TalkingData AdTracking Fraud Detection Challenge

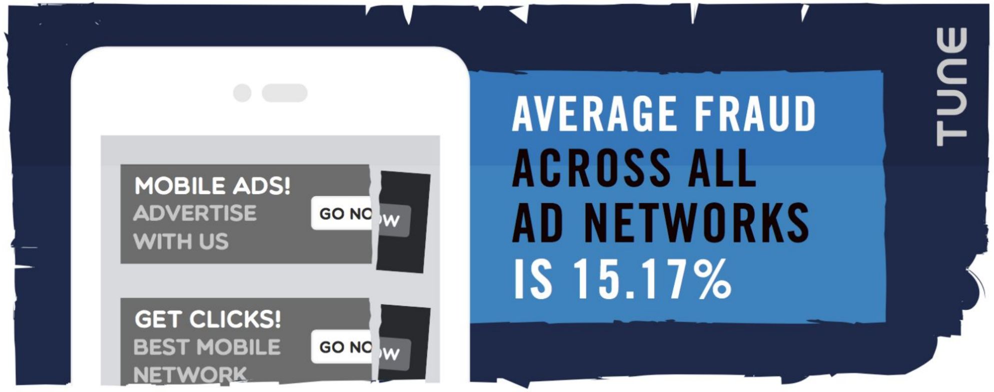
Can you detect fraudulent click traffic for mobile app ads?

TEAM NAME : hire-us

CHHAYA CHOUDHARY, IRINA VIRNIK,

VANDANA IYER

MOTIVATION



Ref: https://blog.branch.io/mobile-ad-fraud-what-24-billion-clicks-on-700-ad-networks-reveal/

GOAL

Predict a probability for the target is_attributed variable for each click_id in the test set.

is_attributed: the target that is to be predicted, indicating the app was downloaded.

Click id	is_attributed
1	0.003
2	0.001

Evaluation:

Submissions are evaluated on area under the ROC curve between the predicted probability and the observed target.

DATASET STATISTICS

DATASET NAME	NUMBER OF ROWS	SIZE(Unzipped)
Train	187,903,890	7.5 GB
Train Sample	100,000	4.1 MB
Test	18,790,469	863.3 MB
Test supplement	57,537,505	2.7 GB

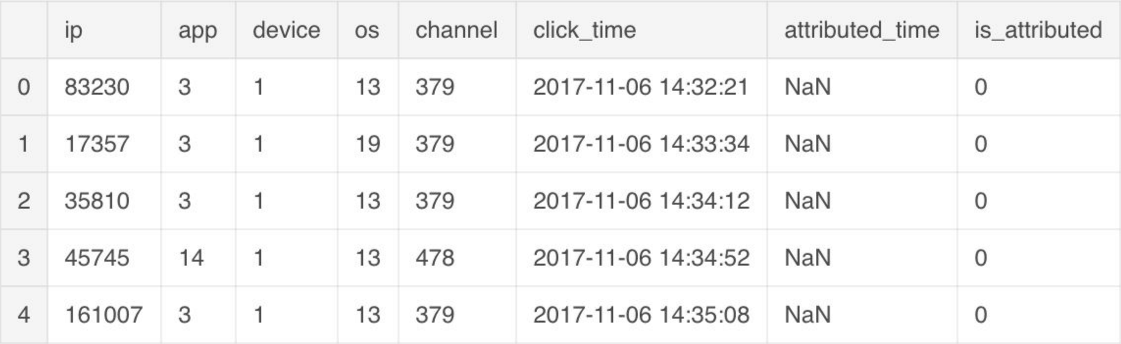
% of Negative data: 99.8%
% of Positive data: 0.2%

More complicated than we initially thought

Image Source: google.com

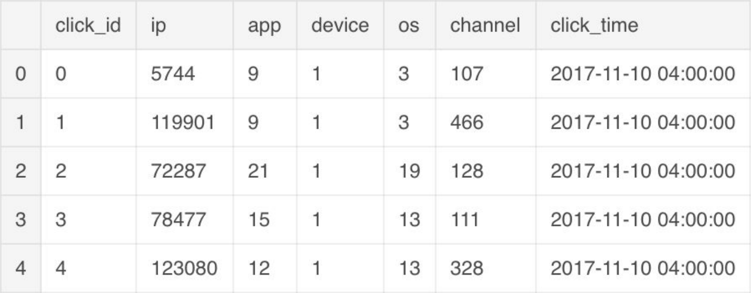
DATASET DESCRIPTION

Train Dataset



	ip	app	device	os	channel	click_time	attributed_time	is_attributed
0	83230	3	1	13	379	2017-11-06 14:32:21	NaN	0
1	17357	3	1	19	379	2017-11-06 14:33:34	NaN	0
2	35810	3	1	13	379	2017-11-06 14:34:12	NaN	0
3	45745	14	1	13	478	2017-11-06 14:34:52	NaN	0
4	161007	3	1	13	379	2017-11-06 14:35:08	NaN	0

Test Dataset



	click_id	ip	app	device	os	channel	click_time
0	0	5744	9	1	3	107	2017-11-10 04:00:00
1	1	119901	9	1	3	466	2017-11-10 04:00:00
2	2	72287	21	1	19	128	2017-11-10 04:00:00
3	3	78477	15	1	13	111	2017-11-10 04:00:00
4	4	123080	12	1	13	328	2017-11-10 04:00:00

OUR METHODOLOGY

- Exploratory Data Analysis - Time Series Analysis
- Handling large dataset - Down Sampling
- Feature Engineering - Time based, Velocity based
- Model selection and preparation
- Hyperparameter tuning

Resources utilized:

