COMP9414: 人工智能作业2： 评级预测

交付日期。第9周，7月27日，星期三，晚上11:59。

价值: 25

这项任务的灵感来自于一个典型的现实生活场景。想象一下，你被一家大型电子商务零售商聘用为数据科学家。你的工作是分析客户评论，以确定是否可以预测新产品的评级，以便在你的网站上进行推广。

在这项任务中，你将得到一个亚马逊客户评论的集合。每条评论由短文（几句话）和五个*评级*之一组成：一个从1到5的数字。 你需要使用各种特征和设置来评估各种有监督的机器学习方法，以确定哪些方法在这个领域的评级预测中效果最好（这些特征随后可以用来根据用户的兴趣向他们推荐商品）。

这项任务有两个部分：通过*编程*产生一个用于评级预测的模型集合，以及一份评估模型有效性的*报告*。编程部分涉及开发Python代码，对评论进行数据预处理，并使用NLP和机器学习工具包进行方法实验。报告涉及使用各种指标对模型进行评估和比较。

你将使用NLTK工具包进行基本的语言预处理，并使用scikit-learn进行特征构建和评估机器学习模型。你将得到一个如何使用NLTK和scikit-learn来定义机器学习方法的例子（example_ py），以及一个如何在图表中绘制度量的例子（plot_ py）。

数据和方法

*训练*数据集是一个包含若干评论的.tsv（tab分离值）文件，每行有一条评论，评论中的换行符被删除。tsv文件的每一行都有三个字：实例编号，文本和评级（从1到5的数字）。测试数据集是一个.tsv文件，其格式与训练数据集相同，*只是你的代码应该忽略评级字段*。训练和测试数据集可以从提供的文件Review_tsv中提取（见下文）。对于模型的评估，我们将使用这个文件的一个80-20分割。

对于所有的模型，认为评论是一个词的集合，其中一个词是由至少两个字母、数字或符号/（斜线）、-（连字符）、$或%组成的字符串，以空格为界，*在*用空格替换两个连续的连字符--、倾斜符号~和任何省略号（三个或多个点...）后，再去除标签（<和>之间的最小文本跨度，包括）和所有其他字符。两个字符是scikit-learn中CountVectorizer的默认最小字长。请注意，删除 "垃圾 "字符可能会产生更长的单词，而这些单词之前是由这些字符分开的，例如在删除标签、逗号和句号之后。

使用讲座中讨论的监督学习方法。决策树（DT）、伯努利天真贝叶斯（BNB）和多项式天真贝叶斯（MNB）。不要对这些方法进行编码：而是使用scikit-learn的实现。阅读scikit-learn关于决策树[1] 和奈何贝叶斯的文档，[2] ，以及描述这些方法参数的链接页面。

[1] https://scikit-learn.org/stable/modules/tree.html [2] https://scikit-learn.org/stable/modules/naive_bayes.html

看看example.py，看看如何使用CountVectorizer并训练和测试机器学习算法，包括如何为开发的模型生成指标，以及plot.py，看看如何将这些指标绘制在图表上以纳入你的报告中。

任务的*编程*部分是在Python程序中生成DT、BNB和MNB模型以及你自己的评级预测模型，这些程序可以从命令行中调用，对从正确格式的.tsv文件中读取的评论进行训练和分类。任务的*报告*部分是使用各种参数、预处理工具和情景来分析这些模型。

编程

你将提交四个Python程序：(i) DT分类器。py，(ii) BNB分类器。py，(iii) MNB分类器。py和(iv) my分类器。py。其中前三个是标准模型，定义如下。最后一个是你在对数据进行实验后开发的模型。使用给定的包含2500条标记的评论的数据集review.tsv来开发和测试这些模型，如下面所述。

