|  |
| --- |
| **机器学习** |
| **Quora同类问题识别** |
| 主 研 人：杜思君  参 研 人： |
| 审 核 人：  方 向：工业算法案例研究  版 本 号：A |
|  |

**二〇一九年二月**

**版本更新**

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **更新描述** | **作者** |
| 2019/03/29 | A | 初稿 | 杜思君 |

**选题表格**

|  |  |  |  |
| --- | --- | --- | --- |
| **时间** | **竞赛名** | **竞赛背景描述（50字以内）** | **类型（分类/回归）** |
| 2019/3/29 | [Quora Question Pairs](https://www.kaggle.com/c/quora-question-pairs) | 根据问题内容，定位已经被回答过的类似问题  避免用户问重复的问题，也节省答题人的时间 | 聚类（Cluster） |

目录

[1. 背景描述 4](#_Toc25575_WPSOffice_Level1)

[1.1 竞赛赛题描述 4](#_Toc5940_WPSOffice_Level2)

[1.2 评估指标描述 5](#_Toc12357_WPSOffice_Level2)

[2. 数据来源及描述性统计分析 5](#_Toc5940_WPSOffice_Level1)

[2.1 大赛数据来源 5](#_Toc22683_WPSOffice_Level2)

[2.2 数据的描述性统计 5](#_Toc16174_WPSOffice_Level2)

[2.2.1 数据基本情况描述： 5](#_Toc5940_WPSOffice_Level3)

[2.2.2 数据字段介绍： 5](#_Toc12357_WPSOffice_Level3)

[2.2.3 数据描述性统计 6](#_Toc22683_WPSOffice_Level3)

[3. 优秀算法思路 11](#_Toc12357_WPSOffice_Level1)

[3.1 方案一 1](#_Toc32293_WPSOffice_Level2)1

[3.1.1 方案一数据预处理及特征工程部分方案 11](#_Toc16174_WPSOffice_Level3)

[3.1.2 方案一模型设计、建立部分方案 11](#_Toc32293_WPSOffice_Level3)

[3.1.3 方案一结果、排名等 14](#_Toc29922_WPSOffice_Level3)

[3.1.4 方案一算法流程图 14](#_Toc4787_WPSOffice_Level3)

[3.2 方案二 1](#_Toc29922_WPSOffice_Level2)5

[4. 算法比较 17](#_Toc22683_WPSOffice_Level1)

[表4-1 算法比较 17](#_Toc7253_WPSOffice_Level2)

[5. 总结与展望 17](#_Toc16174_WPSOffice_Level1)

[5.1 总结 17](#_Toc23198_WPSOffice_Level2)

[5.2 建模思路 17](#_Toc16619_WPSOffice_Level2)

1. 背景描述

除了在Quora之外，还有哪里可以让物理学家帮助厨师解决数学问题并获得烹饪技巧吗？ Quora是一个获取和分享知识的地方，这里也是是一个提出问题并与提供独特见解和高质量答案的人联系、交流的平台。这使得人们能够相互学习，能更好地了解世界。  
 据统计，每个月会有超过1亿人访问Quora，因此很多人提出类似措辞的问题也就不足为奇了。具有相同意图的多个问题可能会导致想要了解问题得任花更多时间找到问题的最佳答案，并使回答者觉得他们需要回答同一问题的多个版本。 Quora向来重视规范性问题，因为它们为活跃的想要了解和提出问题的人以及问题的回答者提供了更好的体验，并且从长远来看为这两个群体提供了更多价值。

* 1. **竞赛赛题描述**

目前，Quora使用随机森林模型来识别重复的问题。在本次比赛中，Quora的拥有者希望参赛者通过应用更加先进的技术来识别问题对是否重复，从而解决这种自然语言处理问题。这样做可以使用户更容易地找到问题的高质量答案，从而改善Quora的作者，提问者和读者的体验。

