机器学习第三次作业实验报告

病历文本多分类问题

摘要：

本次实验主要解决两个问题：根据收诊病历信息判断病人所患糖尿病并发症；分析预测病人的住院时间。对第一个问题，我们主要用 TF-IDF 获取词向量，并用线性核 SVM 对文本进行分类；对第二个问题，我们用 TF-IDE 获取词向量，并用不同的回归预测器进行预测。第一问在验证集上的准确率达到了 70%左右，第 2 问的绝对平均误差在两天左右。

遗憾的是，我们通过测试发现这两个数值并不能说明我们的模型成功刻画了数据集的特征。本篇报告中，我们将对上述内容做出进一步阐释。

1．导言

时至今日，文本分类问题的解决方案有很多，具体内容在第一次报告的导言中有过讨论。在本次实验第 1 问中，我们的目标是解决一个多分类、长文本、样本不均衡、小数据集的分类问题。可以发现此次的数据集处理是比较困难的。

2．实验过程

对第 1 问，我们有如 Fig.1 所示的算法流程：

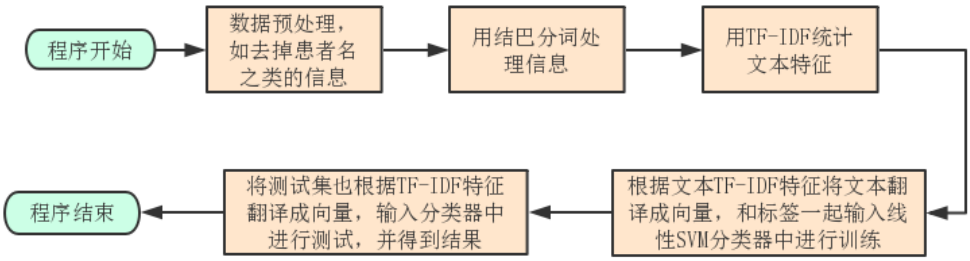
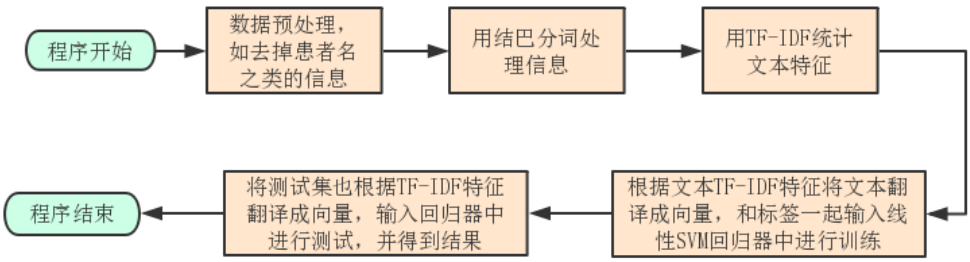


Fig.1 Algorithm flowchart for problem 1

我们采用 TF-IDF，基于如下的理由：

1. 一开始我们并没有采用 TF-IDF 来将文本转成向量，而是考虑的 word2vec 算法。但是通过进一步的分析，我们发现这个问题不适合 word2vec 算法。数据集太小，而文本又过长。Word2vec 中一个词常常有上百个维度，那么一个文本转成向量可能有上万个维度。想要在只有一千多个样本的数据集中训练出上万维度的向量是不现实的。因此 word2vec 算法不合适
2. 分析发现，不同的并发症对应的病历在词汇的偏向性上很强。例如，一个足病患者，常常会设计“溃疡”、“流脓”等词，眼病患者的病历常常涉及到“视物不清”、“模糊”等词。因此用我们考虑用 TF-IDF 来将文本转成词向量。

对第 2 问，我们的算法处理流程和第 1 问类似，如 Fig.2 所示：




```
方差可解释性 0.5799569059187349
绝对误差 2.4322082459572223
均方误差 21.83339108338772
```

Fig.2 Algorithm flowchart for problem 2

3. 结果分析

3.1 实验环境

硬件：

处理器 Intel® Core™ i5-7200U CPU @2.50GHz 2.71GHz

内存 8.00GB

软件：

Windows10, Python 3.6.8

3.2 参数说明

使用 sklearn 默认参数

3.3 实验结果

3.3.1 Q1

```
pred acc 0.7084870848708487
-------------------- 肾病 --------------------
肾病 0  酮症 3  心脏病 0 眼病 1  周围神经病 8 足病 1
-------------------- 酮症 --------------------
肾病 10 酮症 0  心脏病 0 眼病 2  周围神经病 6 足病 0
-------------------- 心脏病 --------------------
肾病 1  酮症 2  心脏病 0 眼病 0  周围神经病 0 足病 0
-------------------- 眼病 --------------------
肾病 8  酮症 0  心脏病 0 眼病 0  周围神经病 3 足病 0
-------------------- 周围神经病 --------------------
肾病 10 酮症 8  心脏病 0 眼病 0  周围神经病 0 足病 1
-------------------- 足病 --------------------
肾病 3  酮症 2  心脏病 0 眼病 0  周围神经病 10足病 0
```

