GREYATOM

Hackathon

Problem Statements

By

GREYATOM

Table of Content

GREYATOM

How will this Hackathon work?

Structure of the Hackathon:-
The Hackathon is scheduled for two days. Each team can spend 30 mins with the mentor on both days to sound off ideas/approaches. Mentor will NOT give away solutions. Students will be provided with these problem statements on Friday. **Each team need to select one problem statement**

Day 1: Saturday, Mar 02, 2019

- Hackathon kick-off by the mentor
- Data Exploration Graphs, Correlation, Feature Engineering should be completed by 4.30PM

Day 2: Sunday, Mar 03, 2019

- Feature Selection, ML Models must be ready by 11 AM
- Understanding and fine-tuning ML models by 3 PM
- Rest of the time on proper code documentation, packaging repository for anybody to use and presentation
- Final Presentations - 3:00 PM - 6.00 PM

Note: You need to hand over your GitHub repo details and presentation files to the mentor before the presentation.

GREYATOM

1. IndusOS - Predict the next recharge

About IndusOS

India's #2, Indus OS is the World's First Regional Operating System empowering the next 1 Billion smartphone users in the emerging markets.Indus OS is addressing one of the developing world's biggest challenges - to develop technology to cater to the economic, social and regional diversity. We are the first operating system to meet the real needs of the emerging market's regional language speaking citizens through innovation, simplification and localization.

Problem Statement :

The problem statement of the exercise is to prepare a model which predicts
a) Given a user, when will he do his next balance recharge
b) The amount of the recharge

Data

Data Dictionary
1)Device-id- Unique Device ID
2)Event_arrival_timestamp- Server Timestamp when the event was recorded at server
3)Event_timestamp: Event timestamp on Mobile at the time of event. (This is time of the device at which the event is fired. If the user has changed the device clock time, same would be recorded
4) Main Prepaid balance: Main Balance of the user at the time of the event.

https://s3-ap-southeast-1.amazonaws.com/indusopenbucket/aegis000.gz

GREYATOM

2. Outlier Detection - Haptik

About Haptik
Haptik is India's first Conversational Commerce platform that is powered by both AI and real humans and we aim to redefine the way people get their everyday jobs done using chat as the underlying interface.

A combination of Artificial Intelligence, Natural Language Process & Machine Learning has helped Haptik create technology that assists their assistants work a lot faster; while the bot learns every time a new query is answered.

In short:
Haptik caters to a limited number of services. Users tend to ask queries, out of the scope of Haptik's reach. To identify these queries as outliers and handling those by gentle denial is the best practice.

Description:
Haptik provides help with a limited set of services to users. These being: Flights, Trains, Cabs, Reminders, Recharge, Nearby. Though, these categories are clearly mentioned on the platform, as the interaction medium being chat, users often ask queries which are out of scope of Haptik.

For Example:-

-What is the price of Iphone 7?
-Can you get me an appointment with physiotherapist?
-HSC results
-I want to download a hollywood movie for free. Please tell me a good website

Though, humans can most of the times cater to these queries with web-search, it is better to not to cater to such queries as it creates a different/false image of Haptik as a service/product. We call the queries which are out of scope of Haptik as, "outliers". The outlier detection also helps other modules such as domain classifier, to remove noisy data from training set.
How do we solve this problem? An important fact to notice is that most of the queries fall under the scope of Haptik and only a small portion of queries fall under the Outliers category. The outlier queries are small in number, while the types of queries has a wide range of variety. Does this hint towards unsupervised approaches? Anomaly detection? What are the commonly used algorithms for anomaly detection? Do these work good for text data? What about large feature vectors? Would clustering with little human intervention work?
Formally, given a large set of strings, the task is to identify potential candidate strings which are different from most of the queries.

Datasets:
The dataset contains a large set of strings, most belonging to either of the domains catered by Haptik, or casual/generic queries. The dataset can be found here.

Evaluation:

For Evaluation, you need to generate a label (Outlier/ Not Outlier) for every message in the dataset. Accuracy will be calculated on the results on evaluation data.

3. Improve the user experience on a fast-growing mobile platform, for hosting coding challenges - Recommendation Challenge

Your client is a fast-growing mobile platform, for hosting coding challenges. They have a unique business model, where they crowdsource problems from various creators(authors). These authors create the problem and release it on the client's platform. The users then select the challenges they want to solve. The authors make money based on the level of difficulty of their problems and how many users take up their challenge.

The client, on the other hand, makes money when the users can find challenges of their interest and continue to stay on the platform. Till date, the client has relied on its domain expertise, user interface and experience with user behavior to suggest the problems a user might be interested in. You have now been appointed as the data scientist who needs to come up with the algorithm to keep the users engaged on the platform.

The client has provided you with a history of last 10 challenges the user has solved, and you need to predict which might be the next 3 challenges the user might be interested to solve. Apply your data science skills to help the client make a big mark in their user engagements/revenue.

Dataset description:
We have three data files,

- Train.csv
-

Variable	Definition
user_sequence	Unique ID for the sequence
user_id	USer ID

challenge _sequence	Challenge sequence number(1-13)
challenge	Challenge ID

- Challenge.csv

Variable	**Definition**
Challenge_ID	Challenge_ID
programming _language	Programming language for the challenge
challenge_series _ID	Series for the given challenge
total_submissions	Total submission by all users
publish_date	Publishing date for the challenge
author_ID	Author ID
author_org_ID	Organization ID for the author
category_id	Type of challenge

Evaluation Metric

- The evaluation metric is Mean Average Precision (MAP) at K (K = 3). MAP is a well-known metric used to evaluate ranked retrieval results. E.g. Let's say for a given user, we recommended 3 challenges and only 1st and 3rd challenges are correct. So, the result would look like — 1, 0, 1
- In this case, The precision at 1 will be: 1*1/1 = 1 The precision at 2 will be: 0*1/2 The precision at 3 will be: 1*2/3 = 0.67 Average Precision will be: (1 + 0 + 0.67)/3 = 0.556. The formula is:

Dataset Download from here

Helpful Techniques:

- Use Matrix factorizations, SVD, collaborative filtering etc
- Split the dataset based on users to evaluate your results.
- Ensembling different models will show some improvements.
- Can use deep learning. Check the link attached for one way to know how to design networks.