Spring 5-20-2019

AN ENSEMBLE MODEL FOR CLICK THROUGH RATE PREDICTION

Muthaiah Ramanathan
San Jose State University

AN ENSEMBLE MODEL FOR CLICK THROUGH RATE PREDICTION

A Project Report

Presented to

Dr. Mike Wu

Department of Computer Science

San José State University

In Partial Fulfilment

Of the Requirements for the Degree

Master of Science

By

Muthaiah Ramanathan

May, 2018

The Designated Project Committee Approves the Project Titled

AN ENSEMBLE MODEL FOR CLICK THROUGH RATE PREDICTION

By

Muthaiah Ramanathan

APPROVED FOR THE DEPARTMENT OF COMPUTER SCIENCE

SAN JOSE STATE UNIVERSITY

Spring 2019

Dr. Mike Wu, Department of Computer Science

Dr. Robert Chun, Department of Computer Science

Dr. Hailin Wu, Google Inc

ABSTRACT

Internet has become the most prominent and accessible way to spread the news about an event or to pitch, advertise and sell a product, globally. The success of any advertisement campaign lies in reaching the right class of target audience and eventually convert them as potential customers in the future. Search engines like the Google, Yahoo, Bing are a few of the most used ones by the businesses to market their product. Apart from this, certain websites like the www.alibaba.com that has more traffic also offer services for B2B customers to set their advertisement campaign. The look of the advertisement, the maximum bill per day, the age and gender of the audience, the bid price for the position and the size of the advertisement are some of the key factors that are available for the businesses to tune. The businesses are predominantly charged based the number of clicks that they received for their advertisement while some websites also bill them with a fixed charge per billing cycle. This creates a necessity for the advertising platforms to analyze and study these influential factors to achieve the maximum possible gain through the advertisements. Additionally, it is equally important for the businesses to customize these factors rightly to achieve the maximum clicks. This research presents a click through rate prediction system that analyzes several of the factors mentioned above to predict if an advertisement will receive a click or not with improvements over the existing systems in terms of the sampling the data, the features used, and the methodologies handled to improve the accuracy. We used the ensemble model with weighted scheme and achieved an accuracy of 0.91 on a unit scale and predicted the probability for an advertisement to receive a click form the user.

ACKNOWLEDGMENTS

I would like to express my gratitude towards my project advisor Dr. Mike Wu for guiding me through the extensive research process that became this master's project. Furthermore, I would also like to thank my committee members Dr. Robert Chun and Dr. Hailin Wu for their suggestions and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In the recent years, much of the business is made available to the users as the digital content over the internet. It has become inevitable for most of the businesses to market their products online by running promotional advertisements that can help them scale multiple folds in their business' expansion. The advertisements on Google's Search Engine Results Page, the ones on Facebook pages and Amazon's website are some of the examples. There are many kinds of advertisements that are shown to the users these days. The text-based advertisements, video-based advertisements and other interactive advertisements are a few of the them.

The click through rate (CTR) [1] is a relative measure to understand the number of users that see an advertisement to the number of people who actually end up clicking on it. A high value of the click through rate is directly proportional to the usefulness of an advertisement. Given this measure, a business can be informed of how well the keywords that they have chosen is performing in their advertisement campaign. Its recurrent in a way that if the businesses are able to have the expected CTR before they subscribe to an advertisement campaign, they can choose the keywords and other attributes for any advertisement accordingly. A business could choose multiple advertisements(ads) by varying the keywords, the bids, the banner positions and also the display size. In a very competitive and volatile market, the companies are very keen in advertising their products aggressively using the search engines and the social media and pay charges basis the number of the clicks that their ads receive [2]. From the perspective of the search engines, this is seen as a fair way of billing the customers and the click through rate acts as a good justification.

1.1 Importance

Banking upon the fact that advertising has become the key source of revenue for a lot of websites and mobile applications, it is certainly valuable to study the effectiveness of the advertisement campaigns. Thus, the prediction of the click through rate has become paramount importance and hence this study. This study will help the business owners to target their potential customers with the right advertisement structure on a quality website. On the other hand, this study will as well be useful for the search engines and the website owners in deciding the category of advertisements that are suitable to facilitate the online advertisers with the bidding rate for an ad, space and orientation of the ad based on the traffic and rank of their website [1]. Hence, this is an interesting area of research as it will help both the advertisement platform owners as well as the business providers with the intuitive insights in choosing the advertisements that are profitable in terms of the monetary conversion.

A lot of research in this filed has not been able to progress as because the data was sensitive and not made public [2]. Most of the search engines captured the data and used the insights for them to decide on the cost per click of an advertisements. At present, there are few data sets that are available from Avazu Labs and Alibaba, an ecommerce business to business (B2B) platform for the researchers.

Any improvements in predicting if an advertisement will be clicked or not will enable a lot of companies that want to try a new advertising platform to rightly plan their finance [3] for online marketing and in turn study the effectiveness of their business in the minds of the consumers as a whole.

1.2 Motivation

An improvement, although insignificant in the prediction of the click through rate could affect the stake holders' finance widely [2][3]. The whole process of the online advertisement depends on the probability of the consumer who sees an ad, clicking it or not. While, it is vital to note that a user clicking on an advertisement does not guarantee to become the converted customer, it is one of the best ways to take a safe assumption that this user could be a potential customer in the future. Using the pay per click payment system, it is more important for the ad publisher to accurately predict the probability that a user is clicking on an advertisement, so they can hold a lot of advertisers under their realm. As this is the only key factor in determining the final revenue, it becomes a good area of research for the machine learning enthusiasts to work on.

Nevertheless, the problem can be seen from the perspective of the publishers who run the advertising platforms, the advertisers who subscribe to these platforms, the consumers who see the ads along with their search results on the search engines. Accurate advertising is yet another huge area of research that most publishers are working on right now to figure out ways of improvement, as this will help them display the most relevant ads to the consumers. It is certainly not enough if the publishers display the ads only based on the keywords that the advertisers choose for their campaign as there are many other highly weighted attributes that are important in this problem. For instance, it might be worthy depending heavily on the geo-location in addition to the search term from the consumer. Hence, considering this as a bigger problem, prediction of the click through rate becomes the sub problem in aiding the publishers to charge the advertisers and also for the advertisers to choose the right mode of online advertisements.

1.3 Problem Statement

The fundamental problem addressed in this paper is to predict the click through rate of the advertisement which will be directly proportional to its efficiency. There are several key aspects that are considered when measuring the effectiveness of an advertisement. They are the click through rate (CTR), the purchase list, the follow-on search behavior amongst other domain specific ones. As the advertisements are sold to the business on a pay per click basis, it is important for the publishers to maximize the clicks which in turn is a direct benefit for the business in terms of the promotion they receive. We will use the previous statistics from an advertisement publisher and predict if an advertisement will be clicked by a user or not based on the features of the advertisement.

The click through rate (CTR) can be defined as the ratio of the number of clicks that an advertisement obtains to the number of impressions of it [2]. An impression can be defined as the total number of times an advertisement is displayed to the user. This gives the probability estimate that a user will click on the recommended advertisement. Throughout this study, we rely on the fact that an advertisement will be effective if it has a better ratio of the number of clicks to the number of views.

Another problem is around the various factors that influence the CTR of an advertisement. There are certain obvious factors like the user's profile, the search history of the user and the advertiser's profile, etc. But there might be many hidden correlated features that might affect the CTR. It is imperative to study and consider these feature interactions when designing the models to predict the probability. As the features are implicit and nonlinear [1] like the traditional beer and diaper correlation, this research also concentrates on the feature selection and analysis using different machine learning models.

The dataset for this research mostly combines the advertisement details that are logged by the publishers and the behavior logs of the consumers. A lot of the features in the data comes as anonymous attributes for security and privacy reasons and most of them are categorical features, the dataset itself becomes very highly dimensional [2]. There have been many different machine learning models that were used to solve the problem of predicting the click through rate. This scenario gives a large lee way to analyze and ignore the features whose contributions to the predicted probability are insignificant.

In summary, the key problems are to analyze and choose the right set of minimal features amongst the high dimensionality of the data and to predict the click through rate using an ensemble of the machine learning algorithms like the Random Forest and Logistic Regression, Decision trees which are all suitable for this binary classification.

1.4 Research Objective

The specific objectives of this research are listed below.

- Exploratory analysis on the data set to understand the features on the distribution of the data based the features.
- Preprocess the data to remove the null values and meaningless information.
- Feature selection on the dataset to identify and remove those features that do not contribute significantly to the prediction of the probability of a user clicking any given advertisement.
- Once these rich set of features are decided, one hot encode the features to create vectors and build machine learning models to find the probability.
- Bench mark the predictions of the various models using the confusion matrix, ROC AUC (Area Under the Receiver Operating Characteristics) curve, the sensitivity and specificity curves.
- Also, create an ensemble of the models by varying the feature set and bench mark the same.

Since most values in any dataset that we could use for the CTR prediction will have the details of the user and the advertiser profiles as unique identifiers, the cardinality of the data is huge. To bring it into a reduced dimensional space before we pump the data to the machine learning model, the task is to do a feature engineering [4] and select the attributes that could significantly contribute to the results.

The flow of the project is given as a flow chart below.

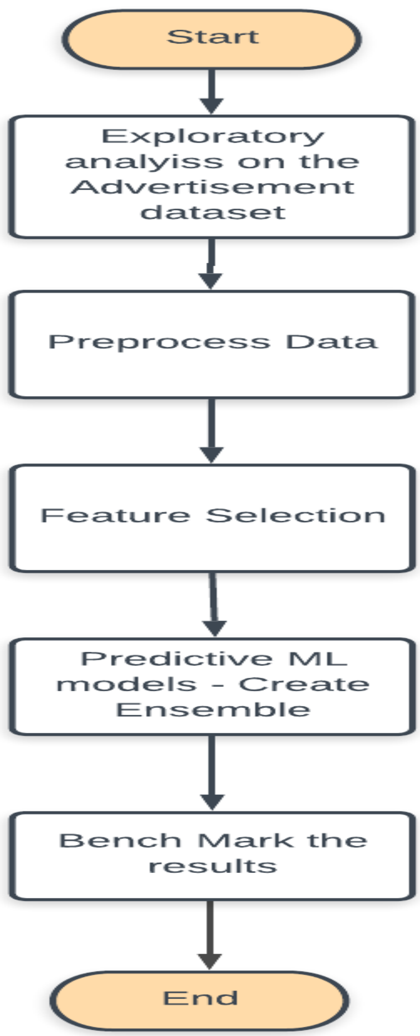

Figure 1 The flow of the research

CHAPTER 2

EXISTING SYSTEM

2.1 Methodologies

The click through rate prediction is a classification problem in which we predict if the user will click the advertisement that is displayed to them or not. The literature study indicates that various machine learning models can be made use of to learn the features from the training dataset. Most of the machine learning algorithms are comparable to each other in the way they preprocess the data. The key difference between the existing researches is in the feature engineering of the dataset [1]. Some of the early researches consider the entire high dimensional data set for the training while certain studies use the machine learning techniques to select the set of features that will help in fine tuning the model in turn improving the accuracy of the predictions.

For the purpose of the learning the interaction between the hidden feature, some of the researches also use the convolutional neural networks to obtain the relationship between the features in a more sophisticated manner [2].

Another difference between the studies are the mechanism used to bench mark the prediction results. Some of the studies use the ROC AUC curves to bench mark the results while a few of the studies use the log loss for the same. ROC, the receiver operating characteristic curve is generally used when we need to bench mark any classification machine learning model. The ROC Curve is a plot between the False Positive (FP) and the True Positive (TP) rate at all possible thresholds between 0 (low) and 1 (high). The AUC, Area Under the ROC Curve, is a measure of performance across all probable limit values.

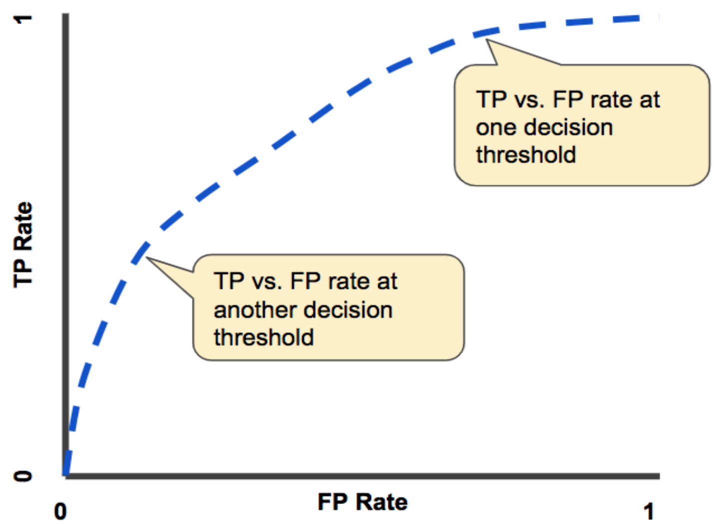

Figure 2 The ROC Curve

Image Source: developers.google.com. (2019) Google Developers. [online] Available at

https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

Because the dataset is observed to have imbalance, the researches use the log loss method to measure the effectiveness of the prediction. The log loss measure also considers the certainty of the prediction again the actual target feature.

