Multi-Agent Artificial Intelligence Group #13 Coursework

Mihail Voicu #17032444
Ismail Guennouni #16095263
Tim Ohlenburg #16117711

1. INTRODUCTION

In the young and exciting field of computational advertising, real-time bidding occupies a central area. According to [11], we trace the first paid-for banner to an ATT ad, to the online version of the magazine HotWired, in October 1994. Things really got going in 1998, when goto.com became the first commercial online ad platform to show success. Later acquired by Yahoo!, it competed with Google's AdWords from 2002. Such systems were for sponsored search ads, which work by running a campaign targeted to specific words. In 2005, real-time bidding emerged as a new paradigm, with DoubleClick and others soon competing in this fast-growing space. In the space of around 100 milliseconds an impression, or advertising opportunity on a webpage, is sold when a user calls the site up. Ad exchanges are the key intermediaries, and they handle over tens of billions of impressions a day across the world wide web. This coursework gives us an opportunity to delve into the varied and challenging problems that the creators of bidding algorithms face.

1.1 Key results

Let us summarise the key results in their order of appearance below.

Constant bidding A line search suggests that 80 CPM is the optimum bid value. It garners an average of 74.6 clicks when sampling random impression orders, resulting in an average CTR of 0.052%.

random interval bidding Sampling many intervals of varying width and level, we find that yields a very similar result, as the random interval with most clicks invariably contains 80, the optimal constant bid. Fr example, for an interval width of 8 fen, the optimum value is 78-86, giving 73.1 clicks.

Competitive random bidding This format yields more nuanced results. When the range of intervals of the agents is restricted to that seen in the data, most clicks are won by an agent bidding 268-283. But when the intervals go from 100-3000 then a moderate range of 300-775 wins the day. When the number of agents is increased from 50 to 100 then the same qualitative results hold, but the winning ranges are inflated upwards.

Linear strategies We used various parameterisations of logistic regression, neural networks and gradient-boosted decision trees to estimated predicted click-through rates. Both neural networks and boosted trees provide a ROC-AUC score of around 70%. A base bid of 55.14 was found to be optimal given our pCTR values. Our best specification resulted in 124 expected clicks on the validation set.

Non-linear strategy Employing the Optimal Real-Time Bidding (ORTB) method introduced in [16], we were unable to attain superior results to linear bidding. Choosing $\lambda = 4 \times 10^6$ and $c = 50$, the best results was 117 expected clicks on the validation set.

Multi-agent strategy Drawing on results from the previous sections, our own simulator, auction theory, game theory and experimental evidence form the learderboard, we determine that a mixed strategy is most likely to guarantee a fair amount of clicks. Assuming a roughly five-fold increase of the price level over the training data, we submit bids that are a combination of constant, linear and non-linear strategies.

2. RELATED WORK

2.1 Empirical studies

We start our review of the literature with two important empirical contributions that highlight typical design aspects of the RTB market. [17] presents data from a Chinese demand-side platform. Its companion paper [6] also introduces the dataset and points to a competition run by the company which encouraged and promoted scholarship in the field. Of course, there are also strong indications that it is exactly this same data which is at our disposal. Another reference work on the practical aspects of the market is [14], which is based on data from a UK ad exchange. It investigates aspects such as the distribution of prices (bid and paid), the pacing of advertisers and key metrics of the market such as the number of impressions and clicks throughout the course of a day. The apparently sub-optimal behaviour of advertisers is another interesting aspect discussed there, and one which provides useful insight to our bidding strategies in this report.

2.2 Economic analysis

The study of markets is a key domain of economists. An influential early example that introduced the field to a wider audience in that discipline was [3], which investigate the typical market design of the generalised second-price auction prevalent in advertising RTB, and find that it is neither incentive-compatible nor stable in dominant strategies. In other words, these auctions invite 'gaming' behaviour and never converge on set strategies. Indeed, in a follow-on publication that analyses market data ([2]), the same authors find evidence of strategic behaviour that lowers market efficiency as well as platform profits. In a similar empirical study complemented with a user behaviour model, [4] estimate that competition among ads depresses click-through rates. In the tradition of economics, mechanism design - how to structure the rules of interaction to achieve some objective - and welfare - who benefits from the outcomes - are topics of relevance for RTB auctions. An example of this line of investigation is [1], which takes a game theory approach to modelling the market. A similarly strong game theory perspective is taken in [8], but they approach the field from a computer science angle via multi-agent systems, where algorithms are a fundamental part of analysis. Such computational considerations are also key for the next area of relevant research.

