COMPGW02: Web Economics Project

Individual Report

Gerard Cardoso Negrie
MSc Business Analytics (with Computer Science)
ucabgc0@ucl.ac.uk

ABSTRACT

Real Time Bidding is quickly becoming one of the most popular methods of buying and selling advertisements online. This has called for many researchers to look at the factors that make advertisements more valuable, and strategies that can outperform other demand service providers online. This paper investigates online advertising bidding and data collected from millions of advertisement bids across multiple exchanges. The data set provided is split into a train, validation and test set, with each containing details on many advertising auctions that have happened. First, this paper will explore the data finding out the factors associated with users clicking ads, making the ad slots more valuable. Then, I will explain the bidding strategy created using Random Forest Classifiers that outperforms constant bidding strategies.

1. INTRODUCTION

The objective of this project is to bid for online advertisements and create the best strategy to optimize, amongst other things, the click through rate of users. The click through rate is defined as the percent of impressions that are converted into clicks. It is important to understand which users and advertisements will lead to a higher possibility of clicks, and thus should be paid a higher price for. This is the essence of real-time bidding (RTB). For an introduction to advertisement exchanges and RTB, refer to Real-Time Bidding Benchmarking with iPinYou Dataset paper [2]. The type of auction that is done in advertisement exchanges today is known as a Vickrey auction, where the highest bid wins the auction, but the winner only pays the price of the second highest bid.

Before the team began to create bidding strategies and click prediction models, it was important to understand the data provided to us. In this paper, I will give an overview of the data, with an explanation of the pre-processing and exploration phases carried out in the project. Following this, I will explain my own RTB strategy which was used in the final strategy submitted by the team.

2. RELATED WORK

Despite Real-Time Bidding rising so much in popularity, there has been up to now little support to the community for research efforts. iPinYou, a Chinese advertising company, released a dataset that was used in its global RTB algorithm competition in 2013. This dataset has enabled researchers to conduct various experiments and analyses for both Click-Through Rate (CTR) estimation and bid optimisation [2]. The authors of this carry out a very similar project to what we are aiming to do, with some data exploration and click prediction models being created.

As presented in Section 6 of this paper, the data we are dealing with is very imbalanced, as the ads that are clicked by the users only make up a very small portion of the overall impressions. To tackle such issues, techniques such as undersampling are commonly used, in order to create more balanced datasets [1]. I will be using under-sampling techniques, described in Section 6, to help improve the click prediction model.

3. DATA OVERVIEW

The data sets for this project were provided by UCL. The data was split into a training, validation and test set, where the test set is used for final project evaluation and grading purposes. Exploration will only be done on the training data, as the training and validation data are equivalent in structure and very similar in values. The validation data will only be used to validate our models.

The training data contains 2,697,738 rows with each row representing one auction. Each row contains details of the advertisement slot, the user and the bidding side. An example of a row contained in the training data can be seen in Table 1. Note that all money and cost values are expressed in the units of Chinese Fen in Cost Per Mille format.

Most of the data contains IDs of certain pages and slots that allow Demand Service Providers to recognize the features and qualities of the slot being auctioned. While the data exploration section will look more at slot factors and their effect on CTR, my initial hypothesis is that the identifiers will have an impact on the final click prediction model.

4. DATA PRE-PROCESSING

Before continuing with the exploration and modelling of the data, it is necessary to ensure that the values contained in the data are validated, and that the data is in its cleanest form. The data cleaning steps that were carried out were to remove error rows where the pay price is higher than the bid price (33,579 rows, 1.2% of data), and remove the log type (only contains value 1) and URL ID (All values are null) columns.

Furthermore, there are opportunities to add more information to this data through careful feature engineering. For example, the user agent field currently contains both the OS and browser of the user, so this can be split into two fields, browser and OS, which can be analysed separately. Simi-

larly, a slot dimension field can be created from slot width and slot height. The thought behind such a field is that there are different format advertisement slots online, such as banners or side ads. It might usually be the case that a larger slot height would cost more but, at the same time, banners might cost more than side ads even though having a smaller slot height. The slot dimension field will capture this in the final model, discriminating between all the different shapes of ads.