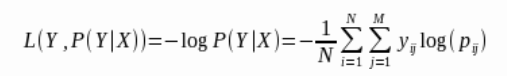
本次比赛的目的是预测哪些题目包含两个意思相同的题目。最基本的事实是一组由人类专家提供的标签。但基本事实标签在本质上是主观的，因为句子的真正含义永远无法确定。人类标记也是一个“争辩”的过程。因此，这个数据集上的基本事实标签应该被认为是“已通知的”，但不是100%准确，并且可能包含不正确的标签。我们认为标签通常代表一个合理的共识，但数据集中的项目通常不是这样。

**1.2 评估指标描述**

竞赛中评估模型优劣的指标。

评估指标是关于预测值和基本事实之间的对数损失（log loss）。

对数损失函数的计算公式如下：



其中，*Y* 为输出变量, *X*为输入变量, *L* 为损失函数. *N*为输入样本量, *M*为可能的类别数,*yij*是一个二值指标, 表示类别 *j* 是否是输入实例 *xi*的真实类别. *pij*为模型或分类器预测输入实例 *xi*属于类别 *j* 的概率.

1. 数据来源及描述性统计分析

**2.1 大赛数据来源**

赛题的训练集数据全部源自Quora上真实的问题数据，作为反作弊措施，Kaggle已经使用计算机生成的问题对补充了测试集，所以测试集中的某些行并不是来自Quora，不计入分数。

以下是数据的超链接：

<test.csv> （测试集数据）

<train.csv> （训练集数据）

<sample_submission.csv> （提交样例）

**2.2 数据的描述性统计**

**2.2.1 数据基本情况描述：**

数据集包含有训练集和测试集，训练集包含6个字段，测试集包含3个字段，具体含义如2.2.2所示

**2.2.2 数据字段介绍：**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **变量含义** | **变量类型** | **缺失率** |
| id | 训练集问题对的id | 连续（int64） | 0% |
| qid1 | 问题一的id(唯一) | 离散（int64） | 0% |
| qid2 | 问题二的id(唯一) | 离散（int64） | 0% |
| question1 | 问题一的全文 | string | 0% |
| question2 | 问题二的全文 | string | 0% |
| is\_duplicate | 目标变量，如果question1和question2具有基本相同的含义，则设置为1，否则设置为0。 | Binary | 0% |

Table 2.2.2-1 train数据表字段介绍

|  |  |  |  |
| --- | --- | --- | --- |
|  | **变量含义** | **变量类型** | **缺失率** |
| test\_id | 测试集中问题对的id | 连续（int64） | 0% |
| qid1 | 问题一的id(唯一) | 离散（int64） | 0% |
| qid2 | 问题二的id(唯一) | 离散（int64） | 0% |