The researches also differ in the datasets that they use for bench marking. As the dataset might contain much of the categorical and domain specific features [5], it is also important we consider this factor when we choose the relevant literatures.

2.2 Adscape

Barford et al. [2] proposed the research Adscape, wherein they developed a crawler that crawled multiple websites across the internet to gather the data. This was one of the first few researches in the field of the click through rate prediction outside of the advertisement publishing enterprises, when the data was not available open source to the researches. The crawler had the capability to scale and obtain the details of the advertisements across the internet and models were built to explore and analyze the dynamics and the trends of the advertisement and the clicks [2].

The research mainly concentrated on creating a focus set for this problem. The research reports that the user profile and the advertiser's profiles are the key attributes that decided the effectiveness of the advertisement campaign. The authors implemented a profile builder that mimics to build the profiles of the user based on the website interactions of the user. This way the crawler could capture the data about the advertisement along with the profile of the user using the web tracking technologies. On the other hand, once the data was collected, they also used the services like Bluecoat and Alexa ranking to categorize the advertisers' websites.

Using these techniques, this research identified the alignment between the profiles of the user and the ads that they are displayed with. It is also important to note that there were many websites that displayed ads to the users with no consideration of their profiles. Nevertheless, the research concludes that 80% of the display ads in the data corpus used the profiles of the users as the primary strategy to zero in the target audience. This research helps us understand the various attributes to look for in the dataset we use and decide on the features that we plan to use to train our classification models [3][4].

2.3 DeepFM

Guo et al. have proposed a deep neural network-based model. It is important to understand the correlation between the features to select the focus set. The authors claim that the neural networks are powerful to figure out the interactions between the features. This research proposes a novel method called Deep Factorization Machines [6]. This neural network model could sophisticatedly learn the implicit correlation between the features and learn from them. The dataset used in this research also contains both the categorical (gender for example) and continuous features (age for example). It is important to wisely convert these features into vectors. The DeepFM uses two components namely the Factorization Machine and the Deep Neural Network model that take the same input. A scalar weight is used to measure the importance of a feature and a vector is generated to learn its interaction with other features [1].

Thus, this model learns both the low level (age and banner position for instance) and the high-level feature interactions (age and site category for instance). The authors evaluated their neural network model on the Criteo dataset and show that their model is 0.37% better than the other neural nets in terms of the ROC AUC values and 0.47% better in terms of the log loss values.

In summary, we could say that the improvements in the results are due to the learnings from both the higher order and the lower order features. Additionally, it also comes from the fact this approach considered both the wide and the deep attributes in the highly dimensional sparse dataset that they considered for predicting the click through rate.

2.4 Online Learning

Hao et al. claims that key factor to accurately predict the CTR is to understand the non-linear relations between the attributes. This study proposes a field aware factorization method that identifies the combination of the features that have high influence in the sparse dataset. The authors use the KDD cup dataset for their experiments [4]. With this field aware model, the authors were able to map the non-linear lower order mapping between the gender and the products. For example, the gender 'male' is identified to be mapping with a lot of sports products while the gender 'female' mapped to a lot of cosmetic items. These relationships are totally implicit and require extensive feature engineering process to accurately learn them.

The authors also tried to build an online supervised learning machine learning algorithm using the gradient descent method. The batch processing machine learning algorithm takes in the input from the previously captured data and trains on variants of the train dataset to avoid over fitting on the test predictions [7]. This method although brings in a high accuracy in general, but is not able to support real time processing of the data which is of supreme importance. The online learning algorithms are able to take the near real time streams of the data and are able to improvise the previous learnings to help in predicting the new trends. This possibility of predicting the new trends is generally known to be traded for a slight dip in the accuracy of the prediction [3].

This study reports that they outperform other models like the logistic regression by 0.65% to 6.44%. However, it is important to note that this study used a different data set from the ones discussed above to bench mark their results.

2.5 Hierarchical Extreme Machine Learning Model

This research by Zhang et al [5] have predominantly studied the imbalance learning in predicting the click through rate. The authors propose a hierarchical extreme output learning scheme that consists of two levels. At the first level, two weighted models Weight Output Extreme Learning Machine (WO-ELM) and the Weighted Extreme Learning Machine (W-ELM) get trained on the different set of combined fields. These two models will have difference in the results of the prediction which is obvious as they both used different combination of the input attributes. At the second level, these two predictions are weighted and combined in order to get a clear prediction if an advertisement will be clicked [8].

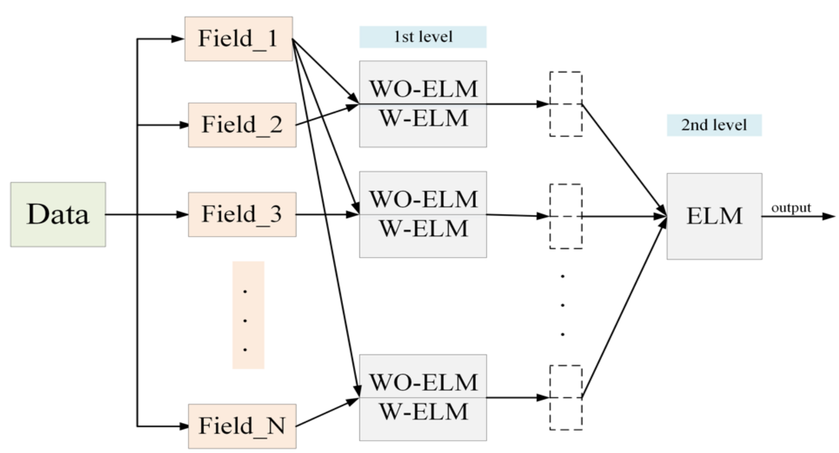

Figure 3 The Hierarchical model

Image Source: [5]

This research uses the Area Under the ROC to bench mark the results and achieved an AUC score of 0.731 on 1. This model clearly shows an improvement of the AUC score but at the cost additional computations. The future work of this model also talks about identifying the deep interactions between some of the features that this model might have ignored.

2.6 Deep Interest Network

Fan et al. [3] have been able to propose this novel approach of using a deep interest network to predict the click through rate of a candidate ad. This study is first hand from the researchers at Alibaba.com. Since the authors had access to all the attributes and dimensions of the dataset, they had worked on avoiding the fixed length vectors to reduce the dimensions of the dataset. The study has identified the historical behaviors of the users with respect to any advertisement and designed an activation function to adaptively learn them.

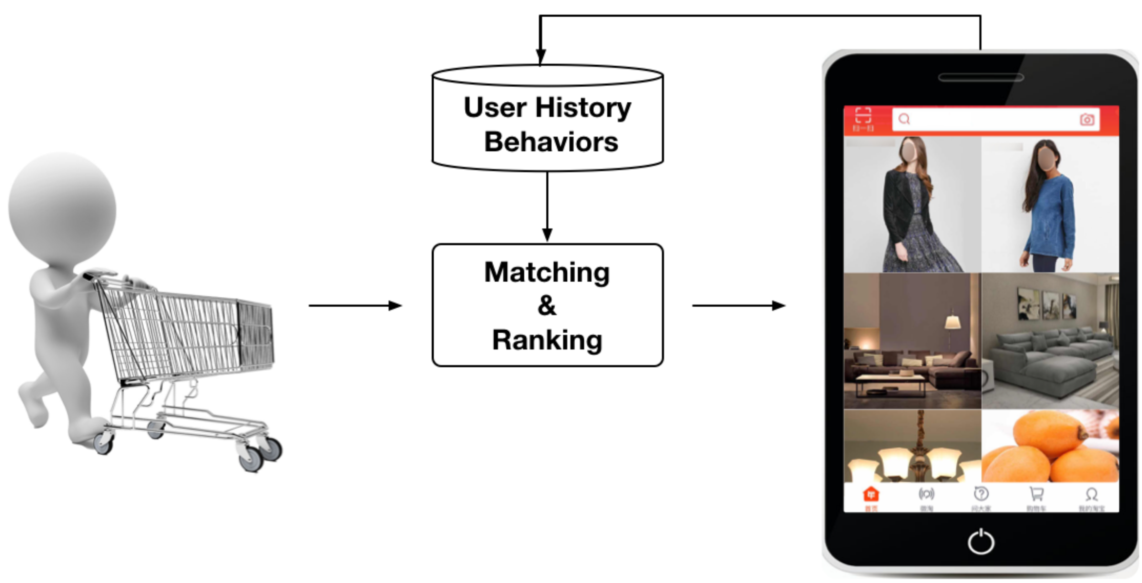

Figure 4 The Running procedure of the display ads in Alibaba.com

Image Source: [3]

The e-commerce website, Alibaba.com first generates a list of ads that are useful to be displayed to the user based their search using techniques like the collaborative filtering. Once this step is done, they use the deep interest neural network that uses multi hot encoding to vectorize the categorical attributes (the user behavior) and predict the CTR of any advertisement. These prediction probabilities are ranked and are used in displaying the ads to the consumer. It is observed that the Deep Interest Network model outperformed the other models discussed by an absolute value of 0.0059 [3]. The observations discussed here will be more useful when there are no anonymous attributes [8] in the dataset or the relationship between the anonymous attributes are studied in full.

CHAPTER 3

DATASET

This section discusses the dataset that this study has used for experiments and the bench marking. The details of the data are as below.

3.1 The Avazu Advertisement Dataset

The Avazu is a leading advertising platform that has provided the researchers with the dataset. The data set contains rich information about the advertisement that was displayed over a period of 10 days sorted according to increasing time [1]. This can be used for the training purpose. Additionally, the test data also covers the same set of features as the training data and has features of the ads that were displayed in a day. The details about the ads that were clicked and not clicked are said to be sampled according to some undisclosed strategies [9]. The dataset is available online at https://www.kaggle.com/c/avazu-ctr-prediction/data

The dataset is seen to contain 21 different attributes of which the target attribute is the click details which is binary typed, 0 meaning an advertisement was not clicked and 1 meaning the click of an ad by the user. The dataset contains about 8 of the 21 features as the anonymous features. These are categorical variables and could be the detailed information about the users' profile and the advertisers' profile and are hashed to a unique value for the researchers to frame the vectors [4][9].

Some of the key qualities that qualify this data set for this research are its enormity and coverage about the user, the advertisement and also about the advertiser. This helps us with rich useful information in categorizing the data. Although there are anonymous attributes, they are hashed to unique values and hence it becomes seamless when we need to reduce the dimensional space of the dataset in hand.

The various attributes and their definitions are tabulated below.

Attribute	Definition
id	The unique identifier for all details that corresponds to one occurrence of an advertisement. This is a continuous variable.
click	The target variable, 0 means an advertisement was not clicked and 1 means the ad was clicked. This is a categorical variable, binary typed.
hour	The hour, in YYMMDDHH format. We could break this down and add additional features during the cleaning process. This is a continuous variable.
banner_pos	The position in the screen where the advertisement was displayed. This shows the prominent place for an advertisement to get the attention of the user. This is a categorial integer
site_id	The identifier to unique identify a site in which the advertisement was displayed. This is a hashed value.
site_domain	The domain information of the website in which the advertisement was displayed.
site_category	This is a categorical variable representing the field to which the website belongs to. This can be used

	to understand if any site category has more visitor attraction during any particular time.
app_id	The identifier to unique identify a mobile application in which the advertisement was displayed. This is a hashed value.
app_domain	The domain information of the application in which the advertisement was displayed.
app_category	This is a categorical variable representing the field to which the application belongs to. This can be used to understand if any app category has more visitor attraction during any particular time. This is similar to the site category and can be compared relatively to check if app has more clicks over the website.
device_id	The unique identifier that marks the device from which the click was captured. This is a hashed continuous variable and can be repeated in the data set.
device_ip	The ipv4 address of the device from which the click was received. Hashed to a different value for privacy reasons to avoid trace back to the device.
device_model	The model of the device. We choose not to use this value.
device_type	The type of the device, is a categorical variable and has around 7 categories.

device_conn_type	This is a hashed value about the connection type. We do not use this value for forming the vector.
C1	An anonymous variable. It has influence over the prediction.
C14 – C21	Anonymous categorical variables that might have information about the advertisers' profile and the users' profile like the age, gender, etc.

Table 1 Features in the dataset

We could classify these attributes in the data set into four sets and one target variable 'click'

- Site specific and app specific attributes
- Device specific attributes
- Ad specific attributes
- Anonymous attributes

The site and app specific attributes give information about the advertiser who are running the advertisement campaign. The anonymous attributes could contain the user specific and revenue specific details which are hidden for privacy reasons. The train data is approximately 6 Gigabytes and test data is approximately 1.5 Gigabytes in size. We intend to randomly sample the data to avoid over fitting during bench marking.

CHAPTER 4

FEATURE ENGINEERING

4.1 Data Preparation

The feature engineering [4] is a process to filter and choose quality features for the machine learning algorithms to learn from and predict the target value accurately [1]. These features in the training dataset are the key ones that correlate directly to the target variable ‘click’. The literatures [3][8] recommend that the quality features should be independent of each other. If two features could mean nearly the same, it is not worth including in the feature set as it will reduce the accuracy of the model. Our dataset has features like the device_id, device_ip etc., which will not help us in predicting if an advertisement was clicked or not.

