Web Economics Project 2017 (Individual Report)

Chun Siong Poh
University College London
School of Computer Science
chun.poh.16@ucl.ac.uk

1. INTRODUCTION

The trend of online advertising is fast evolving. Business Insider Intelligence estimated that Real-Time Bidding (RTB) would grow from 16% in 2013 to 33% in 2018[Hoelzel]. In RTB, a typical click through rate is 0.01% and it is in the interests of advertisers to improve it. In this coursework, iPinYou dataset is used to evaluate a RTB bidding strategy. The bidding strategy comprises of a utility and cost estimator to estimate the probability of click through and price to bid respectively. Gradient Boosted Trees is used to model the click through probability. It achieved 0.875 AUC score on the validation set. Advertisers may have different campaign objectives such as to reach a huge number of customer within a short period (i.e. Clicks) or achieving greater value for their money (i.e. CTR, CPM). A bidding strategy is developed and demonstrated to suit such objectives through parameter tuning. The code and report is uploaded and available in Github[1].

2. LITERATURE REVIEW

In RTB ecosystem, advertiser or DSP aims to buy the best-matched impressions that suits their objective via real time auction. To determine the best-matched impressions, models analyses the contextual and historical features for a given impression to determine the click through probability and estimate its worth before submitting a bid all within 10-100ms.

In utility estimation, literature covers machine learning [Perlich et al. 2012] [Zhang et al. 2014] [Lee et al. 2012], collaborative filtering [Robinson 1997] [Menon et al. 2011] [Zhang et al. 2016] and clustering [Regelson and Fain 2006]. Machine learning has been gaining popularity and one learning method is gradient boost trees used in [Zhang et al. 2014] [?] for utility estimation. Gradient boost trees is an ensemble method where each tree is an expert on the errors of its predecessor. XGBoost is an open source implementation of gradient boosted trees that demonstrated state of art results in many domain and achieves high performance[Chen and Guestrin 2016].

In cost estimation, Claudia et al. proposed an aggressive strategy that doubles the base bid when the model estimates exceed a predefined step function and obtained favourable result from Media6Degrees dataset [Perlich et al. 2012]. Typically, in RTB second price auction is employed. In second price auction truth telling is the dominant strategy. However, in a multi-auctions environment with fixed budget truth telling may no longer be a dominant strategy[Wang et al. 2016].

[1]Github URL: https://github.com/jax79sg/webecon2017

3. APPROACH AND RESULT

3.1 Data Exploration

The dataset for this coursework originated from iPinYou Information Technologies Co., Ltd. It was used for their global RTB algorithm competition in 2013. In its original form, the dataset contains 4 log types, bids, impressions, clicks and conversions. For the purpose of this coursework, the dataset provided is limited to impressions log type (i.e. type 1) and is split into 3 files for training, validation and testing.

Two logical errors are detected with the affected rows removed.

- Pay price > bid price (33579 rows 1.2%)
- Slot price > pay price (13413 rows 0.5%)

Table 1 shows the statistical summary of train.csv with erroneous rows (46992 rows 1.7%) removed. Statistical summary for Validation.csv is shown in Table 2. Conversions rates is not computed due to insufficient data in the dataset.

Adv	Imp	Clicks	Cost	CTR	CPM	eCPC
1458	534599	446	36790	0.000834	68.82	82.49
2259	145845	44	13579	0.000302	93.11	308.63
2261	120328	37	10749	0.000307	89.34	290.54
2821	231015	144	20578	0.000623	89.08	142.91
2997	53542	248	3338	0.004632	62.36	13.46
3358	289290	203	24359	0.000702	84.20	120.00
3386	491570	356	37904	0.000724	77.11	106.47
3427	439414	323	33212	0.000735	75.58	102.82
3476	342023	173	26280	0.000506	76.84	151.91
Total	2647626	1974	206789	0.000745	78.10	104.75

Table 1: Dataset Statistics for Train.csv

From Table 1, Advertiser (Adv) 2997 stands out with 0.46% click through rate and 13.46 cost per click. The CTR is about 5 times better than the next highest Adv 1458. On average across train and validation dataset, CTR is 0.07% with a cost per click of 105.17. These statistics serve as the baseline performance for the model to exceed.

