Data Contest Report

Team Name: Thunderstats

Arsh Bawa MM17B001

Abhimanyu Swaroop MM17B008

Pattern Recognition and Machine Learning

CS5691

September – December 2020

Prof. Harish Guruprasad Ramaswamy

1. Problem Statement

The task was to build a ranking model for members of Biker-Interest-Group and in order to rank bike-tours in decreasing order of preference for each biker in the test dataset. The datasets provided contained information such as biker's previous interests, his/her demographic details, past tours, friend circle etc. We had to use these datasets to create features and come up with a machine learning model that ranks biker preference among tours. Mean average precision at k, where k is the rank, was used as the evaluation metric.

2. Creating the Dataset

The following steps were followed to create the final dataset:

- The two main files – train.csv and test.csv consisted of a 'timestamp' column. The timestamp was converted into datetime format and two features, namely day and month were created from this. Day had integer values ranging from 1 to 31 while month had integer values ranging from 1 to 12. Since the year was the same throughout the dataset, it wasn't used.
- The training data and test data were then merged and tour information from tours.csv was used to augment the dataset. Thus, the data now contained information about the tour organizer, tour date, city, state, country, pin code, latitude, longitude as well as the number of occurences of 100 common words in the tour description for each tour.
- The tour date was converted into datetime format to extract features for day, month and year for each tour.
- The bikers_network.csv dataset was used to extract different features to be added to the existing dataset:
    - Whether the biker corresponding to a particular row was the organizer of the tour or not. This column was binary encoded. If the biker was the tour organizer, the entry was 1 and 0 if not.
    - Whether someone from the biker's network was the organizer. The bikers' friends were given as a string with each friend separated by a space. Using this information, this column was filled with 1 if the biker's friend was the tour organizer and 0 if not.
    - The number of friends of a biker that were going for the tour. The information regarding bikers going/not going/invited/maybe going was taken from tour_convoy.csv.
    - Similarly, columns with the number of friends of a biker not going, invited and maybe going for the tour were also added.
- The time (in hours) between the tour date and the time when the biker was informed about the tour was calculated and added as a new feature to the dataset.
- The joining date of the biker was taken from bikers.csv and converted into datetime format. The day, month and year of joining were added as new columns.

- The bikers.csv file was merged with the dataset in order to add information about biker demographics. The time differences (in hours) between the date of joining and date of the biker being informed as well as the tour date were computed and added as two features.
- The column ‘dislike’ was removed from the training data. The ‘like’ column was taken as the target variable. This was done because many of the bikers had not given feedback on the tours so the assumption was that most of the bikers who do not give feedback, do not like the tour. The ‘biker_id’ and ‘tour_id’ columns were dropped from the data.
- The area, pin code, city, state and country of the biker were changed to categorical variables along with the gender, year they were born, language and location.

3. Models used

1. **Light GBM -** The light gradient boosting machine classifier was used to predict the probability of a biker liking the tour. Light gbm has an inbuilt capability to take care of missing data, so the missing data was left as it was. The probabilities obtained were arranged in descending order to obtain a ranking of tours according to each biker. This gave us the first submission file. The hyperparameter values were taken as the default values for light gbm.
   Score on private dataset = 0.69105
2. **Catboost + Light GBM Ensemble -** The catboost classifier was similarly used to predict probabilities, but catboost does not have the functionality to deal with missing data. So, for categorical variables, missing data was replaced by the mode for each column, while for numerical variables, missing data was replaced by the median for each column. The probabilities obtained via catboost and light gbm were averaged to give a ranking of tours for each biker. This gave us the second submission file. The hyperparameter values were taken as the default values for catboost as well as light gbm.
   Score on private dataset = 0.71804

4. Validation

For the purpose of validation, the training data was split in the ratio 80:20. The split was stratified on ‘biker_id’ so that we had rows for training as well as validation for every biker. Since there was no ranking data available, a different metric was to be used for the purpose of validation. The predictions carried out for the purpose of validation was binary classification – 1 if the biker likes a tour, 0 if the biker dislikes the tour. Area under the curve was used as the metric for validation. Validation accuracies for light gbm and catboost varied from 0.7-0.75.

5. Other approaches tried

The following were some other approaches tried in order to create features, but significantly lowered the accuracy of the public dataset:

- Each biker's area was used to compute their latitude and longitude coordinates using the python library geopy. These coordinates were then used to compute the distance between the biker's location and the tours, whose geographical coordinates were already known.
- There were numerous missing values in the tour location data. These were imputed using random forest classifier/regressor on some columns and predicting the other. For example, if the rows for which latitude and longitude were missing had values for city, country and pin code, these values were used as independent variables to predict latitude and longitude. The training data for the purpose of this prediction consisted of rows were all these columns had values present. The default values for the random forest hyperparameters were used.
- Each biker was added as a column to the dataset. For each row, the value for a biker was entered as 1 if the biker corresponding to that row was friends with the biker in the column and 0 if not (one-hot encoding).

6. Conclusion

All the given csv files were used to extract useful information about the bikers as well as the tours and create features that were added to the final dataset. The final set of features are quite meaningful and it seems logical that the information contained in these features are very useful in making recommendations to bikers.

After trying the boosting algorithms given above as well as bagging algorithms such as random forest classifier, it became clear that boosting was much more suitable to make predictions in this particular problem. The ensemble model of light gbm + catboost performed the best on the private dataset when compared to only light gbm giving a MAP@k of 0.71804.