We also figured out that the attribute ‘hour’ in the dataset is in the format YYMMDDHH as a number. This feature is worthy in terms of understanding the statistics. For example, a business could expect a lot of enquiries during the peak hours but not during other times. But, from the perspective of prediction, training the model with this feature brought down the accuracy and hence this attribute is broken down to ‘hour’, ‘day’, ’year’,’ month’. We use data frames to add these additional attributes to the dataset.

There were also null values in the dataset, although minimal. Those advertisement that were site based didn’t have values for those one that were app based. The ip address values were not in the right format. In most cases, we attribute the missing values to the most frequently occurring attributes that we found during the data exploration process.

An important problem with the dataset was its monstrousness [2]. The data was very huge that even Google Collab TPU infrastructure was not able to process. We sampled the data for each of our machine learning models to get trained. Also, much of the attributes were categorical and were anonymous. We used one hot encoding to convert the features to vectors when training the model [9].

4.2 Exploratory Analysis

The next step as in any machine learning project is to understand the data [1]. We use the standard exploration techniques and plot the results to get an idea about the various values and implicit meanings contained in the features of our dataset [14].

Out of all the explorations computed, the most relevant and useful insights are included here.

4.1.1 The number of clicks received by Avazu on any single day

```
: sns.lineplot(x="hour", y="click", hue="device_type", style="device_type", markers=True, dashes=False, data=train)
  plt.ylabel('Clicks')
  plt.xlabel('day');
  plt.title('Clicks vs day based on device type')
```

```
: Text(0.5,1,'Clicks vs day based on device type')
```

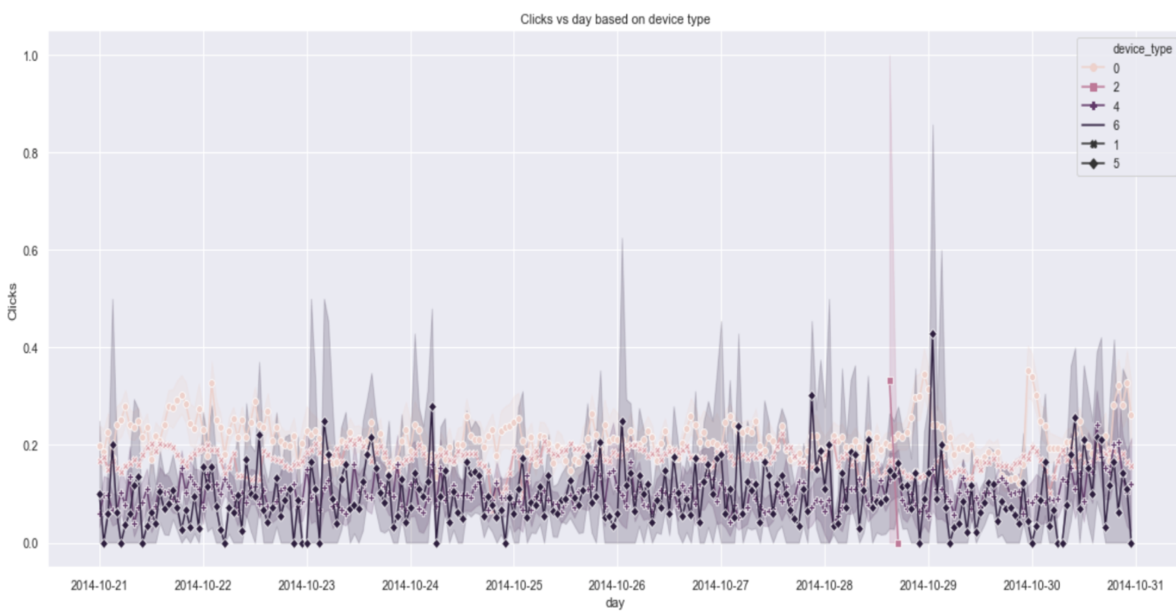

Figure 5 The number of clicks received by Avazu on any single day -based device type

The above plot is between the day when the click is obtained and the number of clicks that were obtained reduced to scale of 1. We also base this insight on the device_type, which is a categorical field that has values from 0 -6. We observe that the maximum number of clicks were received from the device with type 0. The one below shows the same relationship but based on the position of the advertisement where it was displayed instead of the device_type.

```python
sns.lineplot(x="hour", y="click", hue="banner_pos", style="banner_pos", markers=True, dashes=False, data=train,
             palette=sns.color_palette('coolwarm', n_colors=7))
plt.ylabel('Clicks')
plt.xlabel('day');
plt.title('Clicks vs day based on banner position')
```

```
Text(0.5,1,'Clicks vs day based on banner position')
```

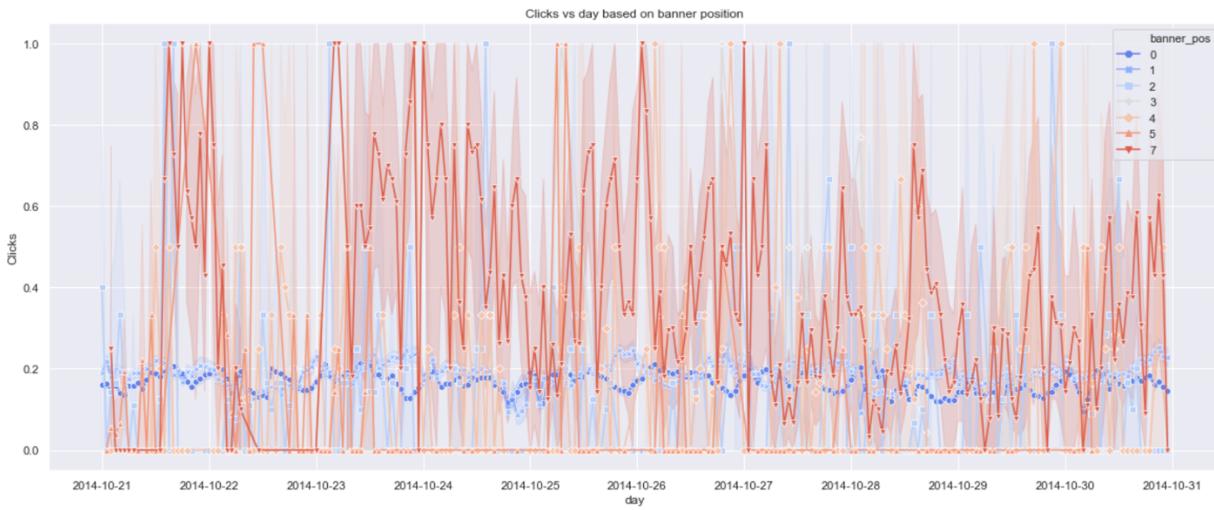

Figure 6 The number of clicks received by Avazu on any single day based banner position

4.1.2 Relationship between the number of clicks and non -clicks received by Avazu on any single day

This is a key insight that helps us understand the relationship between the number of ads that were shown to the user to the number of ads that were clicked by them. These two are the essential factors that help us in determining the click through rate of an advertisement. We have based this plot on the banner position additionally and understand that the values range form 0-7 for this categorical variable and most of the ads in this system were displayed at positions 0 and 7. We also use the device type to understand the relationship.

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(rc={'figure.figsize':(11.0,8.0)})
sns.countplot(x='click',hue="banner_pos",data=train, palette=sns.color_palette("Blues_d"))
plt.ylabel('No. of Clicks')
plt.xlabel('Click');
plt.title('Clicks vs Count based on banner position')
plt.show();
```

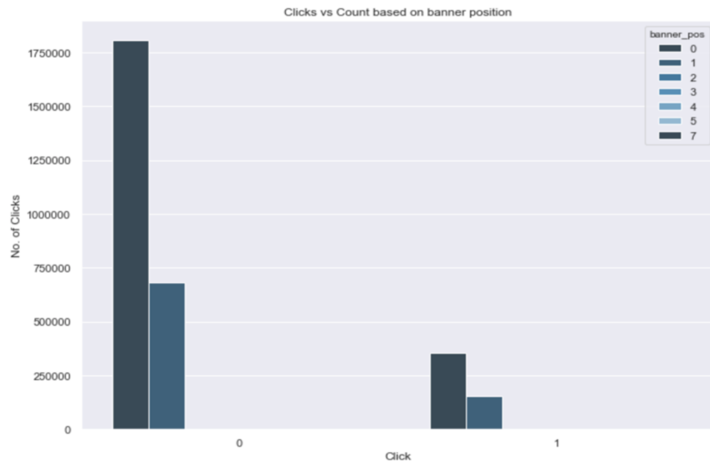

Figure 7 No. of clicks to non- clicks based on banner

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(rc={'figure.figsize':(11.0,8.0)})
sns.set_style("darkgrid", {"axes.facecolor": ".9"})
sns.countplot(x='click',hue="device_type",data=train, palette=sns.hls_palette(8, l=.3, s=.8))
plt.ylabel('No. of Clicks')
plt.xlabel('Click');
plt.title('Clicks vs Count based on device type')
plt.show();
```