2.3 Machine learning innovations

The field of machine learning, and in particular the especially successful area of deep learning, has a strong impact on many data-rich domains, and computational advertising is no exception. Given response times in the order of milliseconds, decision algorithms in this space need to scale to high-dimensional variables and many transactions. One method, inspired by natural language processing, is to encode user and ad characteristics in a dense vector using a hybrid filtering doc2vec approach [9]. Another task in the bidding algorithm pipeline is the prediction of user responses. [15] go a different route, transforming the data with factorisation machines, restricted Boltzmann machines or a de-noising auto-encoder then apply a neural network architecture to the output.

2.4 Bidding strategies

An area of primary importance to practitioners in the field concerns bidding strategies and algorithms. [14] suggests that many bidders run (at least in 2013) fairly simple bidding strategies of daily pacing. That is, they set a daily budget and then try to set their typical bid such that it lasts much of the day, while being likely to deplete at the end. An alternative approach, which seems less optimal, appears to be that advertisers spend their target as fast as they can. However, more sophisticated approaches have been developed that promise lower costs per click. A good example is [16], which uses a Bayesian approach to develop a non-linear response to the expected click-through rate. In particular, they find that advertisers should focus on the less competitive parts of the market as it offers a lower cost per expected click, and they present a tractable formulation for implementing this method. Another is lift-based bidding [12], which considers the impact of showing ads to particular users in comparison to the counterfactual of not showing them ads, thereby maximising the 'lift' in clicks from each ad. Another interesting approach is the 'bidding machine' of [7], which frames the various different aspects of the bidding problem as a joint learning problem that it optimises directly.

2.5 Reinforcement learning

A recent crop of models draws on the machine learning discipline of reinforcement learning, in which agents learn to optimise their reward in a dynamic environment that responds to their actions [10]. One the basis of deep learning methods, (see [15] for an example), modern reinforcement learning methods can pick up on complex patterns. A particularly relevant literature has recently emerged in multi-agent reinforcement learning models, which additionally models the interaction of agent behaviours. An example is [5] looks at the closely relate field of sponsored search, which they also analyse in an RTB setting. They use a model-based reinforcement learning approach that shows promise in real-life A/B testing and shows that reinforcement learning can be a useful paradigm in any computational advertising setting where the problem can be formulated as a Markov decision process. A more direct development of the reinforcement learning approach is [13], an asynchronous advantage actor-critic architecture with multiple reward objectives proposed. This advanced architecture promises improvements over less tailored methods and possibly heralds the start of a reinforcement algorithm arms race in computational advertising marketplace.

3. DATA EXPLORATION

The aim of this section is to describe and analyse the dataset that forms the basis of this assignment. Our description is influenced by [17], which delivered an authoritative and extensive description of the dataset upon its publication as competition source material. The section opens with a broad overview of the dataset at the aggregate and at the variable level. Secondly, we present a few basic digital advertising statistics. The third and final subsection looks at realised bids and prices. Unless otherwise stated, we calculate these figures for the training set only. It is important to note that the data only includes winning bids. Where an impression was offered but no bid was in excess of the minimum 'slot' price, the impression was not included. Our analysis is therefore constrained to a limited view of the data and would have to be adjusted for bidding on live ads.

3.1 Variables and observations

In total there are 3,038,281 observations in the iPinYou dataset, of which around 2,431,000 or 80% are from the training set, with the remaining 20% or ca. 608,000 observations split evenly between the validation and test sets. The structure of the test set varies in that it excludes click, bid and realised price information. Table 1 provides information of each of the variables in the original dataset. It states how many unique values there were of each, and also which share of the number of categories was for impressions that resulted in clicks. As most of our variables are categorical, our primary data preprocessing goal is to convert the categorical variables into binary one-hot vectors. Since this results in many sparse variables, we only included those with a reasonably large number of categories associated with clicks.