这些程序，当从命令行调用时，将从标准输入（而不是硬编码的文件reviews。tsv）读取，并应打印到标准输出（而不是硬编码的文件output。txt），测试集中每个评论的分类器在训练集上训练时产生的实例编号和评级（每行一个，实例编号和评级之间有空格 - 一个从1到5的数字），其中训练集是文件的前80%，测试集是其余20%（文件长度将总是被5除以）。比如说。

```
python3 DT classifier。py < reviews。tsv > output。txt
```

txt是测试集（文件的最后20%）中每个评论的实例编号和评级，由在训练集（文件的前80%）上训练的决策树分类器决定。

当读入数据集时，确保你的代码能读到*所有的*实例（有些Python阅读器使用 "excel "格式，它使用双引号作为分隔符）。

标准型号

你将开发三个标准模型。对于所有的模型，确保scikit-learn不将文本转换为小写字母。对于决策树，使用scikit-learn的决策树方法，准则设置为 "熵"，随机状态=0。Scikit-learn的决策树方法没有实现修剪，相反，你应该确保当一个节点覆盖的训练集少于1%时，决策树的构建就会停止。决策树容易出现碎片化，所以为了避免过度拟合和减少计算时间，对于决策树模型，只使用词汇中最频繁的1000个词作为特征，如上所述，在预处理后去除 "垃圾 "字符。在DT分类器中编写代码来训练和测试决策树模型。py。

对于BNB和MNB，使用scikit-learn的实现，但使用词汇表中*所有*的词作为特征。编写两个Pythons程序来训练和测试Naive Bayes模型，一个是BNB模型，一个是MNB模型，在BNB classifier。py和MNB classifier。py中。

您的模型

通过改变学习者的输入特征的数量和类型、学习者的参数以及训练/测试集的分割，或者使用scikit-learn的另一种方法，开发你的最佳评级预测模型。提交一个程序，即my classifier.py，它以与标准模型相同的方式训练和测试一个模型。进行新的实验来分析你的模型，并在报告中提出结果，证明你选择这个模型的理由。

报告

在报告中，你将首先评估标准模型，然后提出你自己的模型。对于下面的问题1-4，考虑两种情况，在情况1中，有5个类别对应于评级，但在情况2中，有3个类别，评级被合并为 "情感"。

(1) 类是1到5的评级值（1、2、3、4和5），和

(2) 类是一种 "情绪"，其中1、2或3是*负面的*，4是*中性的*，5是*正面的*。对于评估所有的模型，报告在数据集的前2000个实例（"训练集"）上的训练结果和在其余500个实例（"测试集"）上的测试结果。

使用scikit-learn的指标（微观和宏观的准确率、精确度、召回率和F1）和分类报告。用Python图表显示结果（不要对sklearn分类报告进行截图），并对下面的每个问题写一个*简短的*回答。对每个问题的回答应该是自成一体的。你的报告最多应该有10页。不要包括附录。

1. (3分)建立决策树模型进行训练和测试：(a)采用1%的停止准则（标准模型），(b)不采用1%的停止准则。

(i) 显示场景1的测试集上的所有指标，比较两个模型（a）和（b），并解释任何相似和不同之处。

(ii) 显示情景2的测试集上的所有指标，比较两个模型（a）和（b），并解释任何相似和不同之处。

(iii) 解释方案1和方案2之间结果的任何差异。

2. (3分)从训练集中开发BNB和MNB模型，使用：(a)整个词汇(标准模型)，和(b)使用sklearn的CountVectorizer定义的词汇中最频繁的1000个词，在通过去除 "垃圾 "字符进行预处理后，。

(i) 显示场景1的测试集上的所有指标，比较相应的模型（a）和（b），并解释任何相似性和差异。

(ii) 显示情景2的测试集上的所有指标，比较相应的模型（a）和（b），并解释任何相似和不同之处。

(iii) 解释方案1和方案2之间结果的任何差异。

3. (3分)通过比较三个标准模型的预处理效果来评估。(a)只进行上述预处理（标准模型），和(b)另外使用NLTK进行波特干化，然后使用sklearn的CountVectorizer进行英语停止词的去除。