Figure 1 compares the CTR of Adv 2997, 1458 and average of all advertisers against a variety of features. On slot visibility, first view commands the highest CTR probability

Adv	Imp	Clicks	Cost	CTR	CPM	eCPC
1458	59940	50	4113	0.000834	68.63	82.27
2259	16415	11	1518	0.000670	92.51	138.04
2261	13365	5	1194	0.000374	89.40	238.96
2821	25620	16	2277	0.000625	88.91	142.37
2997	6034	26	387	0.004309	64.20	14.90
3358	32137	26	2705	0.000809	84.20	104.07
3386	55097	32	4225	0.000581	76.70	132.06
3427	48781	40	3682	0.000820	75.49	92.06
3476	38322	13	2931	0.000339	76.49	225.49
Total	295711	219	23032	0.000740	77.88	105.17

Table 2: Dataset Statistics for Validation.csv

and Adv 2997 has 88% of their impression using it. Slot visibility refers to the visibility or prominence of the advertisement [12]. iPinYou may have changed their slot visibility enumeration from "0, 1, 2,...,255" to "firstview, secondview, ..." between the season. The categories should be combined if the mapping behind the categories are available.

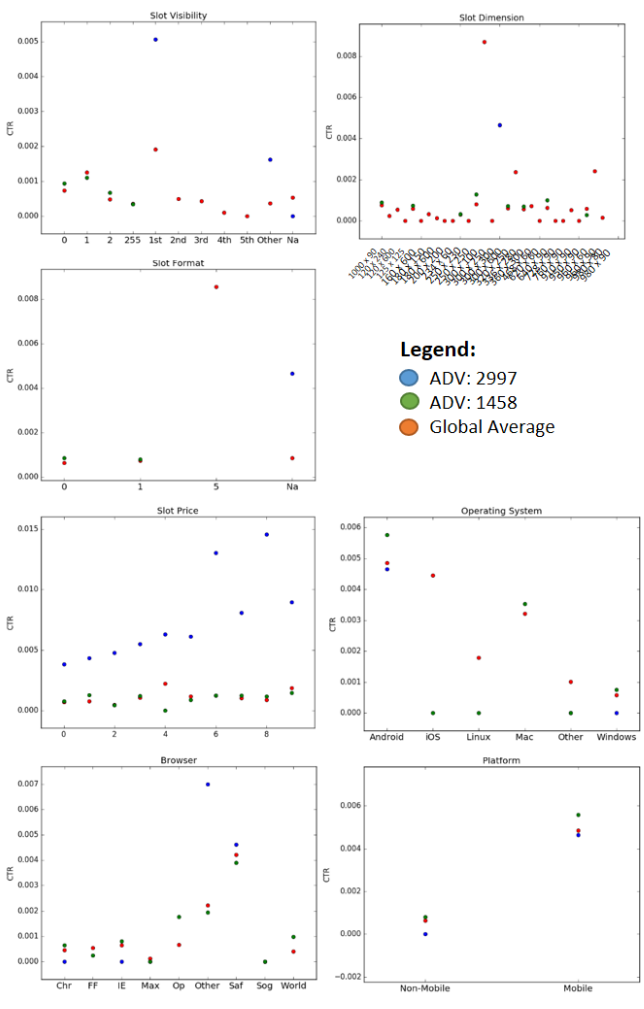

Figure 1: CTR distribution on various features

On the slot dimension, there is 29 combinations. Dimension 300x250, used by 25% of all impressions, stands out among the rest achieving the highest CTR. Interestingly, Adv 2997 is observed to use dimension 300x600 only. Slot format consists of 0, 1, 5 and Na. Format 5 achieves significantly higher CTR over the rest of the format. Slot price is the reserve price for the auction. The slot price is divided into buckets of 20 (i.e. [0-20], [21, 40],âĂę, [181, infinite]). In general, higher reserve price does not translate to higher CTR. However, Adv 2997 is observed to achieve higher CTR when reserve price is higher. Looking at operating system, browser and platform, there are evident higher CTR when advertisement are on the mobile platform. Interestingly, Adv 1458 achieves near-zero CTR when their advertisements are delivered to iOS. Adv 2997, who falls under mobile e-commence app industrial category, chooses to buy only impression for Android indirectly reveals that their customer base are Android users.

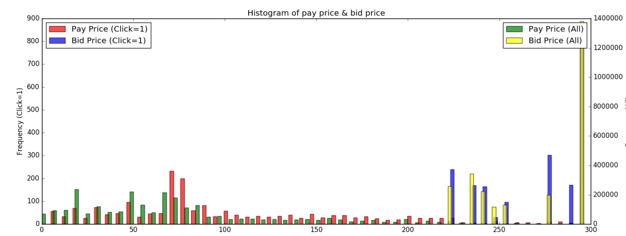

Figure 2: Histogram of pay price and bid price

