PR1: Movie Review Classification

Published Date:
Sep. 26, 2017, 9:00 a.m.

Deadline Date:
Oct. 10, 2017, 9:00 a.m.

Description:

****
This is an individual assignment.
****

Overview and Assignment Goals:

The objective of this assignment are the following:

- Implement the Nearest Neighbor Classification algorithm.
- Handle text data (IMDB reviews of movies).
  - Design and engineer features from text data.
- Choose the best model, i.e., parameters of a nearest neighbor classifier, features and similarity functions.

Detailed Description:

A practical application in e-commerce applications is to infer sentiment (or polarity) from free form review text submitted for a range of products.

For this assignment, you have to implement a k-Nearest Neighbor Classifier to predict the sentiment for 25000 movie reviews provided in the test file (test.dat). *Positive sentiment* is represented by a review rating of +1 and *negative sentiment* is represented by a review rating of -1. In test.dat you are only provided the reviews but no ground truth rating. These data will be used for comparing your predictions.

Training data consists of 25000 reviews as well, provided in the file train.dat. Each row begins with the sentiment score followed by the text associated with that rating. Note that the text may contain HTML artifacts and other text or numbers not associated with the review text. Your parser/tokenizer should be able to handle this type of unrelated content.

For evaluation purposes (Leaderboard Ranking) we will use the Accuracy metric comparing the predictions submitted by you on the test set with the ground truth. Some things to note:

- Some of your classmates may choose not to see the leaderboard status prior to the submission deadline. Please do not share leaderboard status information with others.

- The public leaderboard shows results for 50% of randomly chosen test instances only. This is a standard practice in data mining challenges to avoid gaming of the system. The private leaderboard will be released after the submission deadline, based on all the entries in the test set.
- In a given day (00:00:00 to 23:59:59), you are allowed to submit a prediction file only 5 times.
- The final ranking will always be based on the last submission, not your best submission. Carefully decide what your last submission should be.

format.dat shows an example file containing 25000 rows alternating with +1 and -1. Your test.dat should be similar to format.dat with the same number of rows (25000), but containing the sentiment score generated by your developed model.

Rules:

- This is an individual assignment. Discussion of broad level strategies are allowed but any copying of prediction files and source codes will result in an honor code violation. This includes reusing code posted on the Web by others.
- Feel free to use the programming language of your choice for this assignment.
- While you can use libraries and templates for dealing with text data you should implement your own nearest neighbor classifier.

Deliverables:

- **Valid submissions to the Leader Board website:** TBA
- **Canvas Submission of source code and report:**
  - Create a folder called pr1_SJSU-ID.
  - Create a subfolder called *src* and another called *report*.
  - Put all the source code in the *src* subfolder.
  - Place a 2-page, single-spaced report in the *report* subfolder. It should describe the steps you followed for developing the classifier you used to predict the movie review sentiments. Be sure to include the following in the report:
    1. Name and SJSU ID.
    2. Rank & Accuracy score for your submission (at the time of writing the report). If choosing blind submission, enter 'N/A'.
    3. Your approach.
    4. Your methodology for choosing the approach and associated parameters.
  - Archive your parent folder (.zip or tar.gz) and submit via Canvas for PR1.

Grading:

Grading for the Assignment will be split on your implementation (70%), report (20%) and ranking submissions (10%). Extra credit (1% of final grade) will be awarded to the top-3 performing algorithms and to the submission with the most interesting solution (to be judged by Prof. Anastasiu). Note that extra credit throughout the semester will be tallied outside of Canvas and will be added to the final grade at the end of the semester.

Files you will find on Canvas:

- *Train Data:* train.dat
- *Test Data:* test.dat
- *Format File:* format.dat