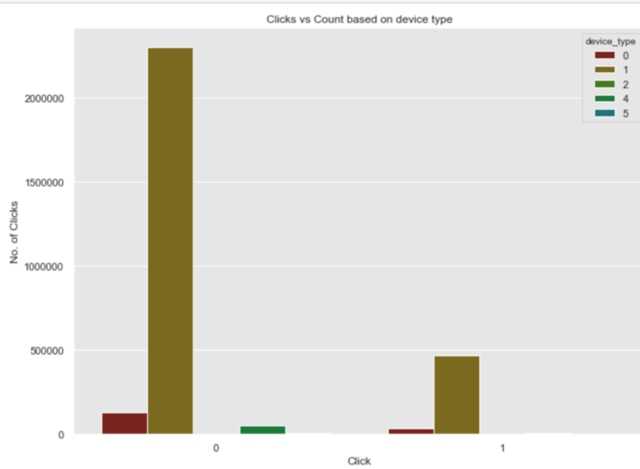

Figure 8 No. of clicks to non- clicks based on device type

4.1.3 Relationship between the hour and Click achieved

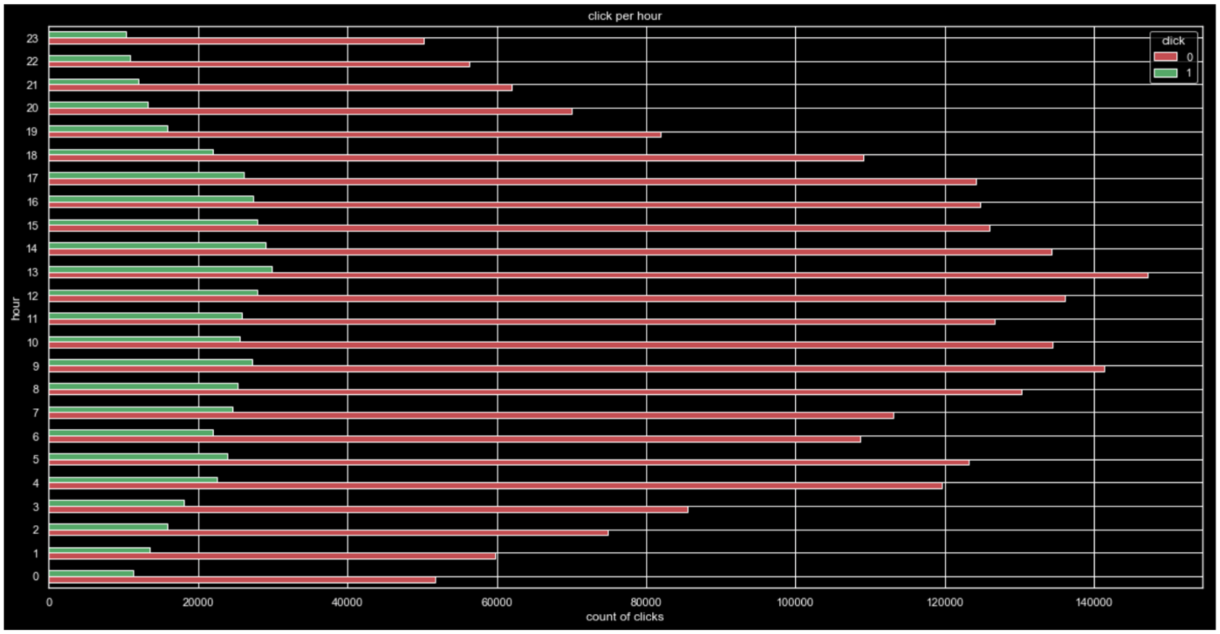

Figure 9 Hour vs click aggregate

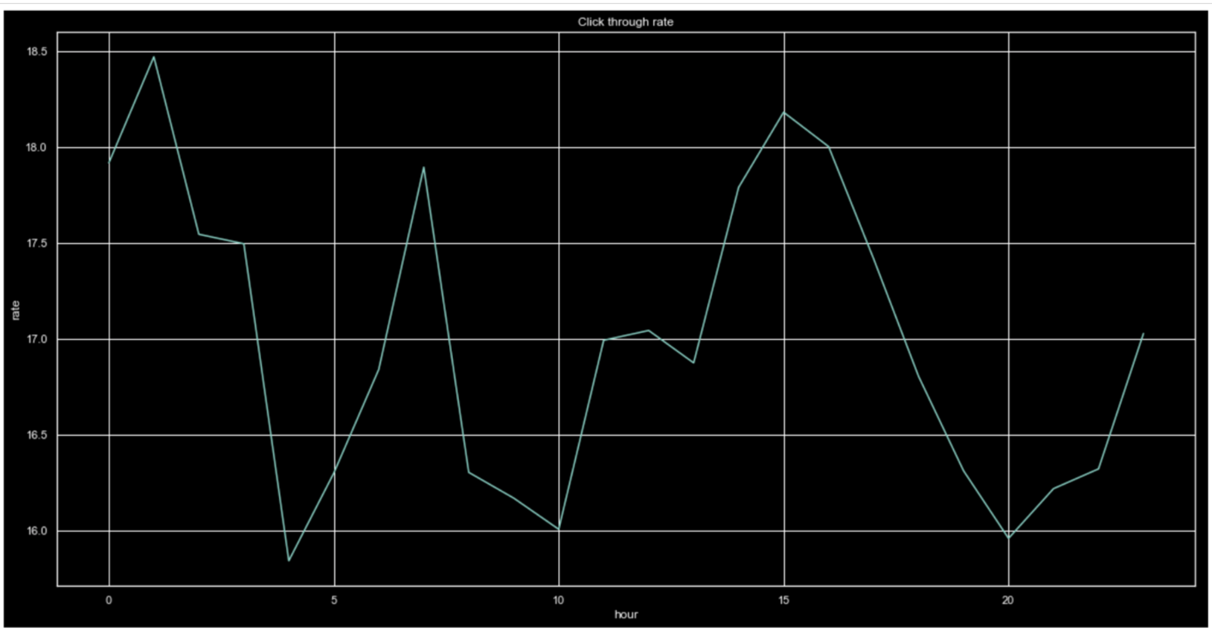

Figure 10 Hour vs click through rate

We generate the CTR as the number of clicks to the number of the impressions and plot them as a graph to compare with the final results of this project. It is interesting to note that the maximum click/CTR is achieved during the night time than during the day.

4.1.4 Relationship between the anonymous attribute ‘C1’ and the count of the clicks

The attribute C1 is a categorical variable that is anonymized. We see this as an important feature for the training as most of the advertisement that received a click or not depends on this feature. We could see that the attribute value, 1005 has the maximum number of clicks and impressions.

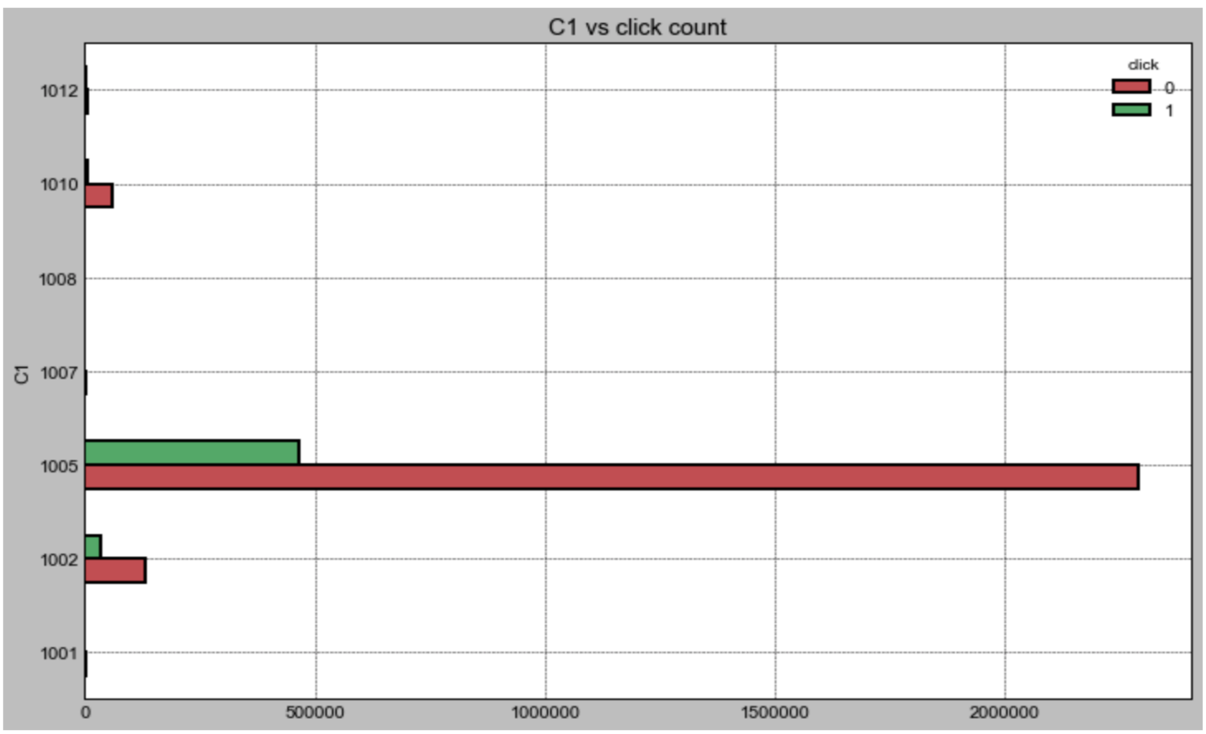

Figure 11 anonymous attribute ‘C1’ vs the count of the clicks

4.1.5 Click Variations by granular time

To understand the granular variations of the number of the clicks, we create a plot between the total number actual clicks in the dataset to the time with-out considering the device type or the banner position.

```
train.groupby('hour').agg({'click':'sum'}).plot(figsize=(20,10),color='blue', marker='|', linestyle='dashed',
linewidth=2, markersize=12)
plt.ylabel('Number of clicks')
plt.xlabel('day')
plt.title('Number of clicks vs hour');
```

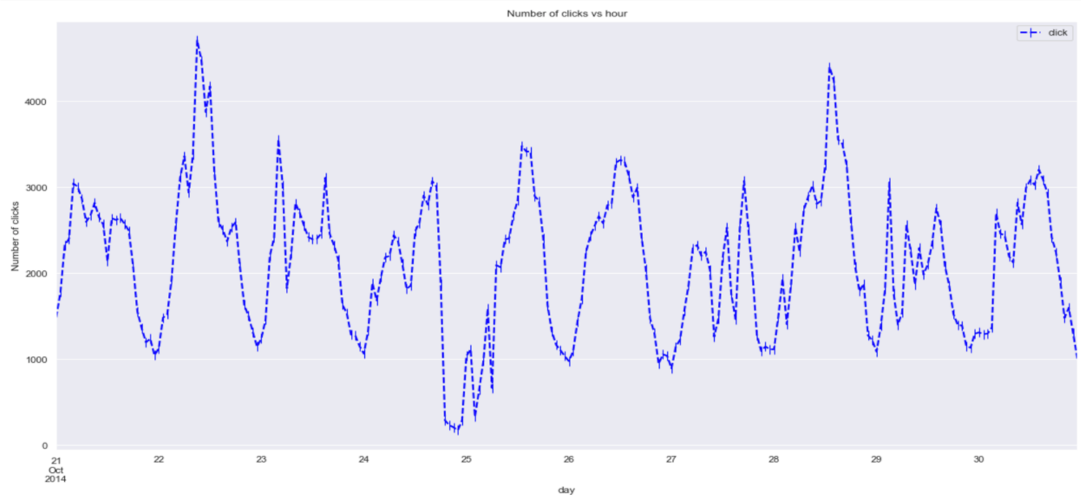

Figure 12 click vs day

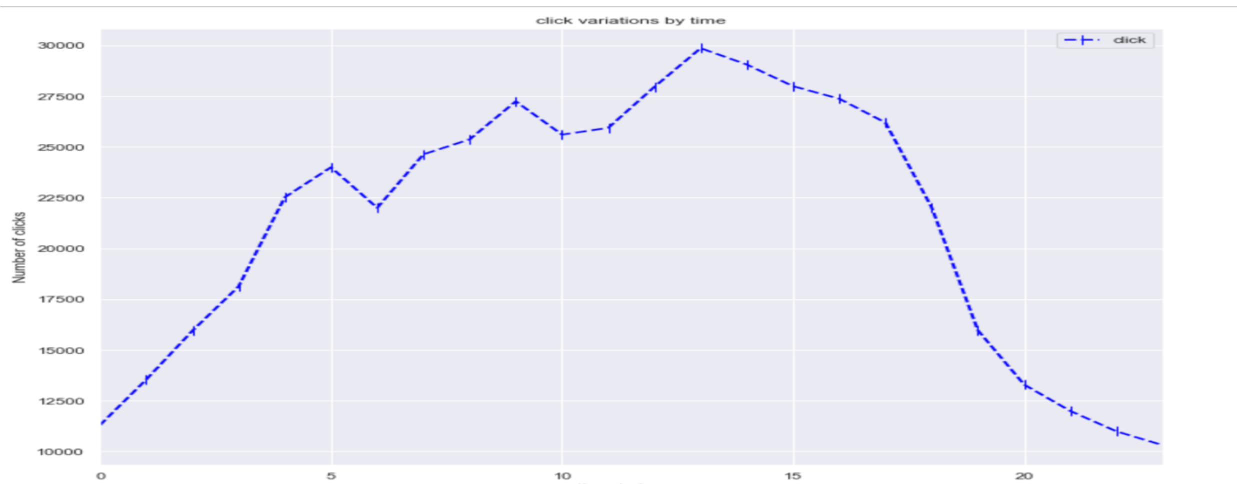

Figure 13 click vs time

CHAPTER 5

ALGORITHMS USED

Based on the literature that studied recommendation of ecommerce products to the users and the relevant literatures that worked on predicting the click through rate of any advertisement dataset, we have used the following machine learning algorithms to predict the probability that a user will click the ad.

- Logistic regression (Online Method)
- Stochastic Gradient Based Logistic Regression
- Random Forest Algorithm
- XGBoost algorithm
- Decision Tree model

We have predominantly used the Scikit learn libraries to implement these models.

5.1 Logistic Regression

The logistic regression [10] is a widely used projecting analysis algorithm for most of the classification problem. It can be used to classify targets with binary values, multi categorical or ordinal. When classification categories are more than two, we could use softmax strategies in the regression learning process. The training happens using the series of occurrences of the binary classification problem and during the testing, we compute the target probability for any occurrence of the same classification problem. The final predicted probability tells us the relationship between the predicted value and all the other features considered during the training phase. We use the online learning feature of the logistic regression algorithm as it is useful to provide the predictions in real time [1] using Apache Kafka streams to build the data pipeline. One important thing to note when using this algorithm is to remove the correlated features to avoid the over fitting or under fitting of the trained model [4]. We use the predict_proba() method in the scikit learn library to compute the probability for every data point in the test set.

5.2 Stochastic Gradient Based Logistic Regression

The problem of predicting the click through rate is rather occurring in the online infrastructure. Although batch processing machine learning models tend to give better accuracy in one time, it is important that in this class of problem, the model has to iteratively learn and change the weight and cost function accordingly to improve the accuracy rate [11][18]. This model looks at the training data, occurrence by occurrence and generates the prediction of the target variable. This prediction will have an error rate. This is compared by looking at the actual target variable. The model learns from the error rate and goes towards the minimum. Thus, this is an optimized model to work in an online environment for predicting the probability of an ad being clicked.

5.3 Random Forest Algorithm

The random forest is an ensemble [12] of multiple decision tree learning algorithm. The algorithm selects different set of features and if we look closely at the implementation of the Random Forest, it could intuitively figure out which of the features that are used as the parameters for the training are strong candidates for the prediction of the target variables. This way, the random forest algorithm is able to learn the importance of any feature relatively [13]. We could also tune the model user the hyper parameters and max features to be used.

5. 4 XGBoost algorithm

The XGBoost is a very famous boosting technique that can be used to improve the efficacy of a model in terms of the training time. The XGBoost [4][14] algorithm is benchmarked and compared with the random forest algorithm on many different classification problems and have seen huge improvements in the running time. The XGBoost is also an ensemble method like the random forest and it uses both the bagging and the boosting technique. The set of features are bagged together and a decision on the target variable is made based on that set. Using the boosting technique, the error residue is calculated and the decision tree in the next iteration matures to reduce the error occurrence. In contrast to the bagging technique, the tree does not grow as much, as boosting will use a smaller number of splits in the iterations. These techniques make this model a natural choice for classification problems.

5.5 Decision Tree model

The decision tree model constructs a tree based on the features that are fed to it during the training phase. The correlated parameters must be removed to handle the fitting of the model [7]. The decision trees will construct the rules and these rules will be matched on the new occurrence of the problem in the test set and the corresponding target value will be returned [7].

We use the ensemble of these models. The data is randomly sampled and fed to these models. When the prediction is made, we aggregate the results of the prediction to return the probability.