3.2 Advertising analytics

Given the lack of data on conversions and the value of these, we are left with click counts as the target outcome of advertising campaigns. The statistics provided in this section[1] are displayed above. Table 2 shows key indicators, which are given for the whole dataset, which would also be calculated for specific partitions of the dataset such as the advertising campaign level for real-life analytics. We see a typically low click-through rate of 0.07%. The cost of one thousand impressions (CPM) to the advertiser was an average 78.15 fen. Each click cost only around 106 CPM on average, meaning that successful impressions were not that much more expensive. However, without data of the

[1] See [17] for definitions of the acronyms used in Table 2.

Variable	Example	# Unique	Included as
click	0	2	orig
weekday	5	7	1hot
hour	22	24	1hot
bidid	B7bea805[...]	2,430,981	–
userid	2e880fb7[. . .]	2,342,677	–
useragent	windows_ie	38	1hot
IP	125.37.175.*	503,975	–
region	308	35	1hot
city	309	370	–
adexchange	2.0	5	1hot
domain	TrqRTvKa[. . .]	23,013	–
url	6447a7df[. . .]	763,961	–
urlid	nan	1	–
slotid	2015392487	52,283	–
slotwidth	300	21	'slotsize' 1hot
slotheight	250	14	'slotsize' 1hot
slotvisibility	second_view	11	1hot
slotformat	0	4	1hot
slotprice	5	286	orig
creative	A4f763f7[. . .]	131	–
bidprice	238	8	orig
payprice	5	301	orig
keypage	0f951a030[. . .]	19	–
advertiser	3427	9	1hot
usertag	10063	744,036	–

Table 1: Training set list of variables with examples, number of unique values and transformation

Impressions	2,430,981
Clicks	1,793
Click-through rate	0.07%
Average cost per mille (in fen)	78.15
Expected cost per click	106.0

Table 2: Training set advertising analytics

return from conversions, we cannot judge whether this was a profitable proposition for the average advertising campaign.

Figures 1 and 2 show the variation of click-through rates for selected variables. For instance, we can note that the middle of the night sees few clicks per website visit, but even fewer during lunch hour, whereas there are fairly many in the early hours of the day. The middle of the week also shows a higher click-through rate, underlining the temporal nature of user behaviour. In addition, we can see from Figure 2 that traffic via some ad exchanges and from some regions results in substantially more clicks than from others. Each of the dataset variables reveals such patterns and we can conclude that there is clear circumstantial evidence for the usefulness of user targeting in determining ad spend.

3.3 Bids and prices

Finally, we take a quick look at the price outcomes of the auctions in our dataset. In particular, we are interested in the maximum price bidders were willing to pay (in the second price auction, participants bid truthfully) and the actual price paid according to the generalised second price auction format prevalent in advertising RTB markets. Figure 3 illustrates that bids are won by bidders willing to pay a maximum of 220 to 300 fen per mille - at Table 1 illustrated, there are only 8 distinct bid values in the dataset.

The price bidders actually pay tends to be significantly lower in many cases though, suggesting either that valuations for a given impression differ widely or that there are so many advertising opportunities that competition is moderate and a committed bidder can usually claim his prize. Although in some cases the second-highest bid is very close to the winning one, most are significantly lower. In fact, ads with the lowest bids - 227 fen per mille - attracted the highest mean price. We can actually make the counter-intuitive observation that the ratio of price paid to bid declines with the bid!

Much more analytics could be produced, but space is limited and the focus of this report is the upcoming substantial part on bidding strategies.

4. APPROACH AND RESULTS

Having reviewed the general context of the real-time bidding problem as well as the dataset, we can now turn to the bidding strategies that we developed. Going from very simple, heuristic-based strategies towards more sophisticated approaches, in each section we explain our fundamental understanding of the decision problem, how we propose to solve that problem, and what the quantitative outcome is.

Our goal is to maximize a key-performance indication (KPI) subject to our total cost being lower than our budget of 6,250 as displayed in the equation below:

$$\begin{aligned} \max_{params} \quad & KPI \text{ of bidding strategy} \\ \text{subject to} \quad & cost \;\; \leq \;\; budget \end{aligned} \tag{1}$$

The KPI we focus on is the number of clicks, but there are other metrics (click-through rate, cost per click, etc.). We also provide the values of these, highlighting that maximizing other metrics achieves different results.