Lastly, Figure 2 shows the pricing relationship between impression with and without click through. One obvious bidding trend is that all winners have submitted bids above 220 while second bid is on average around 78. The huge difference shows that everyone estimates the value of each impression vastly different. The same phenomenon is also observed in oil drill rights [Capen et al. 1971] where bids varies drastically. It is also noted there are rarely 2 high bids for a single impression. For a real-world bidding model, it would need to bid above 220 to stand any chance in winning an impression and advertiser has to be prepared to pay above 220 as the current set of winner would be delegated to second price.

3.2 Bidding Model

3.2.1 *CTR Estimation*

XGBoost is a battle tested gradient boosted trees popularised by Kaggle competition. For the Click Through Rate (CTR) estimation I have chosen XGBoost for its speed and to gain experience in boosted trees model.

A total of 102 features are used to train the model. Data such as width and height are combined to form dimension while data such as user agent are decomposed to OS, browser and mobile/non-mobile platform. User tags are split into binary one-hot fashion to indicate the presence or absence for a particular tag. All categorical data (i.e. region, city, slotformat,âĂę) are encoded into integer representation while continuous data (i.e. slotprice, weekday, hour,âĂę) are used as it is. The occurrence frequency of categorical data is also computed and used as features.

The model is trained using a pruned training set with 20% click through rate and validated using the held out set provided. Errors found in section 3.1 are removed from the train and validation set. XGBoost contains up to 15 tuning

Steps	Parameters
Step 0	Default value - Booster: gbtree - Lambda: 1 - Alpha: 0 - Eta: 0.3 - base_score: 0.5 - scale_pos_weight: 1 - scoring: roc_auc - objective: binary:logistic
Step 1	- max_depth: 5 - min_child_weight: 1
Step 2	- gamma: 0
Step 3	- subsample: 0.55 - colsample_bytree: 0.8
Step 4	- reg_alpha: 0.05
Step 5	- learning_rate: 0.042

Table 3: CTR Model Tuning

parameters and to obtain a good model tuning them is critical. Using sklearn grid search with 5-fold cross validation, the model is progressive tuned over 5 steps. Parameters such as booster, max_leaf_nodes, eta, lambda, alpha uses the default value. Step 0 initialise the model with default value. Step 1 to 5 in Table 3 shows the optimised parameter used in the model. Figure 3 shows the top 15 features used by the model.