Field	Example	Description
Click	0	Ad clicked = 1
Weekday	1	Day of the week (0-6)
Hour	14	Hour of the day (0-23)
Bid Id	fdfae67...	ID of bid for that auction
Logtype	1	Describes the type of row. Always 1 in our data.
User Id	u_Vh1OP...	ID of the user who advertisers are bidding on
User Agent	windows_ie	The OS and browser being used by user
IP	180.107.112*	IP address of user
Region	80	Region of the User
City	85	City of the user
Ad Exchange	2	Ad exchange where auction is taking place
Domain	trqRT...	ID of the website domain where ad is displayed
URL	d48a96...	ID of the URL where ad will be displayed
URL Id	null	All values are null
Slot Id	43328...	ID of the ad slot
Slot Width	468	Width of ad slot in pixels
Slot Height	60	Height of ad slot in pixels
Slot Visibility	1	Level of visibility of ad
Slot Format	0	Slot type
Slot Price	5	Minimum price to pay for slot
Creative	61259...	ID of the ad design used by advertiser
Bid Price	300	Price of the winning bid for the auction
Pay Price	54	Price paid by auction winner (second highest bid)
Key Page	bebefa...	ID of the Page
Advertiser	1458	ID of the advertiser winning the bid
Usertag	13866,10063	Tags describing user features

Table 1: Data Set Field Overview

5. DATA EXPLORATION

Ultimately, advertisers want to know that they are spending money in the most efficient way possible, maximizing the number of clicks which would hopefully lead to maximizing conversions. Essential to this goal, is understanding the effect of different factors on the click-through rate of users. Below I will look at any patterns or effects of different ad factors on the CTR.

Table 2 displays summary statistics of the data. It is immediately clear from these values that obtaining a click from a user is incredibly rare. Impressions still deliver some value in the form of spreading the brand, however clicks are the real goldmine. Obtaining clicks is so rare, that the effective cost per click becomes so high as seen in the Table. The small amount of clicks will complicate matters when trying to predict if an ad will be clicked on using machine learning methods. Tools to solve this are detailed in Section 5.

Statistic	Value
Number Impressions	2,664,159
Number Clicks	1,986
Average CTR	0.075%
Average CPC	$104830.47

Table 2: Summary Statistics

5.1 Browser

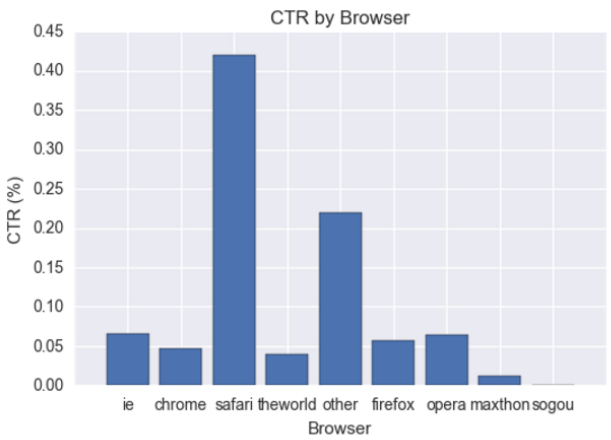

Figure 1: Click-Through Rate by Browser Types

Figure 1 displays the click-through rate for different browsers that are being used. The largest click-through rates come from Safari and Other browsers, while the lowest click-through

rates come from Maxthon and Sogou. While Safari has by far the largest CTR, it only accounts for 4% of browsers in the training date, with IE (60.03%) and Chrome (32.95%) being the most used. This confirms our previous hypothesis that mobile platforms lead to higher click-through rates since the browsers used on Android and iOS have high CTR.