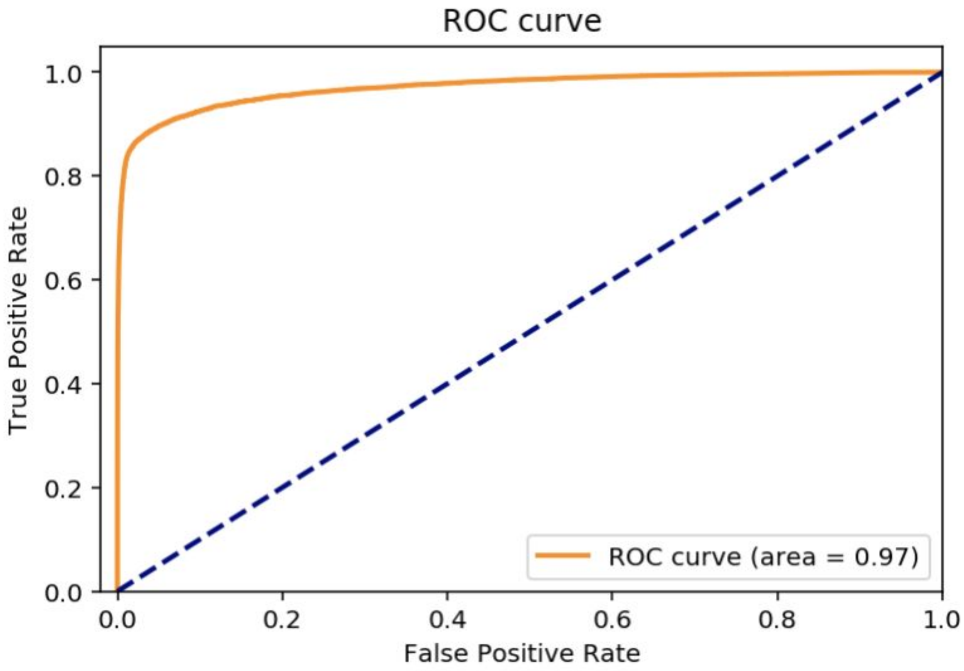
- Kaggle Kernels (12GB RAM)
- Google Colab
- Google Cloud Platform (GCP) - 8 Cores, 52GB RAM

RESULTS - APPROACH

Classifier	LB Public Score(AUC)	LB Private Score(AUC)
XGBoost with down sampling	0.96273	0.96549
XGBoost with training on consecutive 10M rows from end	0.95613	0.95558
XGBoost with full data	0.96107	0.95846
Gradient Boosting Classifier	0.95819	0.95673
Logistic Regression	0.60725	0.62578

LB: Leaderboard on Kaggle

ROC CURVE - VALIDATION SET



Used XGBoost:

- Learning rate = 0.3
- Objective function: binary logistic
- No. of iterations: 200
- Early Stopping : 20
- Eval_metric : auc

FEATURE ENGINEERING

Day of the week
Day of the year
Hour of the click time
Minute of the click time
Second of the click time
Click counts by ip
ip clicks to attribution percentage
Click counts by os
os clicks to attribution percentage
Click counts by device
Device clicks to attribution percentage

BEST WORKING ALGORITHM: XGBOOST

Advantages:

1. **Regularization** - It helps to reduce overfitting
2. **Parallel Computing** - It is enabled with parallel processing (using OpenMP); i.e., when you run xgboost, by default, it would use all the cores of your laptop/machine.
3. **High Flexibility** - XGBoost allow users to define custom optimization objectives and evaluation criteria
4. **Handling missing values** - has an in-built routine to handle missing values
5. **Tree Pruning** - XGBoost make splits upto the max_depth specified and then start pruning the tree backwards and remove splits beyond which there is no positive gain
6. **Built-in cross validation** - XGBoost allows user to run a cross-validation at each iteration of the boosting process and thus it is easy to get the exact optimum number of boosting iterations in a single run

CHALLENGES

❑ Resource and memory limitations

❑ Working with huge imbalanced data

❑ Feature engineering

❑ Dividing tasks within a team

FUTURE WORK

❑ More feature Engineering

❑ Experimentation with neural networks

❑ Deploy using Flask

❑ Experimentation with Dask

❑ Turn this into a product for ad analytics

KEY TAKEAWAYS

- ❑ Learned about handling class imbalance in Click fraud data
- ❑ Hands on - Google Cloud Platform (GCP)
- ❑ More data is not always useful
- ❑ Github:

  https://github.com/WomenInDataScience-Seattle/talking_data_fraud_detection/tree/master/hire-us

QUESTIONS?

Thank You

köszönöm
Kiitos
Asante
Shukria
Vinaka
Maake
Dhanyavadagalu
Terima Kasih
감사합니다
Dankscheen
спасибо
Kam Sah Hammida
اركش
Manana
Dankon
Matondo
Dank Je
Mauruuru
Biyan
Diolch i Chi
Taiku
Tack
Blagodaram
Ngiyabonga
Dziekuje
Chokrane
Arigato
Grazie
Juspaxar
Mochchakkeram
Gracias
Khap
Paldies
Tingki
நன்றி
Bedankt
Dakujem
धन्यवाद
cảm ơn bạn
Ua Tsaug Rau Koj
Grazas
Di Ou Mèsi
Kia Ora
Kop Khun
Gratias Tibi
Obrigado
Děkuji
Nuwun
Nirringrazzjak
ありがとう
Suksama
Hvala
Rahmat
Matur
谢谢
хвала
Welalin
Djiere Dieuf
Misaotra
Danke
Merci
Go Raibh Maith Agat
Tuke
Eskerrik Asko
Salamat
ขอบคุณ
Najis