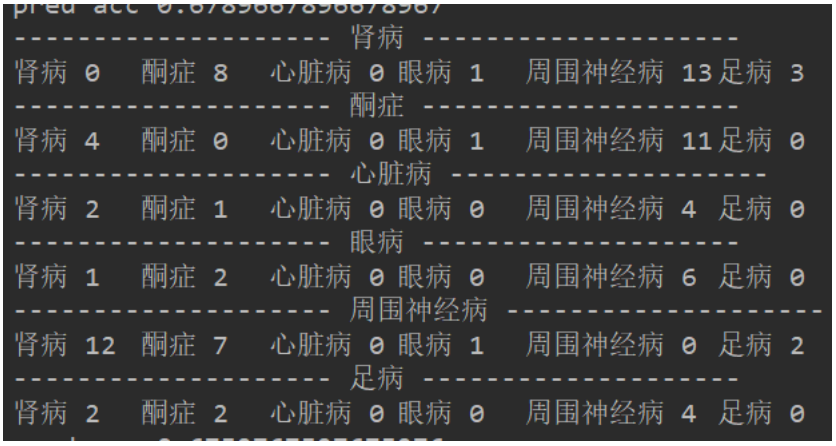
在处理第 1 问时，我们在得到 70%的结果后又进行了多种测试。

a. 提取文本中包含的检查信息，例如随机血糖、尿肌酐之类的检查，并将其结合 TF-IDF 特征作为输入。遗憾的是，这种操作并没有提升分类器性能，而且也没有使分类器性能下降。至少可以说这种操作对分类器性能的影响不明显。

b. 我们猜测可能是 TF-IDF 统计到的单词种类太多，训练效果不好。所以我们考虑手动构建需要的单词。最终我们只需要关注这些词在文本中是否出现即可，这有助于文本的特征更加鲜明。令人遗憾的是，尽管花费了较多精力构建这

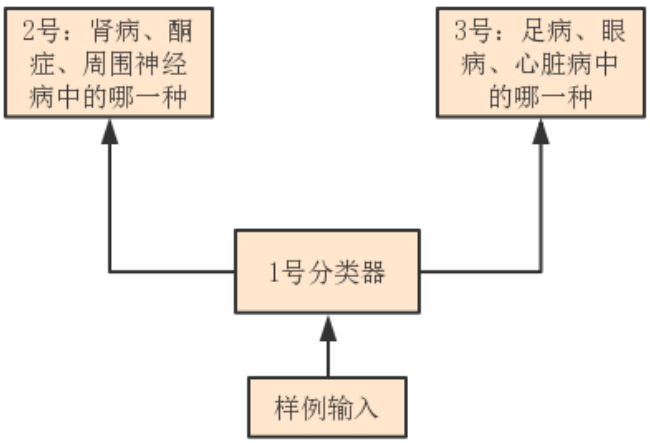
些词，得到的效果非常差，只有 40%~50%，远不如直接将所有中文词汇作为向量进行构建得到的准确率。

c. 此外，我们在测试中发现，肾病、酮症和周围神经病这三种病症是容易被分类成另外两类的的，心脏病、足病和眼病常常不容易区分。如图所示：

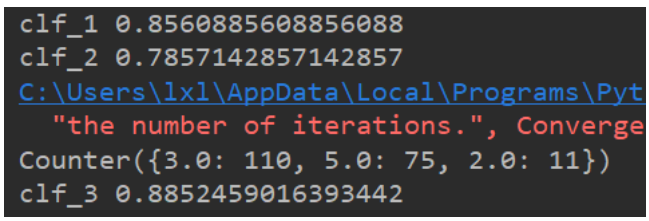


可以看到肾病通常被错分类为酮症和周围神经病，酮症经常被错分类为肾病和周围神经病，心脏病常被错分类为肾病和周围神经病，眼病常被错分类为周围神经病，周围神经病常被错分类为肾病和酮症，足病常被错分类为周围神经病。

因此我们考虑构建 3 个分类器：如图所示，1 号分类器用于判断当前样本是否为肾病、酮症或者周围神经病的一种，2 号分类器用于判断当前样本是肾病、酮症和周围神经病中的哪一种，3 号分类器用于判断当前样本是眼病、足病和心脏病中的哪一种。



测试结果如下，可以看到测试效果没有明显提升，也没有明显下降。当前限制分类器做出正确分类的是肾病、酮症和周围神经病之间的分类。



3.3.2 Q2

测试结果如下：

```
方差可解释性 0.5799569059187349
绝对误差 2.4322082459572223
均方误差 21.83339108338772
```

3.4 算法分析

3.4.1 Q1

我们讨论了多种不同的测试，最后得到的结果却是相近的。我们认为有多种原因：其一，数据集比较小，而类别又比较多，从而将数据分割成训练集和验证集时的随机性比较强，对测试结果影响较大；其二，样例不均衡，最多的肾病有 400 多个样本，最少的心脏病只有 10 个样本左右；其三，文本较长。这些原因综合起来，使得我们的改变的条件影响相对有限。

4．结论

本次实验我们主要讨论了文本多分类问题。可以看到在这类问题上，我们的模型效果是比较差的。

针对数据集较小的问题，我们考虑过一种拓展数据集的方法。将标签相同的文件混合，组成新的文件。最开始测试时其准确率非常高，但是最后发现是因为把测试集也拿去做文件混合了，这就导致测试集被污染，使得测试结果不可靠。由于时间限制，我们没有做进一步的测试，我们考虑对这种数据生成方式做进一步讨论。

此外，本次实验使我认识到理论分析的重要性，报告 3.3.1 中讨论了多种方法，没有一种是有效的，更奇怪的是，简单的线性 SVM 分类器却能轻松达到 70% 的准确率，却分析不出原因，实在是很令人遗憾的。