Table 2.2.2-2 test数据表字段介绍

**2.2.3 数据描述性统计**

1. 训练集中问题对统计：

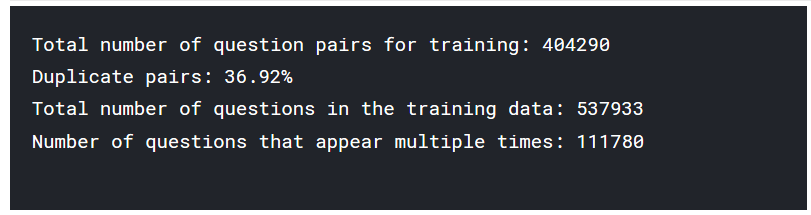


Figure 2.2.3-1: 训练集问题对统计描述

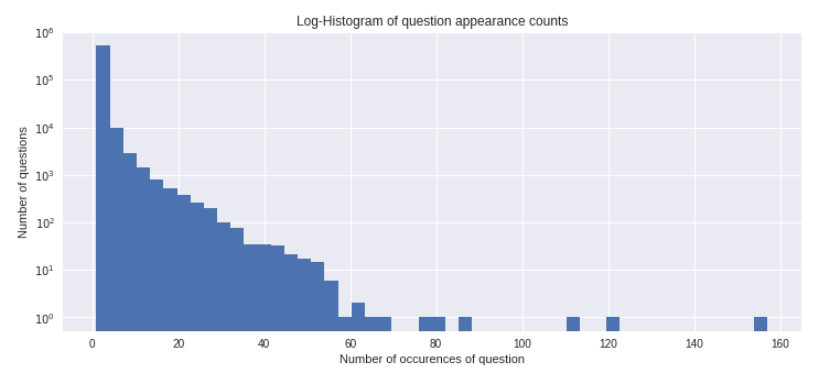


Figure 2.2.3-2: y训练集中问题出现次数与问题数量的对数直方分布图

大多数问题只出现几次，很少出现几次问题（并且出现了几次问题）。 一个问题出现超过160次，但这是一个异常值。  
 还可以看到这个数据集中有37％的正面类； 由于评估使用的是LogLoss指标，而LogLoss查看实际预测而不是预测顺序。

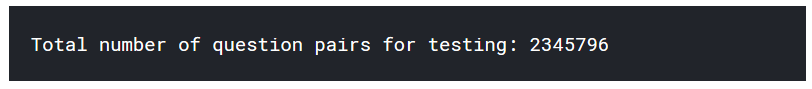
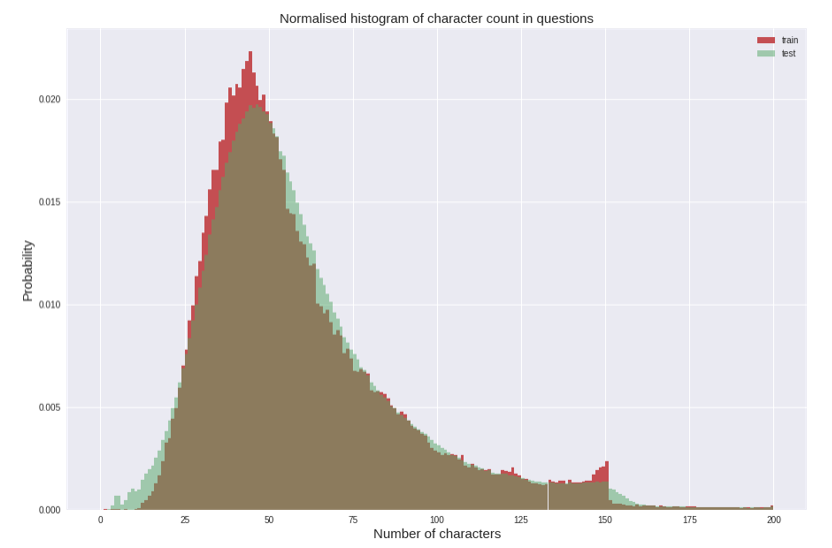
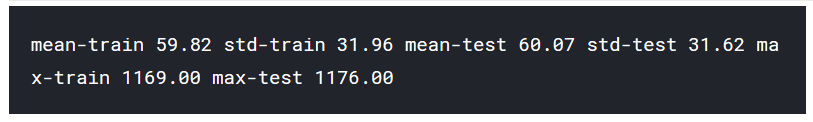
1. 训练集问题个数统计