4.1 Basic bidding strategies

In the following section, we evaluate basic bidding strategies on the validation set, following clarifications from the teaching assistant. There is no training on the training set as such, as we have not been given a budget on the training set

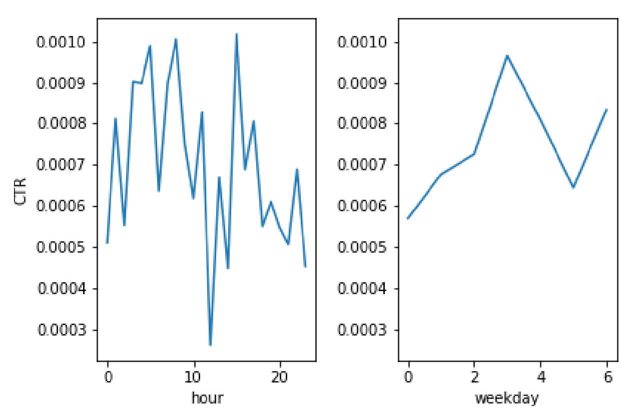

Figure 1: Average click through rates over the day (left) and week (right)

and using the same budget of 6250 would have been incongruous as the training set contains around 8 times as many auctions. Having said that, we would approach the problem in the exact same way on the training set, assuming we get a commensurate budget.

Impressions	303,925
Clicks	202
Click-through rate	0.07%
Average cost per mille (in fen)	78.23
Expected cost per click	105.7

Table 3: Validation set advertising analytics

The two sets are similar in many ways as we can see from the table of summary statistics for the validation set above.

4.1.1 Constant bidding

Decision problem description

In this section, we assume the agent bids a constant bid for all bid-requests, and if the bid is higher than the pay price, the agent wins the auction and pay the pay price. The challenge is to find the optimal constant bid that would maximise the number of clicks on the auctions won. The difficulty is compounded by the budget constraint: with the constant bid, the bidding agent can win some auctions but the budget might not allow for participation in all of them. Therefore we will only be able to participate in a subset of all 'winnable' auctions. Thus, for a given constant bid, we compute the expected number of clicks this constant bid might win. We achieve this by repeatedly randomly sampling out of all the 'winnable' auctions set with that constant bid and calculate the total clicks won per auction. Then we average the total clicks across the number of random samples to obtain the expected number of clicks for that given constant bid.

Proposed solution

Our solution is to go through a systematic grid search of constant bids, between a low of 1 and a high of 301 (maximum pay price+1). There is no need to search higher constant bids because they would be sub-optimal. A constant bid of 301 will win all auctions but the agent will have to participate in fewer auctions since the budget will be exhausted before the end of the auction.

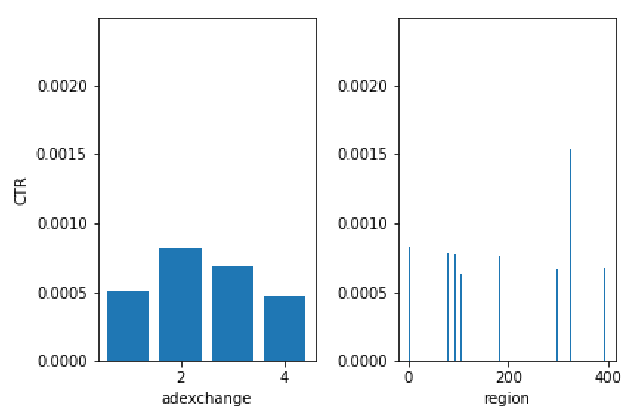

Figure 2: Average click through rate by ad exchange ID (left) and region (right)

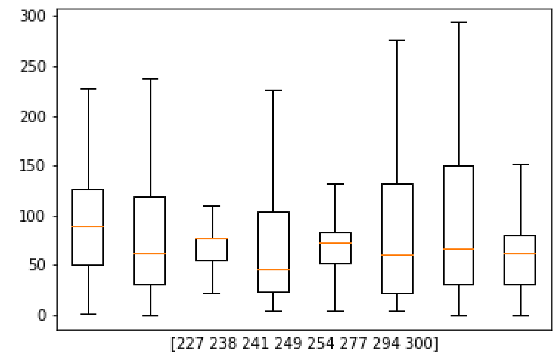

Figure 3: Mean and quartiles of prices paid by bid value (bid values for each of the eight box plots are shown as x axis caption)

