，不适合该训练集；相反，双向 RNN 的准确率在训练过程不断上升，甚至表现出明显的过拟合倾向，体现出模型复杂度较高，捕捉的信息更多，更适合本训练集。(单向 LSTM 和双向 LSTM 也表现出类似的特征，在这里由于篇幅原因不再展开分析)

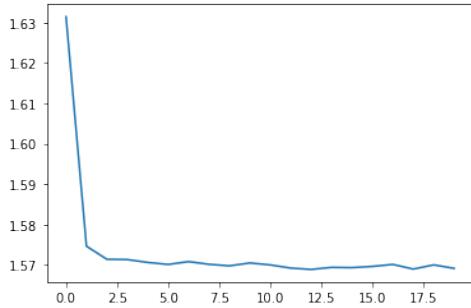


Figure 22: RNN 损失曲线

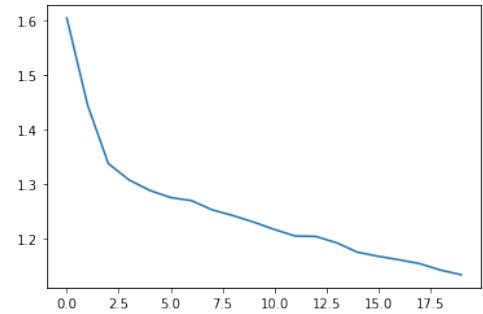


Figure 25: 双向 RNN 损失曲线

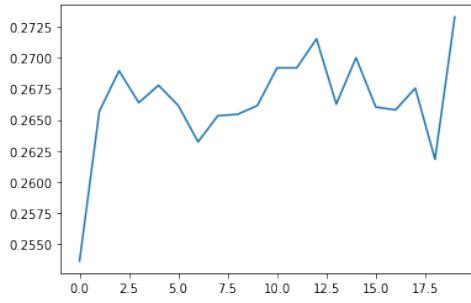


Figure 23: RNN 训练集准确率曲线

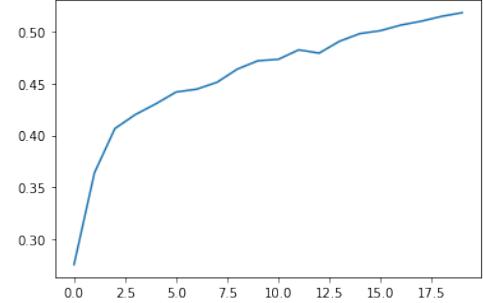


Figure 26: 双向 RNN 训练集准确率曲线

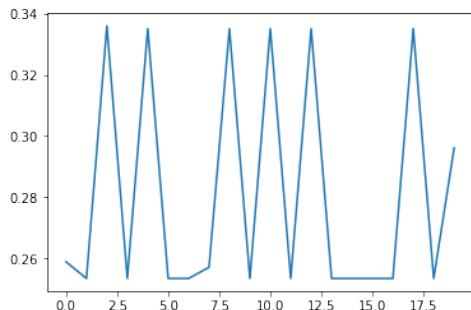


Figure 24: RNN 验证集准确率曲线

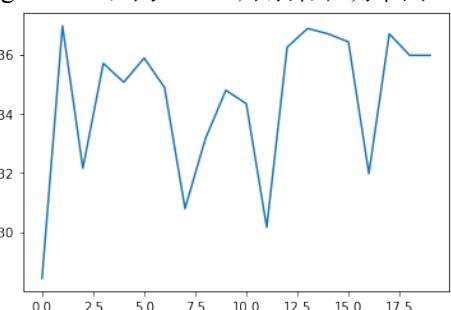


Figure 27: 双向 RNN 验证集准确率曲线

2. **Dropout 的有效性。**从图28和图34中无法分辨出有无 Dropout 对模型的影响，但是从图29和图32中可以部分看出 Dropout 对模型的影响，即加入 Dropout 后训练集的准确率明显下降，因此极大缓解了过拟合问题。从图30和图33则可以更直观的看出 Dropout 对于缓解过拟合问题的作用。注意到无 Dropout 时，模型的验证集准确率迅速上升然后进入震荡期，鲜明地显示了模型的过拟合；而有 Dropout 时，模型的验证集准确率先是一个较快的上升过程，进而开始相对平缓的上升，说明模型的泛化能力一直在上升，而非在不断拟合训练集的噪声。