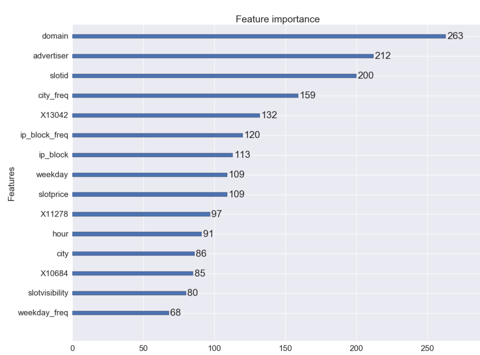

Figure 3: Top 15 feature

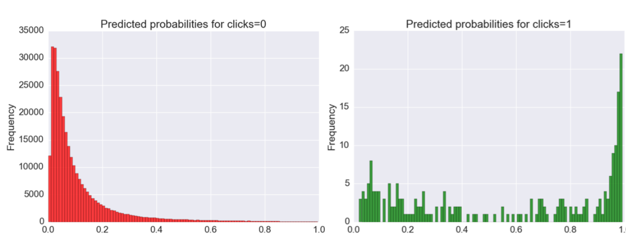

Figure 4: Histogram of click probability

The CTR model scores 0.875 AUC on the validation set. An AUC score of 1.0 implies that all true positive are ranked before true negative, while an AUC score of 0.5 indicates a random ranking. Figure 4 shows the prediction histogram for click against the true value. From clicks=0 histogram (red graph), it is observed that the model performed well in prediction click=0 as the graph quickly taper off at higher probability. However, click=1 (green graph) is not as good, with significant number of click=1 spreading across the x-axis. The ideal shape of click=1 (Green) would be a mirror image of click=0 (Red) where a 'U' shape histogram is formed when the 2 graph are overlapped together.

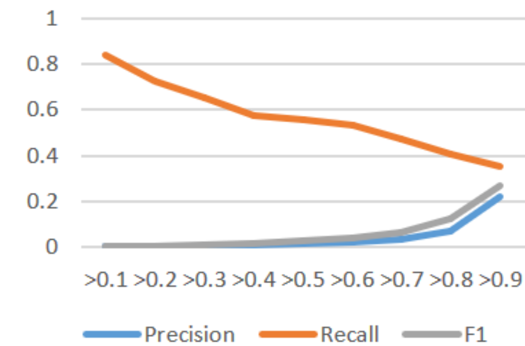

Figure 5: Effect of Probability Threshold on Click=1

Figure 5 shows the F1, Precision and Recall for clicks=1. As observed, a higher probability threshold increases the precision of the model to predict a click through while a low probability increases the recall of click through at the expense of precision. In real world application, the probability threshold could be adjusted for high recall which translates to buying more impressions and achieve more clicks. Or high precision which translates to buying impression with good click through and achieve good CTR.

3.2.2 *Bid optimisation*

In data exploration, it is observed that bid price is at least 220 while the paid price on average is only 78. Translating to real world application, the bid optimisation strategy would need to bid at least 220 to stand any chance of winning. The formula (1) models the bidding strategy used in conjunction with the CTR estimation model.

In literature review, Claudia et al. shows that an aggressive bidding strategy for good impression yield positive result and in data exploration, it is noted that bid price by competitor is at least 220. Factoring these considerations, a bidding strategy that combines a base bid and a variable component is implemented (1).

$$price_i = base\_bid + var\_bid(\frac{y\_pred_i - conf\_thres}{1 - conf\_thres}) \quad (1)$$

The first component base bid (i.e. base_bid) represents the minimal price (i.e. 220) for a bid. The second component is variable bid (i.e. var_bid) multiple by the confidence level of the CTR estimation model to introduce higher bid for higher confidence impression. For the purpose of this coursework, base bid need not be 220 as bid is only compared against pay price and not bid price to determine a win. Recall in Figure 5, probability threshold directly affects precision and recall thus confidence (i.e. conf_thres) is introduced to strike a balance between advertiser's objectives. This confidence threshold can be adjusted to control how aggressive to bid or to conserve budget by bidding only the confident ones. The strategy is condition to bid only if predicted probabilities (i.e. y_pred) is greater than confidence threshold. The subscript i denotes the ith impression. Figure 6 shows the clicks and CTR using different base bid and confidence threshold while variable bid is fixed at 100. It is observed that as confidence threshold increases, the click decreases while CTR improved. This shows that the model is conservative to bid only on high confidence impression. On the other hand, a low confidence threshold allows the model to capture more clicks at the expense of CTR and amount of budget used.

