Task 6: Hygiene Prediction

Help Center

Overview

Sometimes we make decisions beyond the rating of a restaurant. For example, if a restaurant has a high rating but it often fails to pass hygiene inspections, then this information can dissuade many people to eat there. Using this hygiene information could lead to a more informative system; however, it is often the case where we don't have such information for all the restaurants, and we are left to make predictions based on the small sample of data points.

In this task, you are going to predict whether a set of restaurants will pass the public health inspection tests given the corresponding Yelp text reviews along with some additional information such as the locations and cuisines offered in these restaurants. Making a prediction about an unobserved attribute using data mining techniques represents a wide range of important applications of data mining. Through working on this task, you will gain direct experience with such an application. Due to the flexibility of using as many indicators for prediction as possible, this would also give you an opportunity to potentially combine many different algorithms you have learned from the courses in the Data Mining Specialization to solve a real world problem and experiment with different methods to understand what's the most effective way of solving the problem.

You will be competing with your classmates to achieve the best prediction performance, and a **Capstone Competition Leaderboard** has been set up for this purpose.

About the Dataset

You should first **download the dateset**. The dataset is composed of a training subset containing 546 restaurants used for training your classifier, in addition to a testing subset of 12753 restaurants used for evaluating the performance of the classifier. In the training subset, you will be provided with a binary label for each restaurant, which indicates whether the restaurant has passed the latest public health inspection test or not. Whereas for the testing subset, you will not have access to any labels. The dataset is spread across 3 files such that the first 546 lines in each file correspond to the training subset and the rest are part of the testing subset. Below is a description of each file:

- **hygiene.dat**: Each line contains the concatenated text reviews of one restaurant.
- **hygiene.dat.labels**: For the first 546 lines, a binary label (0 or 1) is used where a 0 indicates that the restaurant has passed the latest public health inspection test, while a 1 means that the restaurant has failed the test. The rest of the lines have "[None]" in their label field implying that they are part of the testing subset.
- **hygiene.dat.additional**: It is a CSV (Comma-Separated Values) file where the first value is a list containing the cuisines offered, the second value is the zip code, which gives an idea about the location, the third is the number of reviews, and the fourth is the average rating, which can vary between 0 and 5 (5 being the best).

Note that the training subset is perfectly balanced, i.e., the number of restaurants with label 1 is equal to those with label 0. However, the testing subset is imbalanced where the majority of restaurants have a label of 0 (meaning that they have passed the inspection). Due to this imbalance, the classification accuracy may not be a suitable measure for evaluating the performance of classifiers. Therefore, we will use the F1 measure, which is the harmonic mean of precision and recall, to rank the submissions in the leaderboard. The F1 measure will be based on the macro-averages of precision and recall (macro-averaging is used here to ensure that the two classes are given equal weight as we do not want class 0 to dominate the measure).

Instructions

As you have probably noticed, this task is similar to Task 4 in the programming assignment of the Text Mining and Analytics course; however, there are three major differences:

1. The training data is perfectly balanced, whereas the testing data is skewed, which creates a new challenge since the training and testing data have different distributions.
2. The main performance metric is the F1 score as opposed to the classification accuracy that was used in the Text Mining course. This means that a good classifier is expected to perform well on both classes.
3. Extra non-textual features such as the cuisines, locations, and average rating are given. This might help in further improving the prediction performance and provide an opportunity to experiment with many more strategies for solving the problem.

You are required to participate in the competition by making at least one successful submission in addition to submitting a report. Make sure to check the Evaluation section for details on the grading of the competition and the report as it contains specific requirements to get full grades.

You are free to use whatever toolkit or programming language you prefer. See the **Getting Help** section below for some useful toolkits that are good candidates to complete this task. You should train a classifier over the 546 training instances and then submit the binary predictions for the remaining 12753 instances, each on a separate line. The first line should contain the nickname that you want to have on the leaderboard, i.e., the output file should have the following format:

Nickname
Label1
Label2
.
.
.
Label12573

Submission

For Task 6, you will need to submit two different parts.

1. **Output file.** After creating your output file, you should upload it to the **Task 6 File Submission** on the Assignments page. Make sure that your submission got accepted by clicking on the feedback button next to the task's name. In case your nickname is already in use, the submission will be

rejected and you will be asked in the feedback to change it. After you submit successfully, you can check the Leaderboard for your position. The operation of the Leaderboard is governed by the following rules:

- It will accept a maximum of 30 submissions per contestant.
- It will rank all submissions in descending order of the F1 score.
- In case you make multiple submissions, it will only show the highest F1 score you have achieved so far.
- In the event of ties, the contestant who submitted earlier will have the higher rank.

Submit Task 6 File

2. **Report.** We suggest that it is 1-3 pages long. The report will be peer-graded. Your report will need to include the following elements:
   - A brief description and comparison of all the methods you tried. By "methods" we are referring to the text processing techniques, feature representation and selection, and learning algorithms you experimented with. Try to explain why some methods are performing better than others, and include a failure analysis (i.e., looking at particular cases where prediction is incorrect to understand where you might be able to further improve a method).
   - Details about the method that gave the highest F1 score.

   Submit Task 6 Report

Evaluation

For this task, the file submission part of the task will be graded automatically, and the report part will be peer-graded. Task 6 is graded out of 30 points total, split equally between the output file and report. See the **Task 6 Rubric** page for a point breakdown.

The evaluation period for the report submissions will begin immediately after the submission deadline. You must evaluate **five** of your peers' report submissions or your own submission score will be penalized by 20%.

Evaluate Task 6 Report

Getting Help

You can discuss Task 6 with your peers in the Task 6 Discussion forum. We have provided tools in various languages to help you accomplish tasks on the Data Set and Toolkit Acquisition page. You may use these tools or are free to experiment with other ways of accomplishing the task. For information specific to this task, see below.

Useful Resources for Task 6

Some useful toolkits that are good candidates to complete this task:

- **MeTA:** A very efficient C++ toolkit that allows you to perform text preprocessing (e.g., stemming, stopword removal, etc.) and classification. You can use exactly the same code that was used in the Text Mining and Analytics course (i.e., competition.cpp) without any modification. However, to

make use of the additional features in hygiene.dat.additional you should write your own code. Note: Version 1.3.6 of MeTA (which is included in the VM image) has a bug which swaps the values of precision and recall shown in the confusion matrix. As long as you are aware of this bug you should be fine; however, if you prefer, you can upgrade MeTA to v1.3.8, which includes a patch for this bug.

- **Scikit-Learn:** A well-known Python toolkit with a plethora of text processing and learning algorithms that can allow you to complete the task using a relatively short code.
- **WEKA:** A Java-based toolkit that is widely used in the literature and has a good number of learning algorithms. Note that this toolkit can be opened with a graphical interface (so you might be able to complete the task with very minimal code!).

Created Wed 2 Sep 2015 9:22 PM CDT

Last Modified Sun 27 Sep 2015 10:11 AM CDT