Figure 2.2.3-3: 训练集中问题总数统计

1. 训练集和测试集问题长度比较统计

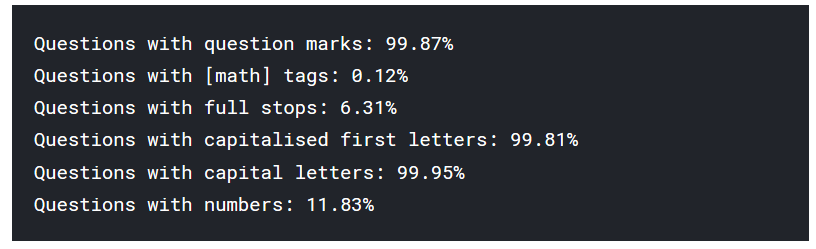
**Figure 2.2.3-4: 训练集和测试集中问题字符个数的分布情况**

****

**Figure 2.2.3-5: 训练集和测试集中问题字符个数的统计描述**

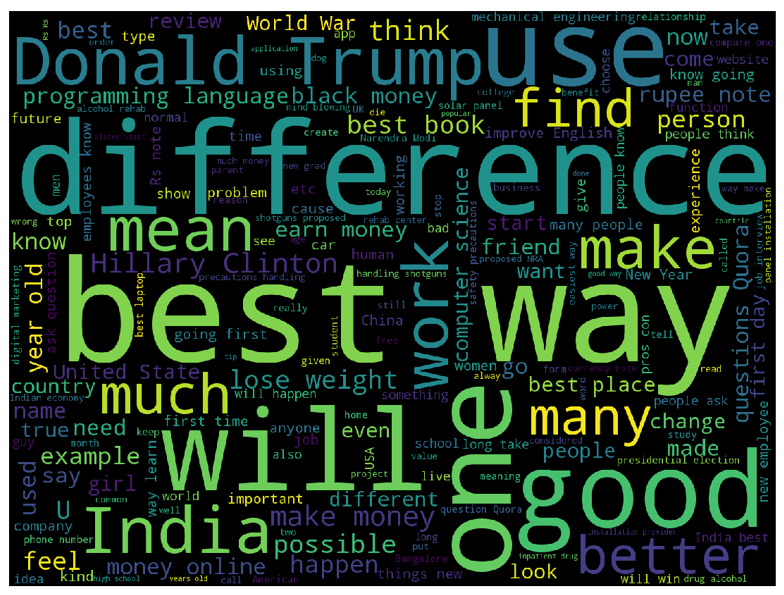
由图2.2.3-可见，看到大多数问题都在15到150个字符之间。 但似乎测试集的分布与训练集有一点点不同，但不是太多（但似乎在测试集中分布更顺畅）  
 值得注意的是，对于大多数问题，训练集的150个字符的陡峭截止，而测试集在150后缓慢下降。这可能是Quora对某些问题大小限制。并且训练集和测试集的最大值都不到1200个字符 - 尽管超过200个字符的样本就已经非常罕见。

