# Lord Of the Machines: Data Science Hackathon

## Problem Statement

Analytics Vidhya sends out marketing emailers for various events such as conferences, hackathons, etc. It has provided sample of user-email interaction data from July 2017 to December 2017.

**It is required to predict the click probability of links inside a mailer for email campaigns from January 2018 to March 2018.**

## Evaluation Criteria

The evaluation criteria for the problem is AUC\_ROC score

## Approach

Aditya and Akash worked on different models and teamed up on 1st August, 2018.

### Aditya’s Approach

Multiple classification model has been create to predict the click probability of links inside mailer for email campaign.

Following derived features has been created, for training different model

1. Different users have been clustered (using kmeans clustering algorithm) based on email of different campaign received by them and each user is assigned a cluster\_id.
2. Count of email received by users
3. Count of emails sent on a given day
4. Count of emails pertaining to communication\_type
5. Day of week
6. Target encoding of user\_id based on ‘is\_click’.
7. Target encoding of user\_id based on ‘is\_open’.
8. Target encoding of Communication Type on ‘is\_click’.
9. Target encoding of Communication Type on ‘is\_open’.
10. Target encoding of cluster id on ‘is\_click’.

#### Cross Validation Approach

Campaign\_id has been used to split data into train and validation.

1. For xgboost, 2 fold had been used and test predictions of 2 models has been averaged. With cross validation and public leaderboard score, number of trees has been fixed to 70.
2. For lgbm, 5 fold has been used and test predictions of 5/(Top 4) models has been rank averaged. Mainly 3 different type of model has been created, 1 with 45 trees, 1 with 51 trees and 1 with 55 trees.

#### Hyper Parameter Tuning

Manual tuning has been performed based on public lb and cv score.

#### Akash’s Approach

Modelling

I posed it as a problem of sequence prediction where we want to find whether a user will click on an email, given his past interactions on platform. The first thing that comes to mind when we think of sequence prediction problems is RNN or more specifically LSTM.

Features

I formed sequences of users' actions (in form of clicked and opened). 4 sequences were formed:

* Clicked (0s and 1s)
* Opened (0s and 1s)
* No of sections in each email
* No of images in each email

These sequences acted as 4 features for sequential input.

Network Architecture(s)

I started with LSTM model followed by a couple of dense layers and tuned its optimisers and no of cells in the LSTM layer.

I added a CNN model followed by a GlobalMaxPool followed by a couple of dense layers and tuned its architecture.

Finally I added a CNN layer followed by a LSTM layer followed by couple of dense layers as the third model.

Prediction and Cold Start

The output of these models gave me a probability whether these users will click the next email or not. I used this probability across all emails sent to that user (I did not want to add prediction to sequence and predict again because that can cause errors to propagate further).

This allowed me to make predictions for the users for whom we have some data (previous behaviour), but in the test set, 20% of the entries were for users for whom we do not have any data (aka cold start).

To deal with Cold Start, I grouped by campaign\_id and sent\_weekday and sent\_quarter\_of\_day and filled the missing values by 90% quantile across each group.

#### Ensemble Model

All the model prediction has been rank averaged to reach the final submission