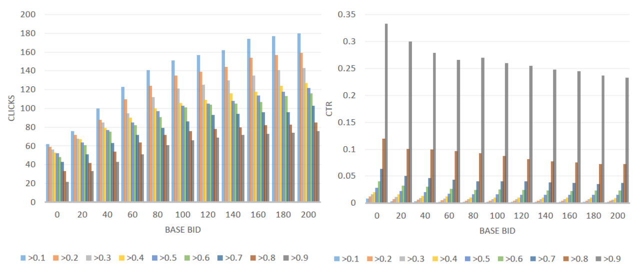

Figure 6: Effect of confidence threshold and base bid on Click, CTR and Spend

To tune the base bid, variable bid and confidence threshold, a grid search is used with a scoring function (2) to score using click and CTR with various weight.

$$Score = \frac{No.of Purchased Clicks}{Total no.of Clicks} * w1 + CTR * w2 \quad (2)$$

	Weight	Optimal Parameter	Result on Validation set
1.	Click(w1): 40% CTR(w2): 60%	Base bid: 160 Variable bid: 110 Confidence: 0.1	Imp: 86790 Click: 174 CTR: 0.002 eCPC: 35.37 CPM: 70.91 Spend: 6154
2.	Click(w1): 20% CTR(w2): 80%	Base bid: 40 Variable bid: 70 Confidence: 0.9	Imp: 119 Click: 36 CTR: 0.3 eCPC: 0.178 CPM: 54.01 Spend: 0.64

Table 4: Effect of weight on Clicks performance

By adjusting the weight, the model could be tuned to fulfil different objectives. Table 4 shows 2 sets of weight with their corresponding parameters and result on validation set. Using a weight of 40% on clicks and 60% on CTR, the model purchased 174 out of 220 (80%) possible impression with click through using a budget limit of 6250. The model achieved a CTR of 0.2% and is much higher than the average CTR of 0.07% in the train and validation set.

Tweaking the weight to 20% on clicks and 80% on CTR tunes the model to become selectively in the impression it purchased. Although a high CTR (30%) is achieved but only 36 clicks is obtained. Advertiser using such model will take a long time for their ads to reach a large customer base.

4. CONCLUSION

The utility (i.e. CTR Estimator) and bid estimator has demonstrated positive improvement over the advertisers in the validation set. The bid estimator has also shown to exhibit flexibility in achieving high number of clicks or to conserve budget while maintaining high CTR. Further works in feature engineering and ensemble could be explored. In feature engineering, new features to extract user behaviours such as time interval between visit, time of visit, and etc could be explored to further refine the model. For ensemble, gradient boost tree could be combined with logistics regression [He et al. 2014] or Neural network to improve the click prediction as the current gradient boost model did not achieve a clean prediction for clicks=1 as seen in Figure 4 histogram (i.e. Green graph).

4.1 Roles

The group consists of Kah Siong Tan (KS), Min Ong (Min) and I. I developed and contributed functionality from reading and writing CSV, constant bidding model, random (Gaussian) bidding model, feature engineering in continuous format, XGboost CTR model, my bidding strategy, click evaluation using F1 metric and bid evaluation using CTR, spend, CPM and CPC. Min contributed click evaluation using AUC and click histogram, CNN CTR model, Random (uniform) bidding model, bidding strategy with grid search, optimised bid evaluation performance, feature engineering in one-hot format. KS contributed SGD CTR model, FM-ALS and FM-SGD CTR model, bidding strategy and setup Github project. Linear bidding using logistic regression with one hot coding is co-developed by all the members. In term of workload, i felt it was evenly distributed. Overall, the team collaborated well and strives to contribute in every way possible to improve the model's performance. The ensemble models are co-developed together with everyone contributing ideas and codes to improve the AUC score for CTR estimation and trying out different bidding strategies. The team has been together for 3 assignments is a strong testament of good rapport and cooperation.