5.2 OS

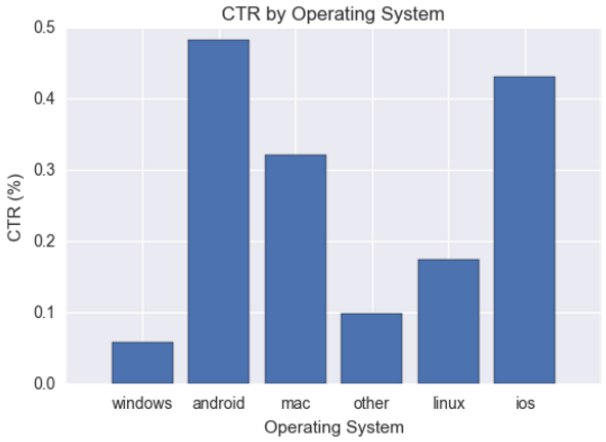

Figure 2: Click-Through Rate by OS Types

Looking at Figure 2, which shows the CTR by operating systems, we can see that the highest click-through rates originate from mobile platforms, namely Android and iOS. This is most probably due to the ease of accidental ad clicking on mobile platforms, or higher proportion of space ads take up on mobile compared to other content on the screen. Thus, iOS, Android, and other platforms have the highest click-through rates. However, iOS and âĂIJotherâĂİ platforms only account for a very small number of the overall data (0.017% and 0.075% respectively) and Android only accounts for 2.62% of the data. Windows OS is present in 95.25% of the auctions, so while mobile operating systems have higher click-through rates, the vast majority of impressions and clicks still come from Windows. It is likely that browsers and operating systems are closely linked in the effect that they have on CTR. As such, it is probable that, much like mobile operating systems have higher click-through rates, browsers used on mobile will also have high click-through rates.

5.3 Weekday

Figure 3 shows the difference in click-through rates by the day of the week. It appears as though mid-week click-through rates are higher than during the weekend, but overall the difference is not too significant, and we cannot conclude that weekday has an effect on the click through rate.

5.4 Hour

Figure 4 displays the click-through rates stratified by the hours of the day. There are several fluctuations in the CTR, but the highest CTR comes from the periods in the early and late evening. This is likely due to more leisurely browsing during those hours which would lead to more clicks than otherwise. Furthermore, the lower CTR is during working