e) 语义分析—标点符号的用法



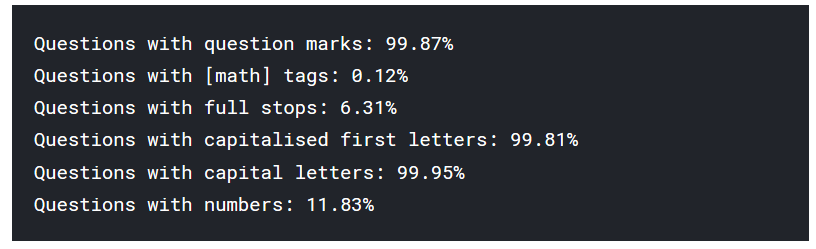
**Figure 2.2.3-6 标点符号统计**

d） 出现最多的词统计



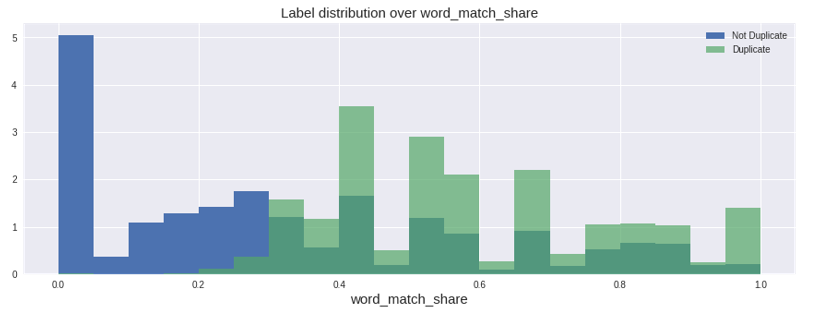
**Figure 2.2.3-5 最常见的词**

1. 语义分析—标点符号的用法



**Figure 2.2.3-6 标点符号统计**

f） 单词匹配



**Figure 2.2.3-6 单词匹配统计**

3. 优秀算法思路

* 1. **方案一**

**3.1.1**方案一数据预处理及特征工程部分方案

**a) 特征分类**

选手将特征分为三类，分别是：嵌入特征，经典文本挖掘特征和结构特征。  
**嵌入特征：**

* 词嵌入特征（Word2Vec）
* 句（子）嵌入特征（Doc2Vec，Sent2Vec）
* 使用来自SNLI训练的ESIM模型的密集层的编码问题对

**注**：句子嵌入受到挑战，但与Word2Vec相比，信息量不大

**经典文本挖掘特征**：

* LDA和LSI嵌入的相似度量。
* 对于1至8的字符（TFIDF重新加权或不加重）的相似性度量。
* Abhishek和owl的共享功能。
* 编辑和序列匹配距离，当问题结束相同或开始相同时，最多1、2、…、6个常见标记的百分比
* 问题长度的差异。
* 大写字母，问号等的数量。
* 问题1/2处以“Are”，“Can”，“How”等开头的指标以及所有相应的数学工程

我们还使用stanford corenlp到tokenizer，postagger和ner来预处理某些深度学习模型的文本输入。

**结构特征（即图表）**：

* 选手从连接的训练集和测试集中的问题对之间的边缘构建的图形中构建了密度特征。选手计算了问题1和问题2的领域数，最小值，最大值，交点，并集，主边切割时的最短路径长度。
* 选手进一步的建立了密度特征来计算相邻问题的相邻数......并且计算相邻问题的相邻问题..（开始）。选手还计算了更高阶的邻域，它们也是较低阶的邻域（循环）。
* 选手尝试了不同的图形结构：构建了无向和有向图（从问题1到问题2的边缘），选手还试图将问题1的密度特征与问题2的特征分开，以产生除交换特征之外的非交换特征。
* 选手构建了描述该对所属的连接子图的特征：边数，节点数，列中边的百分比。
* 选手还在子图上计算了相同的特征，这些子图仅从问题的边缘构建，这些问题都出现不止一次。选手想要的是删除他们认为通过改变其结构破坏图形特征的假问题。

最后，与其他团队一样，该选手使用一些初始模型对图表进行加权。他们尝试了logit并重新调整了预测，但原始预测效果最好。此外，还使用了一个相似特征对图形进行加权。

**3.1.2** 方案一模型设计、建立部分方案

**1）模型建立：**

选手为***NNets***开发了两种主要架构：***Siamese***和***Attention Neural Networks***。

* Siamese LSTM带有预训练的Glove嵌入
* 具有预训练的FastText嵌入的可分解注意力（<https://arxiv.org/abs/1606.01933）。该模型在cv上达到~0.3>
* ESIM（https://arxiv.org/abs/1609.06038），带有预训练的FastText嵌入。这是选手使用的最好的纯粹的深度学习NLP模型，它在CV上达到了~0.27。但是这个模型运行时间太长，选手只在第一个堆叠层添加一次
* 选手注意到DL复杂体系结构在第一层堆叠层中起作用，但在第二层上没有比简单MLP更好

其中一个关键问题是选择并将选手的一些传统特征纳入这些网络。选手使用FastText和Glove预训练的嵌入式使训练= False，因为选手尝试对它们进行微调并没有带来任何改进。  
 最终，在文本序列和选手的图形/文本挖掘特征上训练的神经网络被证明是最好的单一模型。  
 最后，选手还尝试在角色级别上训练***Siamese***模型，以便为堆叠模型提供进一步的多样性，但很难说它是否真的有用。  
 然后，选手尝试了更多经典算法来利用图形功能，例如XGB / LGBM，它们像往常一样得到了不错的结果。

**2） 重新缩放（调节）：**

为了平衡训练集和测试集之间的目标分布的差异，选手还看了一点关于***sweezyjeez***的分析***（https://www.kaggle.com/c/quora-question-pairs/discussion/31179再次感谢您的贡献，这帮助了几乎所有的参与者）***  
 选手通过优化重新缩放来减少对数损失。选手没有找到更好的假设来模拟测试数据集中数据的分布，但通过在数据的本地子样本上使用它来使其更准确。

我们发现3个周边的训练集/测试集非常不同：

* + - 周边1：qid1count = qid2count = 1
    - 周边2：minqidcount = 1＆maxqidcount> 1
    - 周边3：minqidcount> 1