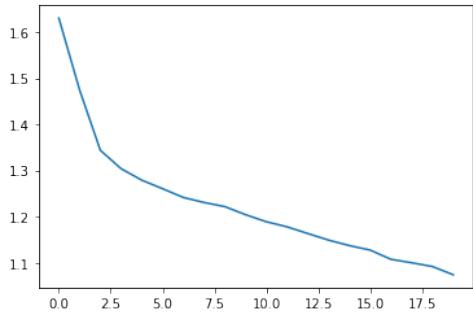


Figure 28: 双向 LSTM (无 Dropout) 损失曲线

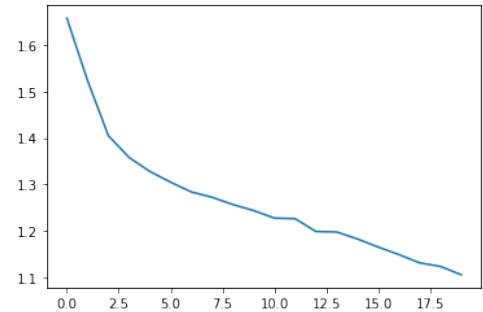


Figure 31: 双向 LSTM (Dropout) 损失曲线

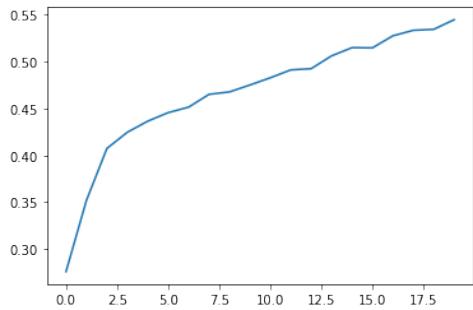


Figure 29: 双向 LSTM (无 Dropout) 训练集准确率曲线

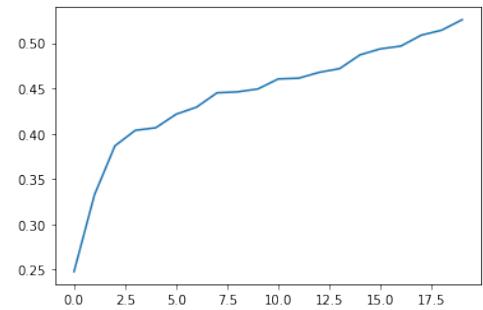


Figure 32: 双向 LSTM (Dropout) 训练集准确率曲线

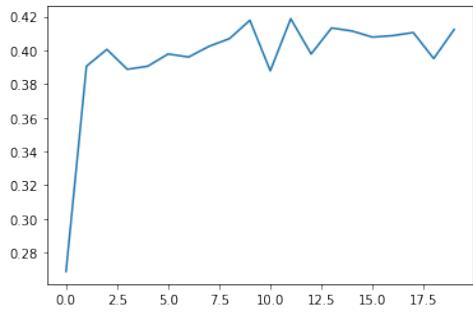


Figure 30: 双向 LSTM (无 Dropout) 验证集准确率曲线

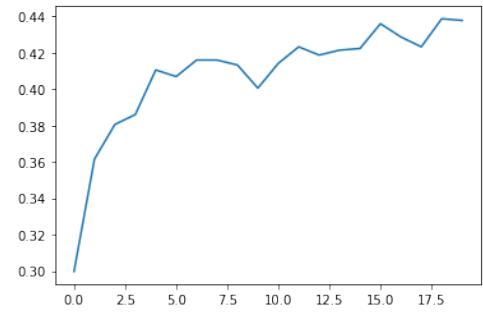


Figure 33: 双向 LSTM (Dropout) 验证集准确率曲线

3. **Adam 优化器的有效性和学习率调参。**以单向 LSTM 的损失曲线为例，我们可以鲜明地看到训练过程存在一段很长的平台期，这往往代表着模型陷入了局部最优点，这侧面证明了选用 Adam 优化器的重要性。事实上，在训练初期，曾尝试过使用 SGD 等计算更为便捷但不包含动量的优化器，但模型权重更新非常缓慢，损失曲线常常进入平台期，难以收敛。从单向 RNN 和单向 LSTM 的损失曲线可以看出，在最初的若干 epoch，均存在一段快速下降的阶段，若学习率过大，则有可能由于梯度下降过快反而导致错过全局最优点，因此最终选择了 0.0001 作为 Adam 优化器的学习率。

