

Part1: Instructions for Predicting Review Stars Using Open AI.

I'm writing to provide you with instructions on how to use Open AI to predict star ratings for review text. Please follow the steps outlined below:

- **Prepare the Dataset:** Please ensure you have a dataset named “sample_new.csv”, containing review text in the “boss_program” folder.
The review text should be consolidated into a single column. If necessary, merge all relevant columns containing review text into one column named “reviewtext”.
Ensure there are no spaces before the 'reviewtext' heading. This structure must be consistent for Part 2 as well.
- **Sign Up for Open AI Premium:** To access Open AI’s premium features, you'll need a subscription to the premium version of Open AI. Please visit the website (<https://www.openai.com>) to subscribe to the premium plan.
- **Generate Open AI API Token:** After subscribing to Open AI Premium, generate an API token from the Open AI dashboard. Once you have the API token, add it to a .env file using the following format:

`OPENAI_API_KEY=Your_API-Token-Here`
- **Run “make_prediction_openai.py”:** After placing the “sample_new.csv” dataset in the same directory as the “make_prediction_openai.py” file, run .py file.

The script will use the Open AI API to predict star ratings for the review text in using advanced natural language processing techniques. This method has a very high accuracy rate, it is almost 100%. However, it is not free.

Part2: Subject: Instructions for Predicting Programmer Quality and Stars Using Cost-Efficient Models

I wanted to share with you an innovative approach to predicting programmer quality and stars using a cost-efficient mixed model. This model leverages both Open AI's advanced natural language processing capabilities and a Naive Bayes classifier to achieve high accuracy at a lower cost.

1- Mixed model:

Here's how you can utilize this model effectively:

- **Prepare the Data:** Ensure it contains the necessary structure following the format discussed earlier.
- **Run the Prediction Model:** Execute the “predict_mix.py” script in your Python environment.
This script will automatically split the dataset:

- Randomly select 20% of the dataset to train using Open AI's model and use the remaining 80% of the dataset to train a Naive Bayes classifier through a pickle file for predicting stars based on review text as discussed before.
- You will have 2 separate csv files, one for the 20% of the dataset using Open AI, second for 80% of the dataset using Naive Bayes model.

Conclusion: Predict with Cost-Efficiency and Accuracy:

- Use the mixed trained models will cause to process predictions faster.
- This mixed model approach offers cost efficiency without compromising on accuracy, achieving approximately 95% accuracy based on our testing.

By following these steps, we can leverage cutting-edge technology from Open AI alongside traditional machine learning methods to make informed predictions about programmer reviews and star ratings. This approach ensures that we maximize accuracy while optimizing costs, making it a valuable addition to our workflow.

- 2- **Naive Bayes classifier model alone:** The developed Naive Bayes model achieves an impressive accuracy score (90%) in predicting programmer quality based on provided data from “sample_new.csv”. To use the model, ensure the dataset is in the same directory as the prediction script (predict_naive.py). Running the script will process the dataset using the pre-trained Naive Bayes model along with a saved pickle file and generate predictions in a csv file. The results will providing valuable insights for decision-making processes. This approach offers a cost-effective solution for assessing programmer quality and optimizing operations.

Part3: Instructions for Predicting Programmer Quality based on the profile photos.

I'm writing to provide you with instructions on how to use the trained models to predict the quality of programmers based on their profile pictures.

Step 1: Preparing Sample Profile Pictures

Before proceeding with the prediction process, please ensure that you have placed the sample profile pictures in JPEG format in the following directory:

“/Users/#3boss_program/part_3/sample_new_data/sample_profile_picture”

Along with a csv file named “sample_new.csv” in the “boss_program “folder which has two columns: first :programmer_id, second: profile_picture with *.jpg format.

Step 2: Running “make_prediction_cnn.py”

- Once the sample profile pictures are ready, please run the make_prediction_cnn.py script.
- This script will use our trained CNN model through a pickle file to predict the category (building:0, dog:1, face:2) for each profile picture.

- The predictions will be saved in a CSV file named “sample_new.csv”, which has 3 columns; programmer_id, profile_picture(*.jpg), and prediction_photo(building or dog, or face) respectively.

Step 3: Using “predict_naive.py”

- Next, you just need to run “predict_naive.py”.
- This script will utilize another trained model and pickle file to predict the probability of stars based on the predicted photo categories.
- The predicted probabilities will be calculated and presented as the quality assessment for each programmer. (bad:1, average:2, good:3)
- After running this file, you will have a complete csv file named “sample_new_with_stars.csv”. It has 7 columns named programmer_id, prediction_photo number, prediction_photo name, prediction of stars, prediction_prob_1(bad), prediction_prob_2(average), prediction_prob_3(good).

Conclusion

Following these steps will allow us to effectively predict the quality of programmers based on their profile pictures.

I used CNN (Convolutional Neural Network) because they are designed for analyzing visual data like image, they are excel automatically learning features from raw pixel value.

I used Naive Bayes Model due to high accuracy score (about 80%) in this model, efficiency, simplicity, and probabilistic nature to interpret uncertainty.

Please be informed I used AI to fix my grammar mistakes in the letter, and some parts of the coding that were not in the class lecture.

Thank you so much for the great semester.

Sincerely,

Elham