选手尝试了特殊重新缩放和相同的重新缩放。它适用于第一层模型，但随着选手在堆叠中的深入，发现公共重新缩放不够强大，而周边的重新缩放太强。选手优化了他们的重新缩放，使它在这两种方法之间处于中间位置，与公共重新缩放相比，它有助于获得~0.001的得分。

**3）模型堆叠**：  
**选手做了4层堆叠：**

* **第1层**：

大约300个模型，*Paul*和*Lam*的神经网络，以及*XGB，LGBM*等经典算法，它们的效果十分不错，此外还有很多*Scikit- learn* 库里的分类算法（*ET，RF，KNN*等）

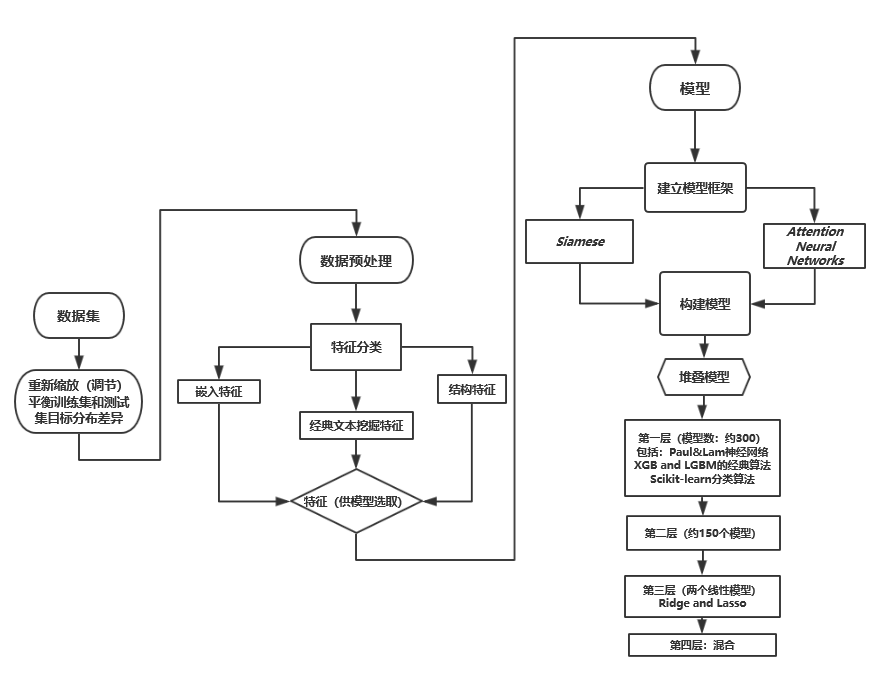
* **第2层：约150模型使用：**  
    **i）** 所有特征输入  
    **ii）** 上述所有算法的预测  
   **iii）** 选手添加了最好的*L1*纯文本*ESIM*模型的隐藏层
* **第3层：2个线性模型**  
    **i )** 在3个最小*Spearman*相关的*L2*预测上，按周长（3个周长，基于最小/最大度）创建脊  
    **ii)** *Lasso*使用*logit预处理*所有*L1*和*L2*预测
* **第4层：混合**  
   55/45，根据公共LB得分（最终和最佳提交）

**3.1.3** 方案一结果、排名等

**排名：1/3307**

**得分：0.11579（logloss）**

**3.1.4** 方案一算法流程图



**Figure3.1.4-1:算法流程图**

4. 算法比较(总结)

|  | **评估指标** | **特征工程** | **基础算法** | **基本库** |
| --- | --- | --- | --- | --- |
| **算法1** | **logloss** | **特征分类，分词** | **Xgboost，stacking等等** | **Sklearn，xgboost, natural networks** |

5. 总结与展望

**5.1 总结**

本题是一道结合自然语言处理的聚类问题，难点在于语义的分析和理解，处理起来较为复杂，想达到好的结果也非常不容易，在读过几位参赛选手的分析报告之后，大致了解的这里以自然语言处理为主的问题的解决流程和常用模型，希望以后有机会试着去学习运用。

**5.2 建模思路**

由于此前对此类NLP问题了解较少，一时间难有好的思路，有机会学习后再来思考。