(i) 显示场景1的测试集上的所有指标，比较相应的模型（a）和（b），并解释任何相似性和差异。

(ii) 显示情景2的测试集上的所有指标，比较相应的模型（a）和（b），并解释任何相似和不同之处。

(iii) 解释方案1和方案2之间结果的任何差异。

4. (3分) 通过比较三个标准模型，评估将所有字母转换为小写的效果。(a)不转换为小写字母，和(b)所有输入文本都转换为小写字母。

(i) 显示场景1的测试集上的所有指标，比较相应的模型（a）和（b），并解释任何相似性和差异。

(ii) 显示情景2的测试集上的所有指标，比较相应的模型（a）和（b），并解释任何相似和不同之处。

(iii) 解释方案1和方案2之间结果的任何差异。

5. (5分) 描述你选择的 "最佳 "评级预测方法。给出你在2000条评论的训练集上训练并在500条评论的测试集上测试的方法的新实验结果。解释这个实验评估如何证明你对模型的选择是正确的，包括设置和参数，与一系列的替代方案相比。提供新的实验和理由：不要只参考以前的答案。

提交

- 确保你的名字和身份证出现在报告的每一页上。
- 使用诸如以下的命令提交你的所有文件（这包括Python代码和报告）。

  ```
  give cs9414 ass2 DT*.py BNB*.py MNB*.py my_classifier.py report.pdf
  ```

- 你提交的材料应包括。
  - 你的指定模型和你的模型的.py文件，加上任何.py "帮助 "文件
  - 一个包含你的报告的.pdf文件
- 当你的文件提交后，将进行一次测试，以确保你的一个Python文件在CSE机器上运行（注意打印出来的任何错误信息）。
- 当在CSE机器上运行你的代码时。
  - 设置SKLEARN_SITE_JOBLIB=TRUE以避免警告信息
  - 请不要在你的代码中下载NLTK。CSE的机器已经安装了NLTK
- 使用命令检查是否收到你提交的材料。

  ```
  9414 classrun -check ass2
  ```

评估

本作业的分数分配如下。

- 编程（自动评分）。8分
- 报告。17分

迟到的惩罚。在到期日之后的5个日历日内，每迟一天或部分时间，你的分数就会减少1.25分，之后分数为0。

评估标准

- 正确性。在标准输入测试中进行评估，使用诸如以下的调用。

  ```
  python3 DT_classifier.py < reviews.tsv > output.txt
  ```

  每个这样的测试将给出一个文件，其中可以包含任何数量的评论（每行一个），格式正确。数据集可以有任何名字，而不仅仅是reviews.tsv，所以从标准输入读取文件。输出应该是一连串的行（测试集中的每条评论都有一行，即输入文件的最后20%），给出实例编号和预测的评分，用空格隔开，每行没有多余的空格，也没有多余的换行符在最后一个预测后的任何换行符之后。三个标准模型的正确性各占2分。

  对于你自己的方法，如果你的方法在包括未见过的例子的评论数据集上是正确的，则分配2分。

- 报告。根据实验分析的正确性和彻底性、解释的清晰性和简洁性以及报告质量进行评估。

  如上所述，第1-4项有12分，第5项有5分。在这5分中，1分用于描述你的模型，2分用于对你的模型进行新的实验分析，2分用于用新的分析来证明你的模型。一般来说，如果描述的质量不好，最多可以得到50%的分数。

剽窃行为

请记住，为这项作业提交的所有作品必须是你自己的作品，不允许分享或复制解决方案（代码或报告）。你可以使用互联网上的代码，但必须在你的程序中适当注明来源。不要使用github等网站上的公共代码库，如果你使用代码库，请确保你的代码库是私有的。所有提交的作业（包括代码和报告）将通过抄袭检测软件来检测与其他提交的作业的相似性，包括过去几年的。在课程期间和结束后，都不要与任何人分享你的代码或报告。你应该仔细阅读新南威尔士大学关于学术诚信和抄袭的政策（从课程网页上链接），特别要注意的是，*串通*（共同完成一项作业，或分享部分作业解决方案）是一种抄袭形式。

不要使用任何来自合同作弊 "学院 "的代码或 "辅导 "服务。这是严重的不当行为，将受到严厉的处罚，甚至自动挂科，并被扣0分。