5. REFERENCES

[Capen et al. 1971] E. C. Capen, R. V. Clapp, and W. M. Campbell. 1971. Competitive Bidding in High-Risk Situations. *Petroleum Technology* 23 (1971). http://www.cs.princeton.edu/courses/archive/spr09/cos444/papers/capen_et_al71.pd

[Chen and Guestrin 2016] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. *CoRR* abs/1603.02754 (2016). http://arxiv.org/abs/1603.02754

[He et al. 2014] Xinran He, Junfeng Pan, Ou Jin, Tianbing Xu, Bo Liu, Tao Xu, Yanxin Shi, Antoine Atallah, Ralf Herbrich, Stuart Bowers, and Joaquin Quiñonero Candela. 2014. Practical Lessons from Predicting Clicks on Ads at Facebook. In *Proceedings of the Eighth International Workshop on Data Mining for Online Advertising (ADKDD'14)*. ACM, New York, NY, USA, Article 5, 9 pages. DOI: http://dx.doi.org/10.1145/2648584.2648589

[Hoelzel] Mark Hoelzel. The Programmatic Advertising Report: Real-time bidding is taking over the digital ad market. (????). http://uk.businessinsider.com/the-programmatic-and-rtb-ad-report-2014-8

[Lee et al. 2012] Kuang-chih Lee, Burkay Orten, Ali Dasdan, and Wentong Li. 2012. Estimating Conversion Rate in Display Advertising from Past Erformance Data. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*. ACM, New York, NY, USA, 768–776. DOI: http://dx.doi.org/10.1145/2339530.2339651

[Menon et al. 2011] Aditya Krishna Menon, Krishna-Prasad Chitrapura, Sachin Garg, Deepak Agarwal, and Nagaraj Kota. 2011. Response Prediction Using Collaborative Filtering with Hierarchies and Side-information. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '11)*. ACM, New York, NY, USA, 141–149. DOI: http://dx.doi.org/10.1145/2020408.2020436

[Perlich et al. 2012] Claudia Perlich, Brian Dalessandro, Rod Hook, Ori Stitelman, Troy Raeder, and Foster Provost. 2012. Bid Optimizing and Inventory Scoring in Targeted Online Advertising. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '12)*. ACM, New York, NY, USA, 804–812. DOI: http://dx.doi.org/10.1145/2339530.2339655

[Regelson and Fain 2006] Moira Regelson and Daniel C. Fain. 2006. Predicting Click-Through Rate Using Keyword Clusters. (2006).

[Robinson 1997] G.B. Robinson. 1997. Automated collaborative filtering in world wide web advertising. (July 24 1997). https://www.google.com/patents/WO1997026729A2?cl=un WO Patent App. PCT/US1996/020,429.

[Wang et al. 2016] Jun Wang, Weinan Zhang, and Shuai Yuan. 2016. Display Advertising with Real-Time Bidding (RTB) and Behavioural Targeting. *CoRR* abs/1610.03013 (2016). http://arxiv.org/abs/1610.03013

[Zhang et al. 2016] Weinan Zhang, Tianming Du, and Jun Wang. 2016. Deep Learning over Multi-field Categorical Data: A Case Study on User Response Prediction. *CoRR* abs/1601.02376 (2016). http://arxiv.org/abs/1601.02376

[Zhang et al. 2014] Weinan Zhang, Shuai Yuan, and Jun Wang. 2014. Real-Time Bidding Benchmarking with iPinYou Dataset. *CoRR* abs/1407.7073 (2014). http://arxiv.org/abs/1407.7073