5.6 Ensemble Model

We create an ensemble of all the above discussed algorithms [15]. We use the weighted classification method to create an ensemble model. The weights for each of these models [11] were decided based on the performance of the models and are discussed in the implementation section later. The training data is randomly sampled (S_1 – S_m) for 300000 data points and are fed to the decision tree, online logistic regression, the stochastic gradient and the random forest machine learning algorithms to develop classifiers C_1 to C_m. The classifiers individually predict the probability and the outputs are combined to infer the ensemble result.

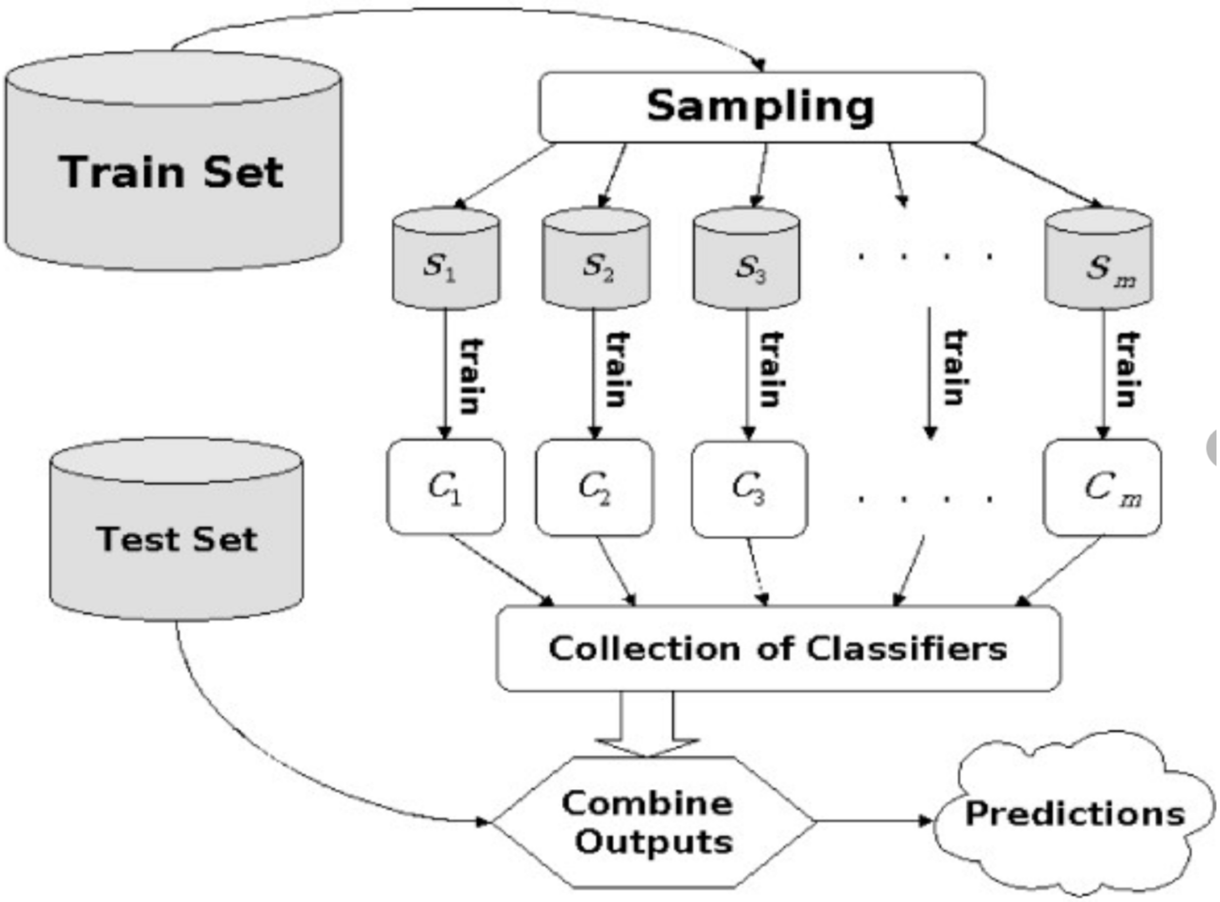

Figure 14 Sample Ensemble Model

Image Source: [27]

CHAPTER 6

BENCH MARKING TECHNIQUES

In this study we use the following evaluation techniques [19].

- The ROC AUC Curve
- The sensitivity specificity curve [21]
- The Confusion matrix

6.1 ROC AUC Curve

The ROC can be expanded as the Receiver Operating Characteristics [20]. The AUC is the Area under the ROC. This measure can be used to bench mark the results of the classification model. We could visualize and see how accurately the model is able to predict the True Positives (TP) and the True Negatives (TN). This is in terms of how the model is able to strictly distinguish between two classes of the target variable that is considered [14]. The ROC AUC curve chooses all possible threshold intervals between 0 and 1 and all occurrences of the positive predictions that crossed the threshold are the true positives (shown in blue) and the negative predictions that crossed the threshold are false positives (shown in yellow) [16]. Similar indications can be observed for True negatives (shown in purple) and False negatives (shown in orange).

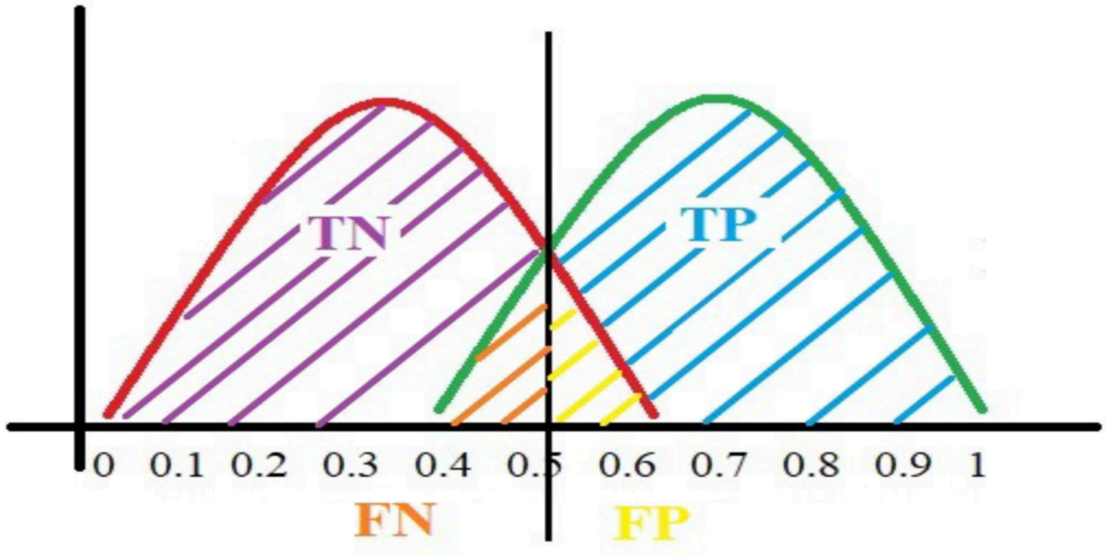

Figure 15 ROC AUC Curve

Image Source: medium.com. (2019) [online] Available at

https://medium.com/greyatom/lets-learn-about-auc-roc-curve-4a94b4d88152

6.2 Sensitivity and specificity curve

This is an additional bench marking that we do to understand the accuracy of the prediction [16][17]. This measure is very important when there is an imbalance in the incident points in the training data. This sensitivity or the recall is the ratio of number of incidents that are predicted as true to the overall predictions. Similarly, the specificity means to say the ratio of the predictions that are false to the overall predictions. The sensitivity and the specificity tell us how suitable a model is to predict a class of the target variable(true/false).

6.3 Confusion matrix

The confusion matrix [23] is a more common accuracy measure in the machine learning projects. It gives the relationship between the actual class versus the predicted class of the target value. We can generate the absolute matrix or the normalized confusion matrix [18].

	Predicted class		
Actual Class		Class = Yes	Class = No
	Class = Yes	True Positive	False Negative
	Class = No	False Positive	True Negative

Figure 16 Confusion Matrix

Image Source: blog.exsilio.com (2019) Exilio blog [online] Available at

[*http://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/*](http://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/)

CHAPTER 7

EXPERIMENTS AND OBSERVATIONS

We wrangle the dataset and use the data preparation techniques mentioned in the section 4. Post these steps, we build the decision tree, the logistic regression, the stochastic gradient descent and the random forest machine learning models discussed using the libraries mentioned below, to name a few.

- NumPy
- Csv
- Pickle
- Scikitplot
- Matplotlib [26]
- Sklearn [24]
- Seaborn [25]

7.1 Machine learning experiments

7.1.2 Approach 1

We sample 300000 data points out of the 700000 data points available for training due to the enormous amount of data available. This approach is chosen safely as the distribution of the clicks versus is the non-clicks is the similar across the days [8]. This sampling technique helped us overcome the computation issues. We induce new attributes, the day, the hour of the day and the date instead of the 'hour' attribute in the dataset. Additionally, all device specific features are removed during the data preparation phase. We use one hot encoding to covert the categorical variables to a fixed length vector [8]. Post this, we run machine learning algorithms discussed above, adding and removing the anonymous categorical variables as part of the parameter tuning process in the motive of improving the accuracy. The experiments and their results are captured below. As mentioned, we bench mark using the ROC AUC curve [22], the precision and the recall curves as well as the Confusion matrix [24]. The same approach is followed for bench marking all the algorithms. We used the Google Collab hosted online at https://colab.research.google.com. We use the Tensor Processing Unit, the default K80 Core of the online google collab environment for quick execution during the training phase.

Results of the Decision Tree Model

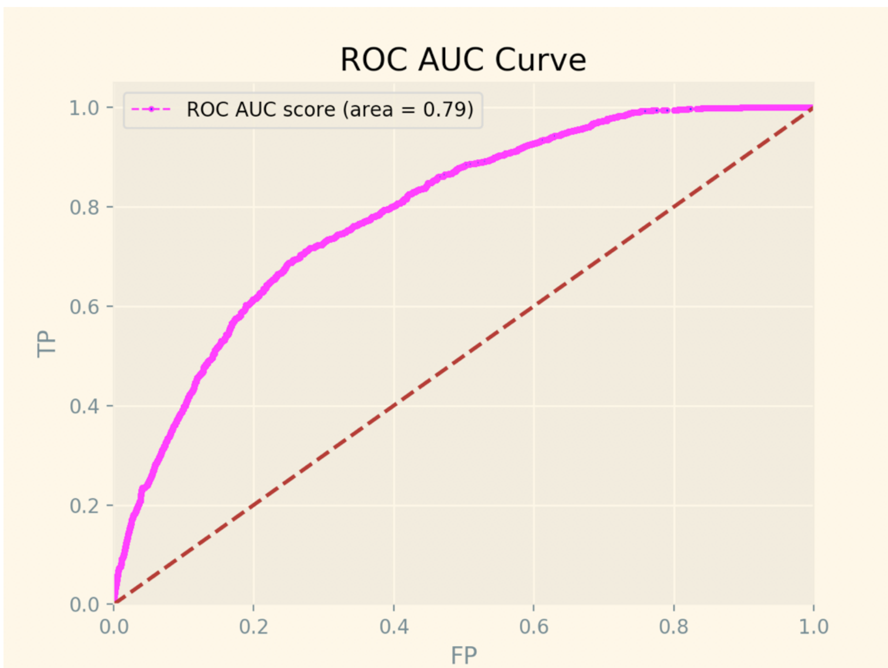

Figure 17 ROC AUC Curve – Decision tree model

The above curve indicates that the decision tree model is able to classify strictly between the positive and negative data points, click vs no click in this case, 79 percent of the occurrences. The ROC AUC is a suitable bench marking technique for the CTR prediction dataset as it has huge imbalance between the class of data points. We observed during the exploration phase that 80% of the advertisement in the dataset has received no clicks versus the 20% of the data samples that recorded a click.

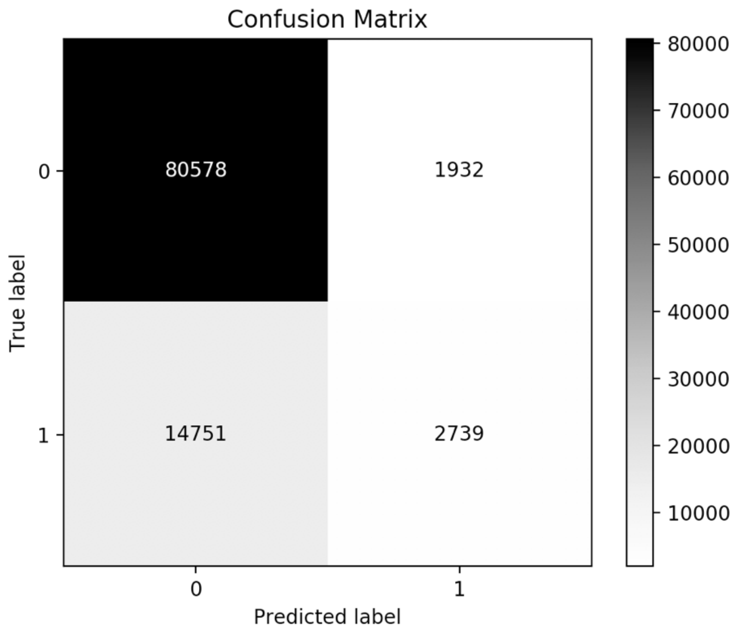

Figure 18 Confusion Matrix Decision tree model

The above confusion matrix shows that the model was able to classify the data points that didn't not receive a click as 0 correctly most of the times. But we see that the model is not able to rightly classify the true positives much of times. We observe this due to the imbalance in the dataset. We overcome this using the sampling techniques discussed in the approach 3.