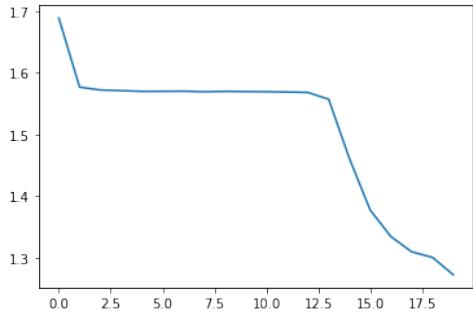


Figure 34: 单向 LSTM 的损失曲线

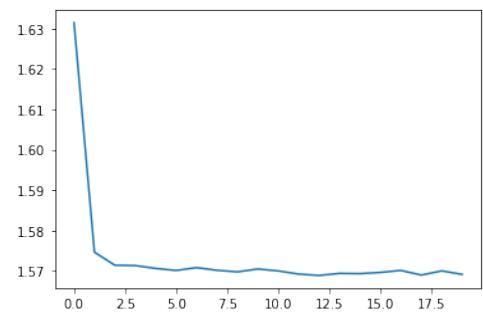


Figure 35: 单向 RNN 的损失曲线

### 5.2.3 讨论

下面针对深度学习模型进行一些讨论：

**第一点，模型目前存在哪些问题，可以如何优化网络结构？**在上述循环神经网络的训练和分析过程中，我们意识到其主要存在两个问题：计算复杂度和不可解释性，不可解释性主要是因为 LSTM 模型和 GRU 模型的门控机制都是将特征选择的权重内隐在中间隐藏状态，因此我们并不能得知其如何利用了序列信息，什么样的序列信息能够最优化模型表现。而针对循环神经网络的计算时间复杂度分析可见上文的结果分析。

针对这两个问题，在查找和阅读文献的基础上，我们提出了一个新模型：

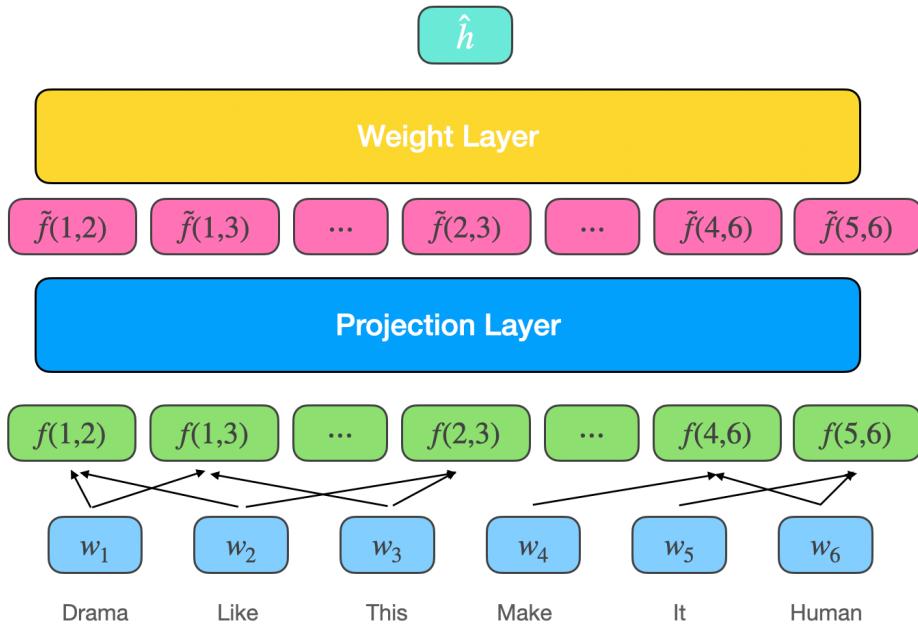


Figure 36: 模型架构

其中部分关键计算公式如下所示：

$$f(i, j) = [w_i, w_j, w_i - w_j, w_i \otimes w_j]$$

$$\tilde{f}(i, j) = h^T f(i, j)$$

$$\alpha(i, j) = \frac{u^T \tilde{f}(i, j)}{\sum_{i,j} u^T \tilde{f}(i, j)}$$

$$\hat{h} = \sum_{i,j} \alpha(i, j) \tilde{f}(i, j)$$

该模型的架构图如图36所示，其中  $w_i$  就代表词向量，其中  $u$  和  $h$  均为模型中可学习的参数。根据前面 EDA 的结果，2-gram 筛选得到的词组的表义性比较明确，我们在这里就利用了这个性质，采用两两组合的形式来提取序列信息和词组信息，这里我们采用的特征计算方法是根据 [1]，分别表示 concatenation, element-wise 的差异和 element-wise 的相似度。得到两两的信息后，我们通过投影将矩阵转化为向量，再通过如图所示的流程为每个特征赋予权重，这种权重计算方法，其表征了  $f(i, j)$  对于最终预测的解释性 [2]，因此该模型天然具有自解释性；最后我们将其转化为 5 类预测。