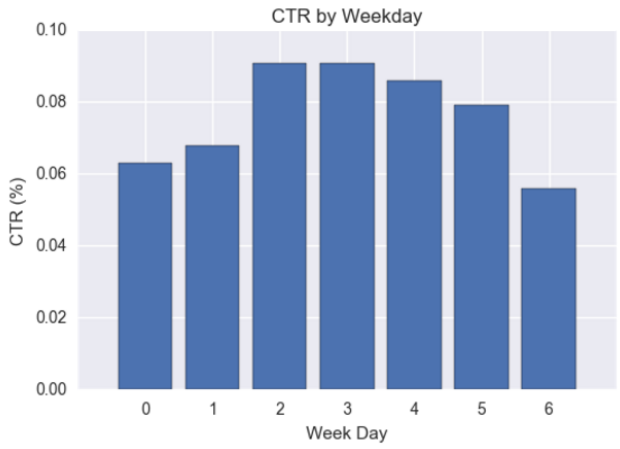

Figure 3: Click-Through Rate by Day of the Week

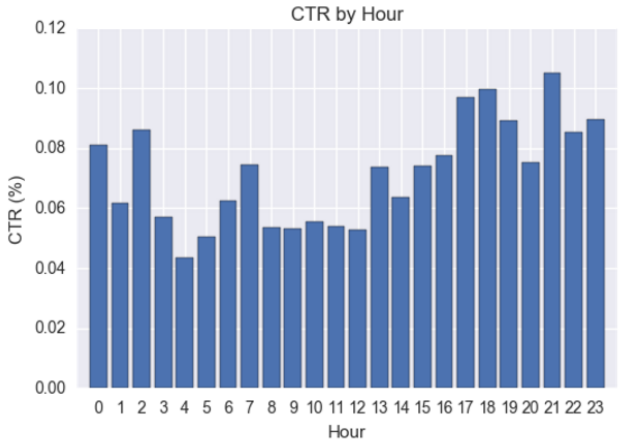

Figure 4: Click-Through Rate by Hour of Day

hours, likely when users are less active browsing and would not be as likely to click on an advertisement at work.

5.5 Slot Dimensions

The slot dimension feature was created as described in the data pre-processing section above, by combining the slot width and slot height to obtain the unique slot shapes in the training data. From these, we can see different ad formats such as mobile ads, banners and side ads and it is interesting to see how they affect the CTR.

Figure 5 shows this, and as you can see, mobile ads have the highest CTR once again as it is all linked to mobile platforms having a higher CTR. The lowest

6. BIDDING STRATEGY

Initially, the team experimented with constant and random bidding strategies to evaluate how effective bidding strategies they are. The results for these bidding strategies can be found in Table X. There exist severe limitations with constant and random bidding strategies, mainly that they do not discriminate between auctions where the slot being auction may have a much higher prior preference for being clicked.

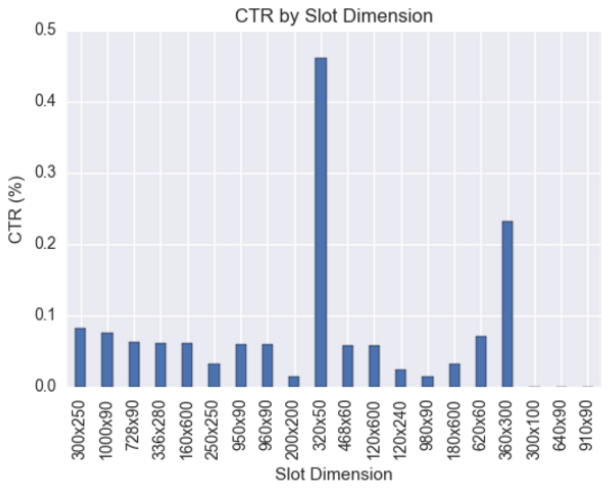

Figure 5: Click-Through Rate by Slot Dimensions

For my best choice bidding strategy, I chose to use a slight alternative to the linear bidding strategy. The linear bidding strategy uses a click prediction probability to determine the bid price, however I will be using the click prediction itself and combine both logistic regression and random forest models. The formula for the bid price in my strategy is:

$$bid = base * (lr_pred + rf_pred)$$

where base is the base bid that will be optimized and tuned on the validation set. Thus, if both the logistic regression and random forest predict that a row is a click, then the base bid is doubled because we are much more certain of that fact. However, if neither of the models predict a click, then the bid price becomes 0.

Feature Engineering

The feature engineering carried out for my bidding strategy & click prediction model is the same one as described in the group report, mainly one hot encoding of the user tags, imputing any missing values and, finally, label encoding the categorical variables in order to use them in the model. Also, the previous features made in the pre-processing section of the report were also included in the model, which are slot dimension, browser, and operating system. The first choice to make, was to either use a random forest model or logistic regression model, which are both widely used classifiers in the machine learning world.

Sampling

One major problem of creating a classifier model with the current training data set at hand is that there is some extreme class in-balance, in that there is only approximately 2000 clicks in a data set of over 2 million rows. What this leads to, is that classifiers will prefer to predict non-click for most, if not all, rows, because that will still obtain a high accuracy. However, that does not suit our objective where we are trying to accurately discriminate between what could potentially be a click and what is definitely not going to be clicks. To get around this, we use under-sampling methods, and for this project I chose to go with random under-sampling, where I take all the clicks in the dataset and take a random sample of non-clicks of the same size as the number of clicks. This 50/50 data is used as the training data for the model.