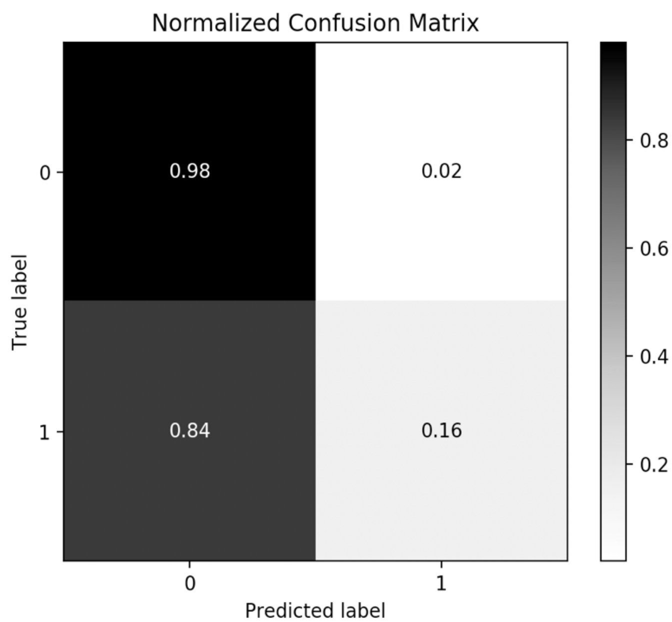

Figure 19 Normalized Confusion Matrix Decision Tree Model

As indicated in the confusion matrix, the model has a very good precision but is inefficient with the recall bench marking strategy.

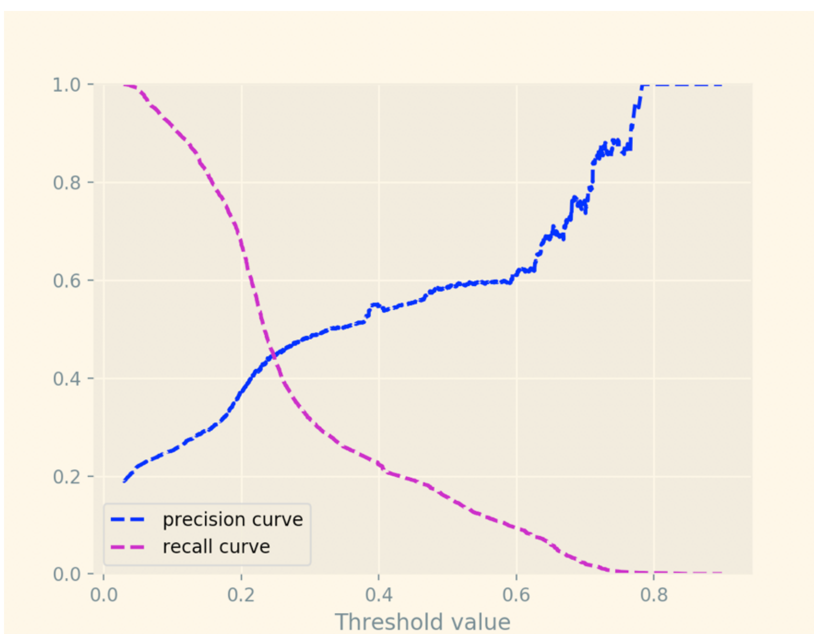

Figure 20 Precision and Recall curves Decision tree Model

The same graphs are plotted for all the other algorithms and are compared.

