Course Project - Task I

Machine Learning: 10-315

Fall 2019

Due dates:
Predictions Oct 30, 11:59 pm
Slides Nov 3, 11:59 pm
In-class presentation Nov 4

1 Course Project Guidelines and Dataset description

Your class project is an opportunity to explore a real machine learning problem in the context of a real-world data set as well as practice presenting your work. You will be working with a data-set of 50,000 movie reviews (25k train, 25k test). Each review comes with an associated binary sentiment polarity labels indicating whether it is a positive or a negative review. The overall distribution of labels in the training set and test set are balanced (12.5k positive and 12.5k negative).

`TASK 1:` In this project your goal is to predict the sentiment of the movie review. Currently, you are also provided with a test set of 25,000 test movie reviews (without labels). You will need to predict the labels for each test example and submit the results.An autograder will compute your classification accuracy on the test set. Additionally, there will be a leaderboard where you can compare your performance to other students in the class.

Note: you can use the leaderboard to change your algorithm and improve your performance. However, to avoid overfitting to the leaderboard, we will use a random subset of the test set for leaderboard results, and the other subset for your final score. Hence, gaming the leaderboard does not guarantee a good final score.

`TASK 2:` Closer to the submission deadline, we will release another held out movie review dataset (labels-only) that come from an entirely different dataset. This evaluates how well your method *transfers* to movie-reviews from another datasets. Hence, you should think of strategies that would work in general, and not exploit statistics of the current dataset. You will be asked to submit your predictions on the transfer dataset as well.

Your final test score will be a combination of the performance on these two tasks.

2 Project workflow

Your workflow during the project will typically be

- Preprocess the data as to have each training instance with its corresponding positive/negative label.
- Preprocess each text file that corresponds to each movie review. Here you are creating your feature-set.
- Choosing a set of relevant article features. You are free to choose the number of features for the problem.
- Applying one or more basic machine learning approaches to establish a performance baseline.

- Going beyond your baseline approach develop and study more sophisticated and hopefully more successful approaches.

3 Remarks and Grading Criterion

1. **You will be working on these projects in groups of 3.**
2. You may look-up different ways and approaches to the problem, but you may not use any pre-existing code. We will use a cheat-checker that compares your solution against code found online.
3. Your final score will be a combination of the performance of your method on a test set from the current dataset (`Task 1`), as well as your performance on a different dataset (`Task 2`). The exact weightage will be released later.
4. We will be releasing baselines that will determine your score in the following format: (for example) if your real test score is above 55%, you get at least 65% grade.

4 Task I & II Deliverables [Total 50 points]

You are required to design and implement a machine learning model that tackles these task. You will then turn in two main components to reflect your work:

1. The classification accuracies of your model on both test sets.
2. An in-class presentation of your methods and results.

4.1 Classifications [35 points]

4.1.1 Data: The provided dataset (`task1_data.zip`) can be found at the Resources section of the 10-315 Piazza webpage: here

There is a top-level directory that contains 2 folders `[train/]` and `[test/]` for the corresponding training and testing sets.

1. The `[train]` folder contains `[train/positive/, train/negative/]` directories for the reviews with positive and negative labels respectively. Reviews are stored in text files named following the convention `[{id}.txt]`, where $1 \leq \{\texttt{id}\} \leq 12500$, inside each `[train/positive/, train/negative/]`.
2. The `[test/]` dataset contains movie reviews without any labels. This contains reviews are stored in text files named following the convention `[{id}.txt]`, where $1 \leq \{\texttt{id}\} \leq 25000$ for `test/`.

4.1.2 Output format and code submission to Gradescope

You will submit a zip file contain the following items –

1. < `task1_predictions.txt` > file: this should be a file containing 25,000 lines, each corresponding to a test example. The ith line should contain a single number - 0 (for a negative prediction) or 1 (for a positive prediction, corresponding to the test example `{i}.txt`.
2. < `code` > folder: this folder should contain all your code that you used to generate the predictions file.

An example submission (`task1_example_submission.zip`) is available in the Piazza resources section here. Note that the predictions in this example `task1_predictions.txt` are *randomly generated* so is not useful in any way.

Please submit a zip file containing both these items to Gradescope: **here**.

4.2 Presentation (time to show your work and effort!) [15 points]

You will put together a **5-minute and 5-slides** presentation that describes your work to an audience. You will be asked to submit the slides and present them in class. You will be evaluated based on your effort and how good you are at conveying your results. All members of the team must present and the slides should include the following:

1. **Title, Student Names, Andrew IDs**
2. **Methods**
    - Your pre-processing steps.
    - Any features that you chose and how you choose them.
    - The ML algorithm(s) that you tried.
3. **Performance:**
    - The accuracies of the algorithm(s) you tried.
    - Any supporting results to demonstrate why your method(s) worked/did not work as well.