这样做就解决了前述模型的问题：

1. **时间复杂度优化。** 注意到现在计算任意一组特征的时间复杂度为  $O(D)$ ，因为现在是向量计算而非 LSTM 中的矩阵计算；而总共需要计算  $O(N^2)$  组特征，因此总的时间复杂度为  $O(N^2 D)$ 。注意到这其实是适配本数据集的特征的，句子长度中位数仅为 20，而词向量维度则有 300，因此  $D$  远远大于  $N$ ，所以在本数据集下该模型的时间复杂度远远低于 LSTM 模型。
2. **权重可解释性。** 由于我们现在可以手动选择特征的计算方式，因此就可以明确权重的意义，即每个特征的解释性。我们进行手动优化特征计算方式，将原本需要花很长时间的深度学习调参过程转变为特征工程，这也有利于整个模型的可解释性，也利于我们发现最终是什么因素能够更好的获取情感倾向。

由于时间所限，我们并没有完成该模型的全部搭建和调参过程，仅从理论上分析了其可行性和优势，若之后有充分时间，会更完善的呈现模型的训练结果。

**第二点，为什么使用无法体现类别 ordinal 信息的交叉熵作为损失函数？** 主要原因在于在机器学习分析过程中，已经尝试了纳入 ordinal 信息进行训练，但效果并不明显，而深度学习模型重新训练的时间成本非常高，还涉及到调参等需要大量调试的环节，因此由于期末周时间所限，并没有更换损失函数以重新训练。而且针对本数据集，使用具备 ordinal 的损失函数（例如 MSE 等）还可能导致异想不到的效果。具体来说，由于本数据集 positive 和 negative 标签多于其他标签，因此模型为了减少损失，可能会选择更多地预测 neutral 这一标签。这是因为单个样本一旦预测错情感倾向（如正向预测为负向），则在 MSE 下，损失至少是 4；相反，如果预测 neutral，则单个样本损失最多是 4。又因为本数据集 positive 和 negative 的样本较多，因此单个样本的损失以较大的概率是 1，以较小的概率是 4，最终强迫模型更多地预测 neutral 这一标签，导致天然存在有偏性。

## 6 总结与展望

### 6.1 情感标注的主观性

由于对电影评论的情感标注存在主观性，不同的标注者对同一段评论的情感评价不一定一致，而且同一个标注者在不同时间、不同地点、不同状态下对同一段文本的标注都有可能不同，尤其是 very negative 和 negative 之间，positive 和 very positive 之间都可能存在混淆。因此，该五分类问题实际上是一个非常难的问题。目前已知的模型表现最好的也只有 59.8 的% 的 accuracy，这也可以从侧面说明该问题的难度，以及标注上的主观性。

对于这种主观性，我们可以考虑将五分类问题转化为三分类问题（将 very negative 和 negative 都归为 negative，positive 和 very positive 都归为 positive），这样可以规避一部分标注的缺陷和问题的难度。

### 6.2 特征工程：词与词之间的关系

在上述的特征工程过程中，我们直接借鉴了论文 [1] 的特征提取方式，但事实上，我们在之后至少可以从两个方面进行优化：

1. **引入注意力机制。** 我们可以首先尝试使用包含 attention 的隐藏层以更好的提取词与词之间的信息。由于前述我们已探讨过情感分类的关键问题是找到为每个词赋权重的方式，而注意力机制恰恰可以关注重要信息，自动学习并关注与情感分类最相关的词语或上下文片段，从而提取重要的特征。因此在计算  $f(i, j)$  之前，先利用注意力机制为不同位置或单词赋权，可以更好的的表征出词与词之间的关系。
2. **针对数据集优化  $f(i, j)$  的计算方式。** 在未来的特征工程中，可以着重发掘针对本数据集更为适用的  $f(i, j)$  方程，以保证更有针对性，也可以增强模型的可解释性。

### 6.3 恢复数据平衡性

在之前模型结果的分析过程中，可以发现数据平衡性导致了很多问题，也限制了很多模型的发挥。因此在未来的处理过程中，还可以通过多种方式恢复数据的平衡性，重新调整和训练模型，以期得到更为精准的结果。

## 7 附录

### 7.1 小组成员分工

- 李培森：Exploratory Data Analysis, Naïve Bayes, Random Forest
- 于骏浩：Data Cleaning, Exploratory Data Analysis, Deep Learning Models based on Embedding
- 周子逸：Dimension Reduction, Basic Machine Learning models based on Embedding(LDA, QDA, Softmax Regression, Cumulative Logit Model)

### 7.2 代码与数据集

<https://cloud.tsinghua.edu.cn/d/c08c459cf2d48eda92b/>

## References

- [1] Lili Mou, Rui Men, Ge Li, Yan Xu, Lu Zhang, Rui Yan, and Zhi Jin. Natural language inference by tree-based convolution and heuristic matching, 2016.
- [2] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps, 2014.
- [3] Zi Yin and Yuanyuan Shen. On the dimensionality of word embedding. *CoRR*, abs/1812.04224, 2018.