Results of Online Learning Model

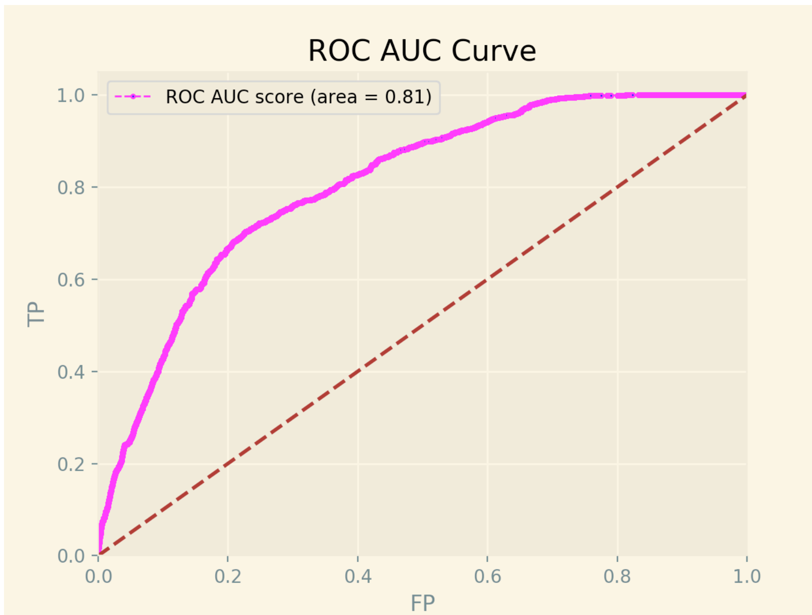

Figure 21 ROC AUC Curve Online Learning

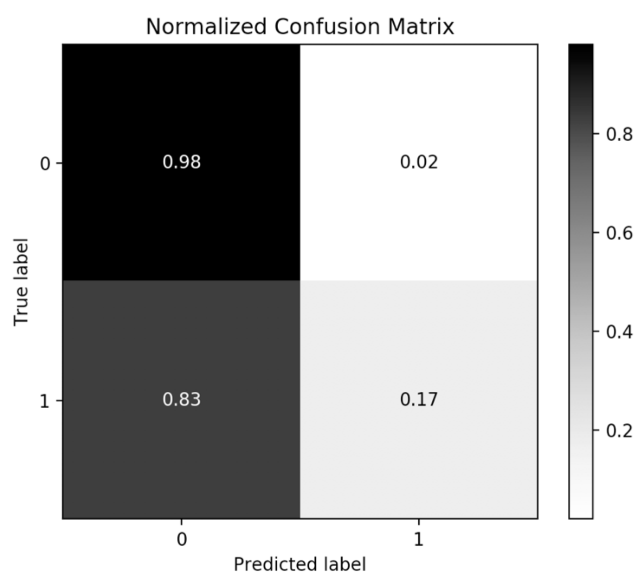

Figure 22 Normalized Confusion Matrix Online Learning

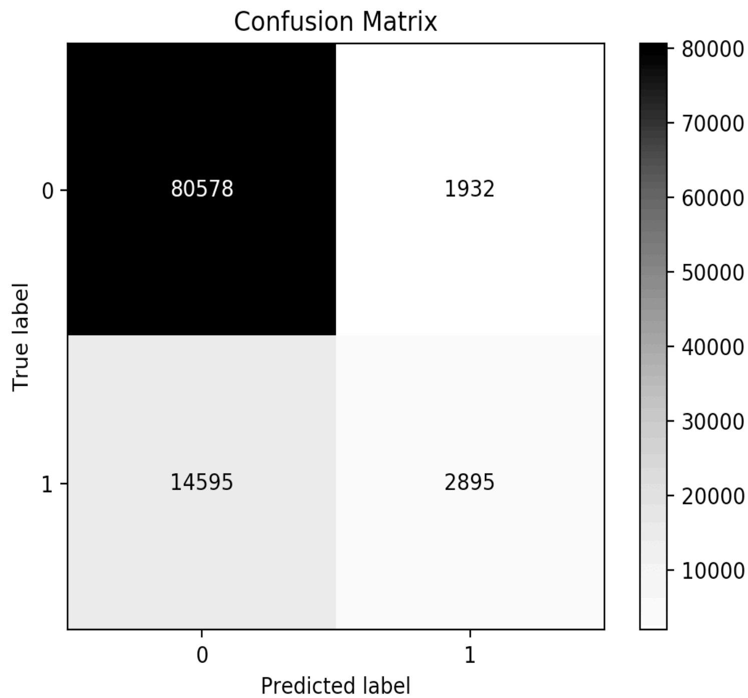

Figure 23 Confusion Matrix Online Learning

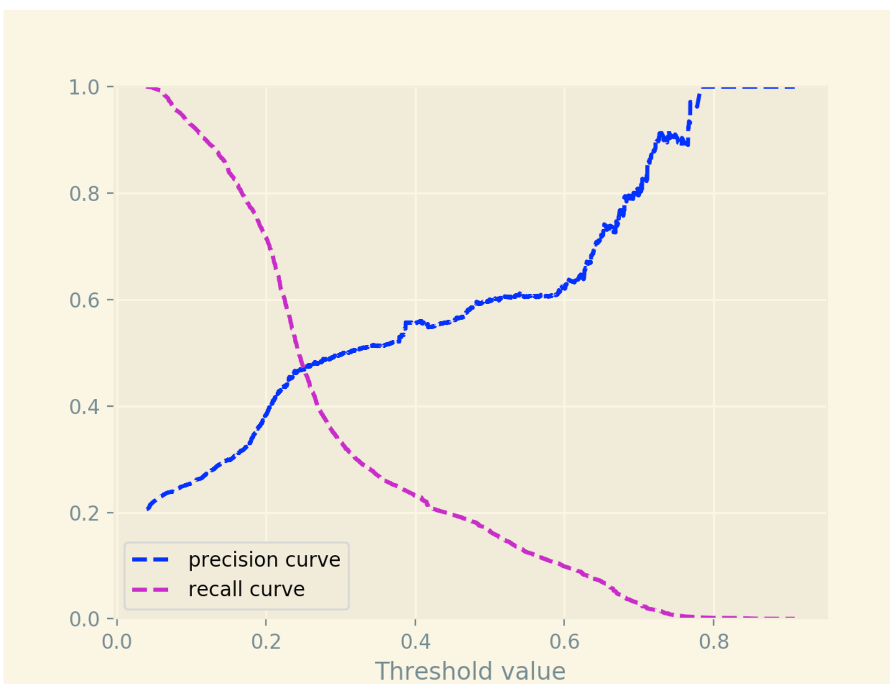

Figure 24 Precision Recall Curves Online Learning

Results of Random Forest Model

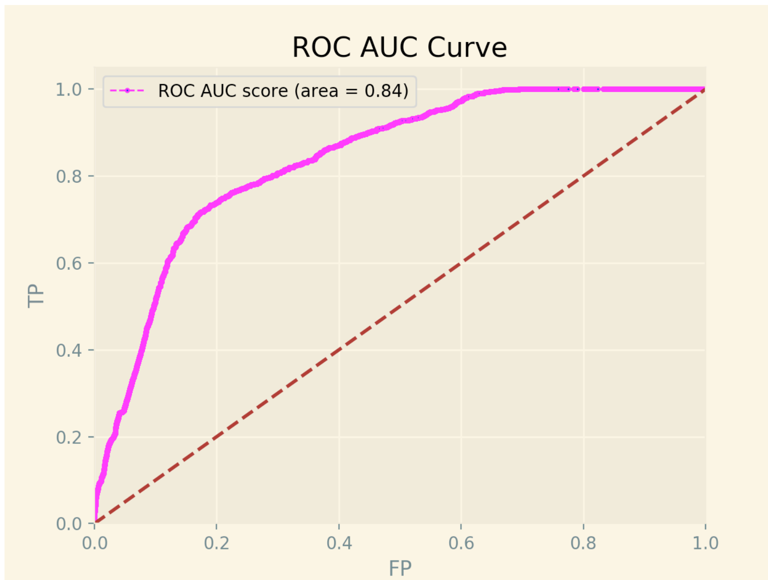

Figure 25 ROC AUC Curve Random Forest Model

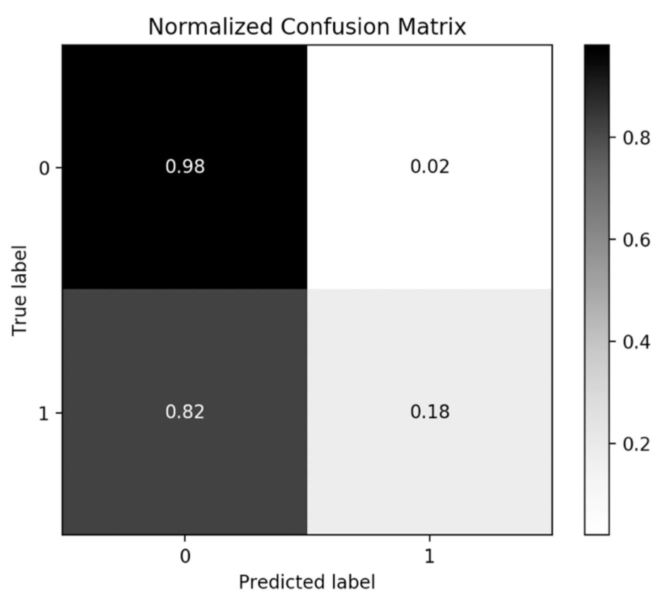

Figure 26 Normalized Confusion Matrix Random Forest Model

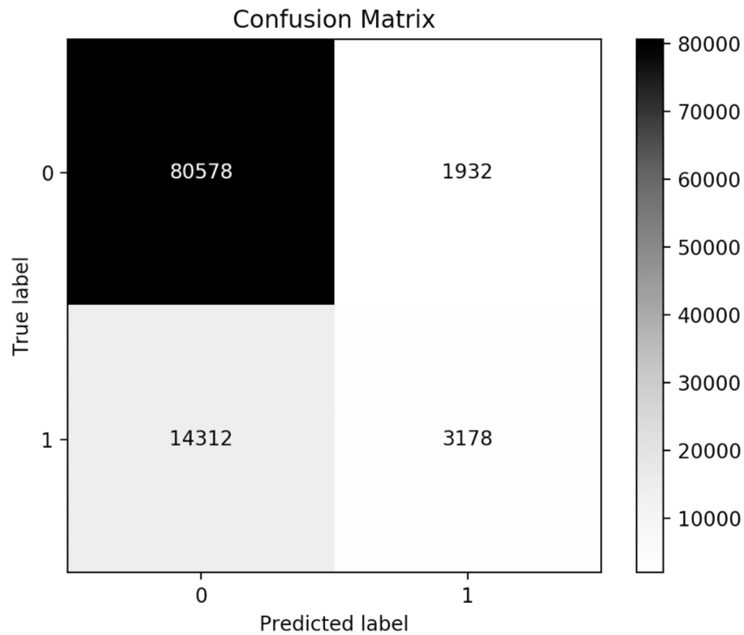

Figure 27 Confusion Matrix Random Forest Model

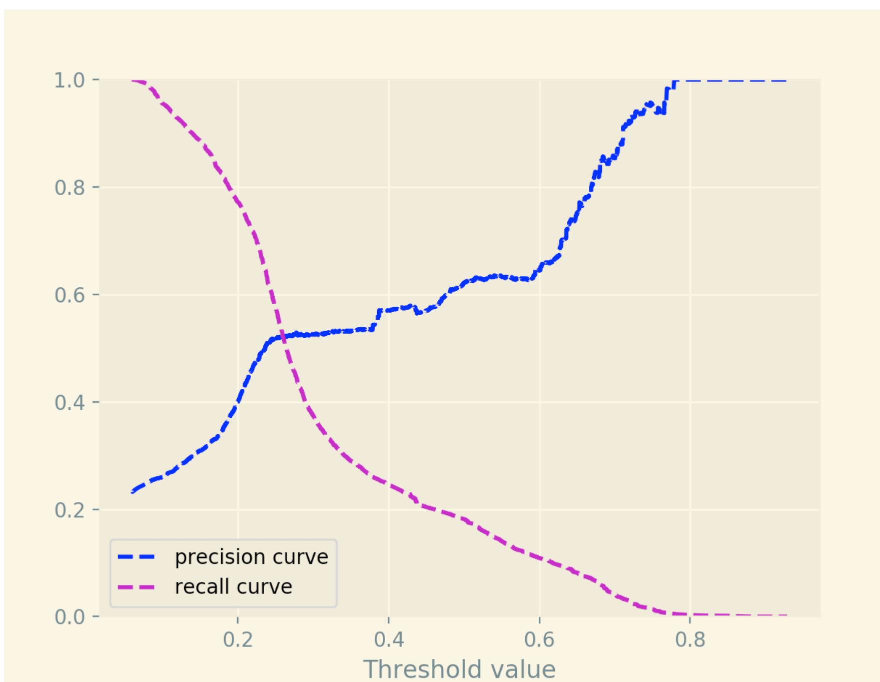

Figure 28 Precision Recall Curves Random Forest Model

The results of the XGBoost technique was also very close to the Random Forest technique.

Results of Logistic Regression (Stochastic Gradient) Model

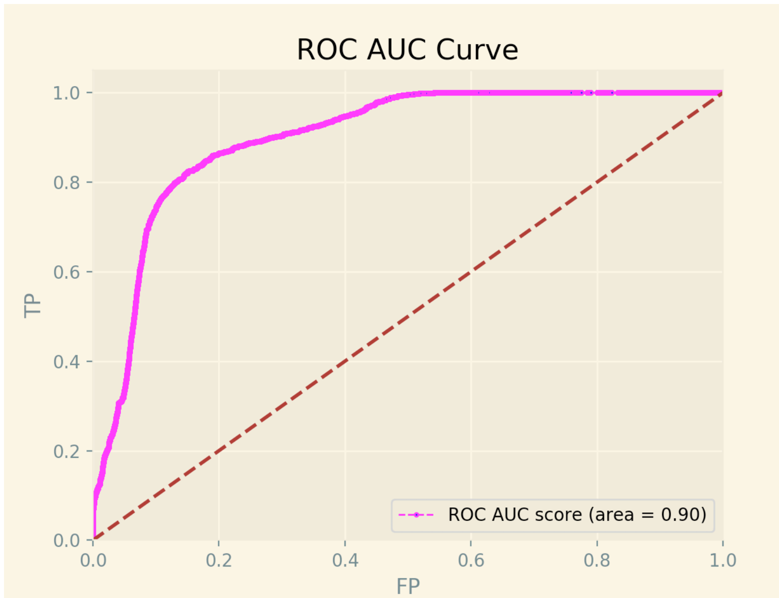

Figure 29 ROC AUC Curve Logistic Regression (Stochastic Gradient) Model

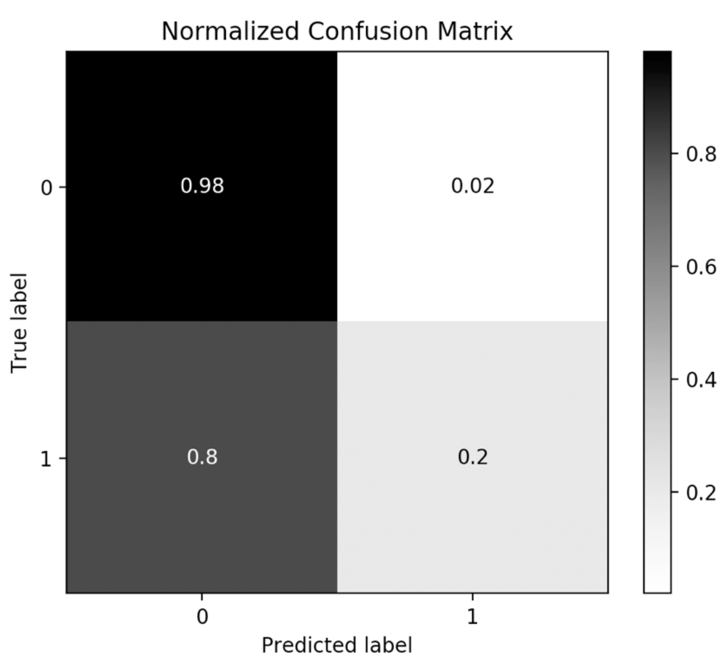

Figure 30 Normalized Confusion Matrix Logistic Regression (Stochastic Gradient) Model

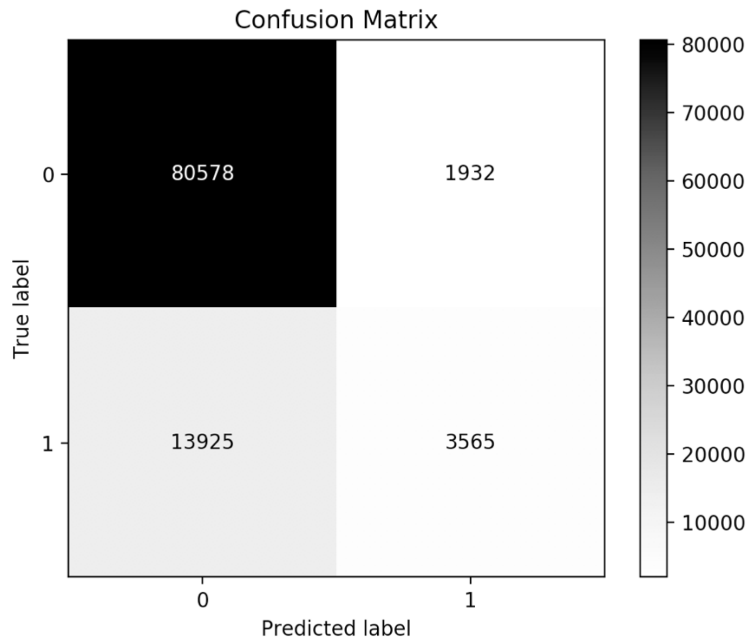

Figure 31 Confusion Matrix Logistic Regression (Stochastic Gradient) Model

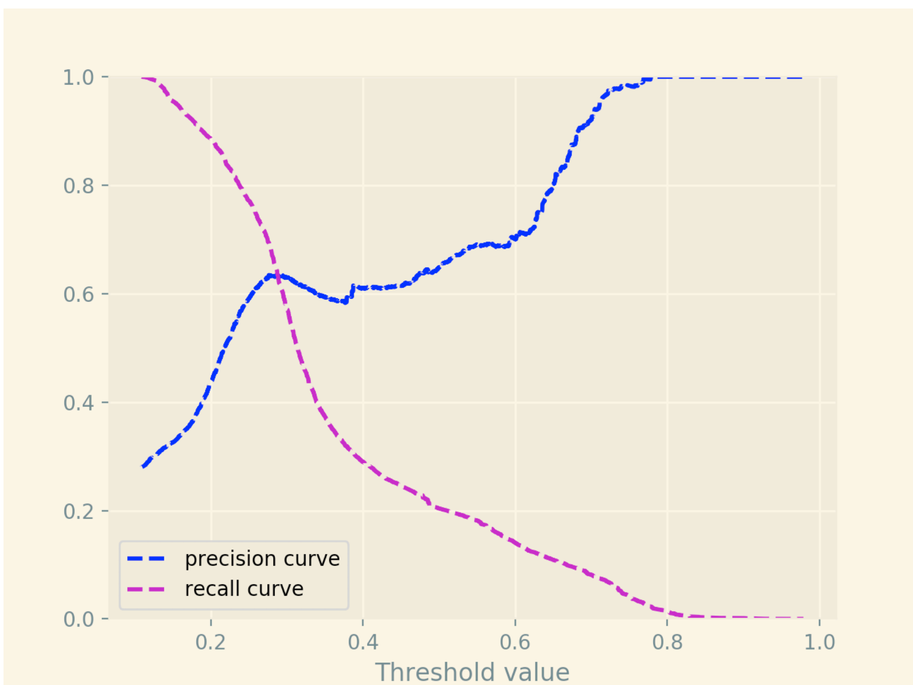

Figure 32 Precision Recall Curves Logistic Regression (Stochastic Gradient) Model

When using each of the above discussed classifier for the prediction, we see that all the classifiers predict the non-clicks better than the clicks due to the imbalance in the data. Additionally, it is also observed that data points are the mis predicted by one model is predicted rightly by other models. This gives a necessity to use the approach 2 which is the ensemble model of all the classifiers seen above.

7.1.2 Approach 2

In this approach, we created an ensemble of all the above models using the weighted technique and the voting scheme. The weighted technique proved to be working well with the imbalance in the dataset. The logistic regression with stochastic gradient descent model is weighted [18] more than all the other models while the second weight being given to the random forest model and all the other models were tested with equal weights. The result of one such experiment is captured below. The ROC AUC score is 0.91 which is the highest overall.

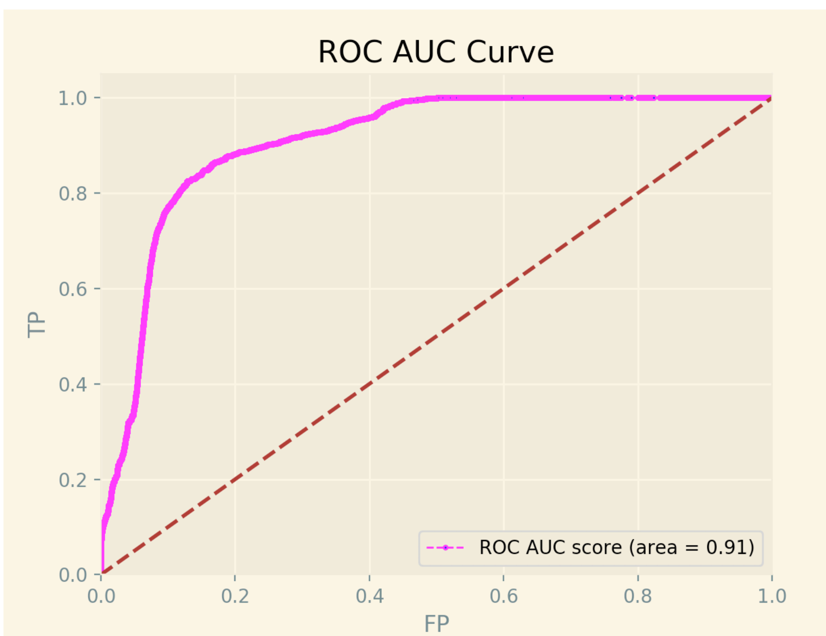

Figure 33 ROC AUC Curves Ensemble Model

7.1.3 Approach 3

With all the above approaches, the model is able to classify a non-clicked advertisement as 0 correctly (True Negative) but predicts most of the clicked advertisement as 0 too, this means that the model has a good precision value but has to improve with the false negatives. This is certainly due to the imbalance in the data set between the two class of target variables. So, the oversampling of the target variable 1 is done and by this both categories of the target are made even. The results of this experiment are captured. The confusion matrix clearly indicates that the model has now performed well on the whole [11][17].