Model Validation and Hyper Parameter Optimization

In order to evaluate our models, there are a variety of metrics that can be used such as precision, recall, F1 score. However, the two metrics that I will use to evaluate my models are the true positive rate, and the false positive rate. The reason for this, is that I want to increase the certainty that when the model predicts a click, there is a high chance of it being a click. Otherwise, if the model creates too many false positives, too much money will be spent bidding on ads that will not be clicked on.

For hyperparameter optimization, I use 3-fold cross validation with search over a defined parameter space. This is only done for the regularization parameter in logistic regression.

Table 3 and Table 4 shows us the confusion matrices of predictions obtained from random forest and logistic regression models on the validation set. These show how the models are similar in the performance, with random forest having a stronger true positive rate but a weaker false positive rate.

	No Click	Click
No Click	255,470	44,053
Click	73	153

Table 3: Logistic Regression Validation Confusion Matrix

	No Click	Click
No Click	252,015	47,508
Click	57	169

Table 4: Random Forest Validation Confusion Matrix

Model	Cost	Imp	Clicks	CTR	CPC
Constant	\$6250.14	96480	70	0.07%	\$89.29
RF	\$6250.04	136311	123	0.09%	\$50.81
LR	\$6250.01	130488	118	0.09%	\$52.97
Combined	\$5668.50	63442	172	0.27%	\$33.07

Table 5: Evaluation of Strategies at 6250 CNY fen budget

Table 5 displays the results from a constant strategy, linear strategy with random forest, linear strategy with logistic regression, and my combined final strategy with 6250 CNY fen budget. My strategy base bid was optimized using the same simulation method as discussed in the Constant and Random bidding strategy section of the group report. It is evident that my new combined strategy outperforms the previous strategies with regards to clicks, achieving a three-fold click through rate and a much lower cost per click. However, if an advertiser prioritises Impressions then my strategy is weaker since it cherry picks the auctions that will most likely lead to clicks. The strategy allows us to bid on the more valuable users, delivering overall higher value for money to advertisers

For the final group strategy, we have chosen to use a similar strategy but combining more models. More details on this can be found in the group report.

7. CONCLUSION

This paper has looked at a dataset of online ad auctions to try to determine the main factors involved in predicting user click behaviour, and create a good strategy. I found that mobile users tend to have a higher click through rate, and the time of day has a significant impact on the click through rate with the evening leading to more clicks. Following this, I created a variety of the classic linear bidding strategy using a click prediction model. Logistic Regression and Random Forest classifiers were both used in this strategy. To create the model, under-sampling was done on the training data so that the model could accurately discriminate between clicks and non-clicks.

Overall, my bidding strategy clearly outperformed the constant bidding and linear bidding strategy with individual models, by having many more clicks, with a similar budget. However, this strategy could be improved through better under-sampling methods such as doing an ensemble of models with more random sampling or even looking at using nearest neighbour methods to determine similar data points to the click data so that the model improves its ability to discriminate. Furthermore, one could look at using better methods to combine the predictions of random forest and logistic regression using weights in an efficient way and using the click probabilities of several models to create a better bid price.

7.1 Group Contribution

My role within the group for this project was to create my individual model and brainstorm how we could effectively combine our models to create our final bidding strategy. Furthermore, I provided some feature engineering efforts for the group.

Overall I felt the work was equally distributed amongst the group and everyone worked on their sections well individually, but also contributed to the group when we needed to collaborate on different sections of the project.

8. REFERENCES

[1] X.-Y. Liu, J. Wu, and Z.-H. Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550, 2009.

[2] W. Zhang, S. Yuan, J. Wang, and X. Shen. Real-time bidding benchmarking with ipinyou dataset. *arXiv preprint arXiv:1407.7073*, 2014.