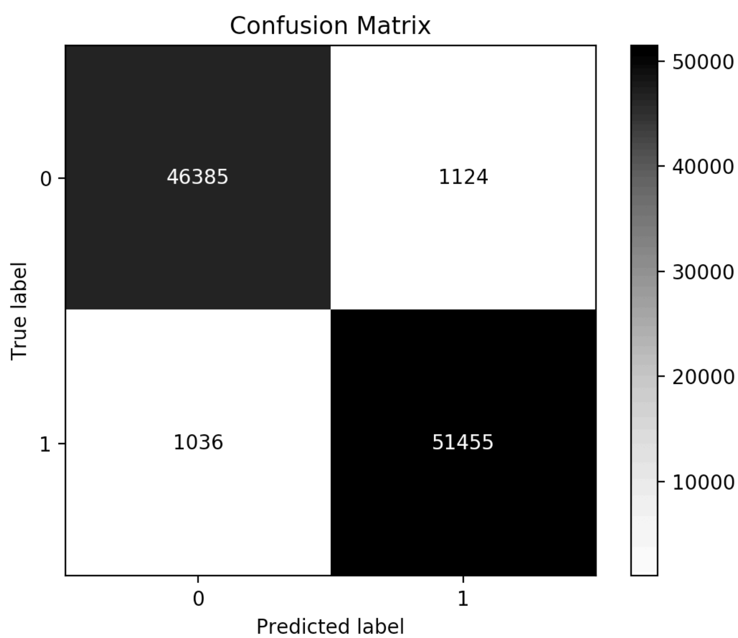

Figure 34 Confusion Matrix Over Sampled data

7.2 Comparison of the approaches

The numbers shown in the table are from the accuracy of the machine learning models that are shown in the confusion matrix and the ROC- AUC plots included in the sections 7.1.1 to 7.1.3.

Model	Accuracy score (+/-0.03)	ROC AUC	Confusion matrix
Decision Trees	0.833	0.79	Works well for True negatives, but misclassification with true positives.
Logistic regression (Online)	0.835	0.81	Works well for True negatives, but misclassification with true positives.
Random Forest	0.838	0.84	Works well for True negatives, but misclassification with true positives.
XGBoost	0.8383	0.84	Works well for True negatives, but misclassification with true positives.
Logistic Regression (Stochastic)	0.841	0.9	Works well for True negatives, but misclassification with true positives.
Ensemble	0.845	0.91	Works well for True negatives, but misclassification with true positives.
Over sampling – Ensemble	0.85	0.91	Reduced error rate

Table 2 Comparisons of results

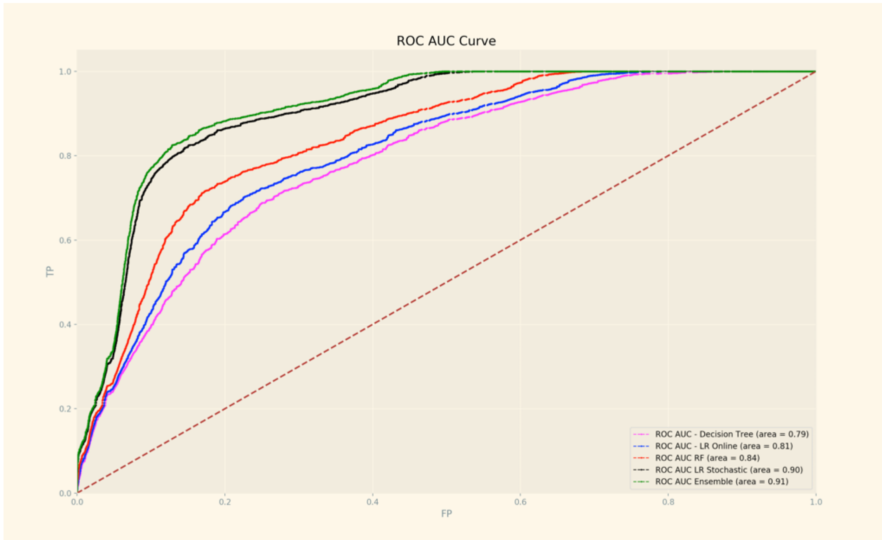

Figure 35 ROC AUC Curves Comparison

7.3 Observations

The following observations are made in this study.

7.3.1 Observation 1

The dataset has an even distribution of data among all the days with imbalance in the target category. Upon comparing the results of the above approaches, the logistic regression with stochastic gradient and the random forest model has improved AUC value and hence these models were given more weights in the ensemble experimentation. The overall AUC is 0.91 which is better than the other similar researches [4][14]. But, the model is highly dependent on the categorical anonymous variables and does not work well with the imbalance in the data [15]. The white papers from advertising platforms like the Google and the Alibaba report better results in terms of the precision and recall. The reason could be that they have complete access to the data with no anonymous variables.

7.3.2 Observation 2

The ensemble model when trained using the over sampled data gave the overall best result. The false negatives are only 1036 upon 50000 positive data samples and false positives are only 1124 out of 50000 negative data samples. This keeps the misclassification error rate to minimum relatively.

7.3.3 Observation 3

The performance of the ensemble model with voting is diminishing when compared to the weighted scheme. This is due to the reason a lot of mis classification occurs when we round off the probability. Instead, we chose the weighted ensemble technique which gave the best results.

7.3.4 Observation 4

For the ensemble, we randomly sampled the data for each of the classifier and results were observed to be consistent. Hence, the ensemble technique with over sampling to get balanced data is the overall winning approach in this study.

CHAPTER 8

CONCLUSION AND FUTURE WORK

The rapid growth of the social media as the online advertising platform has made it compelling for the researches to focus on improving the misclassification rate in the prediction of the click through rate of an advertisement. This will help the advertiser choose the right set of attributes to target the potential audience for their business. The accuracy can also help the advertising platforms to decide on the cost of the advertisement.

In this research, the steps from preparation of the dataset to the building of the classification model were discussed in detail. The results of each step were also discussed and bench marked against the base lines. A series of experiments were also performed and the results were captured and compared. The initial study was hard due to the enormous amount of data at hand. Nevertheless, the feature engineering techniques and sampling strategies were some of the key parameter tuning steps that helped improve the overall performance of the research.

As a next step, this study can be improved in terms of the efficiency making it more usable. Given access to all the anonymized variables, noise in the data can be perfectly removed. We could tune the parameters wisely which could help us solve the more complex problem of choosing the right advertisement at the right time for a user.

REFERENCES

[1] P. Chan, X. Hu, L. Zhao, D. Yeung, D. Liu and L. Xiao, "Convolutional Neural Networks based Click-Through Rate Prediction with Multiple Feature Sequences", Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, 2018.

[2] P. Barford, I. Canadi, D. Krushevskaja, Q. Ma and S. Muthukrishnan, "Adscape", Proceedings of the 23rd international conference on World wide web - WWW '14, 2014.

[3] G. Zhou, K. Gai, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin and H. Li, "Deep Interest Network for Click-Through Rate Prediction", Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '18, 2018.

[4] C. Jie-Hao, L. Xue-Yi, Z. Zi-Qian, S. Ji-Yun and Z. Qiu-Hong, "A CTR prediction method based on feature engineering and online learning", 2017 17th International Symposium on Communications and Information Technologies (ISCIT), 2017.

[5] S. Zhang, Z. Liu and W. Xiao, "A Hierarchical Extreme Learning Machine Algorithm for Advertisement Click-Through Rate Prediction", IEEE Access, vol. 6, pp. 50641-50647, 2018.

[6] H. Guo, R. Tang, Y. Ye, Z. Li and X. He, "Deep FM: A Factorization-Machine based Neural Network for CTR Prediction", JOURNAL OF LATEX CLASS FILES, VOL. 14, NO. 8, AUGUST 2015 11.

[7] M. Freser, D. Kosir, "Analysis of the open advertising data set", 6th Jožef Stefan International Postgraduate School Students' Conference, Pg. 111-120, Volume: 1, May 2014.

[8] Xiangyang She, Shaopeng Wang,"Research on Advertising Click-Through Rate Prediction Based on CNN-FM Hybrid Model", 2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics, Year: 2018 , Volume: 02, Page s: 56 – 59.

[9] Jing Ma, Xian Chen, Yueming Lu, Kuo Zhang" A click-through rate prediction model and its applications to sponsored search advertising", International Conference on Cyberspace Technology, Year: 2013, Page s: 500 – 503.

[10] Antriksh Agarwal, Avishkar Gupta, Tanvir Ahmad "A comparative study of linear learning methods in click-through rate prediction", 2015 International Conference on Soft Computing Techniques and Implementations, Year 2015, Pages 97-102.

[11] Feng Zhou, Hua Yin, Lizhang Zhan, Huafei Li, Yeliang Fan, Liu Jiang "A Novel Ensemble Strategy Combining Gradient Boosted Decision Trees and Factorization Machine Based Neural Network for Clicks Prediction", 2018 International Conference on Big Data and Artificial Intelligence, Year 2018, Pages 29-33.

[12] Qianqian Wang, Fang'ai Liu, Shuning Xing, Xiaohui Zhao, Tianlai Li "Research on CTR Prediction Based on Deep Learning", IEEE, Year: 2018, Page 1-1.

[13] Xiaokang Qiu, Yuan Zuo, Guannan Liu, "ETCF: An Ensemble Model for CTR Prediction", 2018 15th International Conference on Service Systems and Service Management, year 2018, pages 1-5.

[14] Yuyu Zhang, Hanjun Dai, Chang Xu, Jun Feng, Taifeng Wang, Jiang Bian, Bin Wang, and Tie- Yan Liu. "Sequential click prediction for sponsored search with recurrent neural networks.", *AAAI*, Conference on Artificial Intelligence, Year 2014.

[15] Weinan Zhang, Tianming Du, and Jun Wang. "Deep learning over multi-field categorical data - - A case study on user response prediction", Year 2016, European Conference on Information Retrieval.

[16] Jing Ma, Xian Chen, Yueming Lu and Kuo Zhang, "A click-through rate prediction model and its applications to sponsored search advertising," *International Conference on Cyberspace Technology (CCT 2013)*, Beijing, China, 2013, pp. 500-503.

[17] R. Kumar, S. M. Naik, V. D. Naik, S. Shiralli, Sunil V.G and M. Husain, "Predicting clicks: CTR estimation of advertisements using Logistic Regression classifier," *2015 IEEE International Advance Computing Conference (IACC)*, Banglore, 2015, pp. 1134-1138.

[18] B. Tan and R. Srikant, "Online Advertisement, Optimization and Stochastic Networks," in *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2854-2868, Nov. 2012.

[19] Q. Cheng, P. K. Varshney and M. K. Arora, "Logistic Regression for Feature Selection and Soft Classification of Remote Sensing Data," in *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 4, pp. 491-494, Oct. 2006.

[20] “ROC AUC Curve”, 2019, Available online at https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5

[21] “Precision vs Recall curve”, 2019, Available online at https://towardsdatascience.com/beyond-accuracy-precision-and-recall-3da06bea9f6c

[22] “ROC Curve for accuracy” 2019, Available online at http://fastml.com/what-you-wanted-to-know-about-auc/

[22] “Scikit Learn for classification models”, 2019, Available online at http://benalexkeen.com/scoring-classifier-models-using-scikit-learn/

[23] “ROC and Precision Recall Curves” , 2019, Available online at https://machinelearningmastery.com/roc-curves-and-precision-recall-curves-for-classification-in-python/

[24] “The Scikit Learn Documentation”, 2019, Available online at https://scikit-plot.readthedocs.io/en/stable/metrics.html

[25] “Seaborn Plots”, 2019, Available online at https://seaborn.pydata.org/tutorial/relational.html#relating-variables-with-scatter-plots

[26]“The Matplotlib documentation”, 2019, Available online at https://matplotlib.org/api/_as_gen/matplotlib.pyplot.plot.html

[27] Zhu J., Guan Q., Zhao X. (2015) Multi-class JPEG Image Steganalysis by Ensemble Linear SVM Classifier. In: Shi YQ., Kim H., Pérez-González F., Yang CN. (eds) Digital-Forensics and Watermarking. IWDW 2014. Lecture Notes in Computer Science, vol 9023. Springer, Cham.