The algorithm is the following. For each constant bid in the grid, if the auctions won result in a total cost higher than our budget, we repeatedly randomly select auctions from the set of winnable' auctions until the budget is exhausted and record the number of clicks won for that sample. We then average across samples to get an estimate of the expected number of clicks we would win with this constant bid. That is, not knowing what other agents may bid, for a given constant bid, we compute the expected number of clicks won by the constant bid. Then another constant bid is considered. The constant bid with the highest expected number of clicks is the "optimal" bid.

Quantitative result

With the methodology above, and using 1000 sampling iterations from the winnable auctions for each bid, we find that the optimal constant bid is 80 CPM. At this level, we obtain an expected number of clicks of 74.58 and an expected CTR of 0.052%. Interestingly, the CTRs are highest for the higher constant bids, averaging 0.067%, whereas they are lower for lower bids, including our optimal constant bid. This is driven by the change of the denominator of CTR: with higher bids, a lot more auctions are won but the increase in the associated total number of clicks won is not as high. This results in higher CTRs for higher bids as the increase in the denominator (total number of auctions own) is lower than the increase in the numerator (total number of clicks won).

4.1.2 Random bidding within interval

Decision problem description

The challenge here is to come up with an interval from which we repeatedly sample bids for each auction, such that we maximise the number of clicks won. The potentially winnable clicks now vary from one auction to the other as they are randomly sampled. The width of the interval can vary as well adding another element of uncertainty. If we were to systematically evaluate every single interval between min and max pay price, there would be around 45,000 possibilities to explore.

Proposed solution
Rather than the brute force approach, we do an iterative grid search on disjoint consecutive intervals of varying lengths to pinpoint the most promising intervals (that max the expected number of clicks). However, we use knowledge from the constant grid search to select what we already know are good lower and upper bounds for such intervals. From the previous section, we can infer that the most promising constant bids are between 70 and 100. Indeed, bids below 70 and above 100 have significantly less expected number of clicks than those within that range. Therefore, we constrain our lower bounds to be above 70 and our upper bounds to be below 100. We test varying interval widths, namely : 5, 8, 10, 20 and 40.(For instance, start with width 5 [70-75], [75,80]... [95-100]; then width 10 [70-80], [80-90] ...and finish with width 40). We therefore cover various potential widths within the large interval we identified as most promising.

Quantitative result
Irrespective of the width, the intervals with the best performance are the ones in the vicinity of our constant bid optimum of 80. The performances are quite close, and we cannot be sure how close these expected values are to the true values. The means and variances of the expected number of clicks per interval are quite close and we would need a lot more time and computational power to establish the best interval with certainty. Still, based on 100 sampling iterations for each interval, it appears the interval [78-86] has the best expected number of clicks at 73.05, followed by the [80-85] interval which on average gets 72.74 clicks. Surprisingly, the best random interval for clicks is in line with the average CPM over the whole dataset, even though the average CPC is higher.

4.1.3 Competitive random bidding

Decision problem description
In this section, the aim is to create a number of agents (50 to 100) that are all using random bidding from intervals to compete for the auctions in the validation dataset. The winner of the auction is the highest bidder, but they pay the second highest bid (see Winning Criterion 2). All agents have a budget of 6250. Here, a brute force approach is not feasible given computational and time constraints, since there is a very high number of possible intervals (theoretically, each agent can bid up to their total budget per auction) and the problem becomes intractable. Moreover, an "optimal" strategy in the sense that it would be best no matter what the other participants do is very hard to find, if it exists at all. In this strategic setting, our best response is contingent on choices the others make.

Proposed solution
The way we approach this problem is through simulating multiple environments, and coming up with general heuristics that may help us later on with problem 5. First, we simulate environments in which the random bids are drawn from intervals whose bounds are of the same orders of magnitude as the current pay prices 100-300. Second, we simulate environments where agents draw their bids from ranges with very high upper bounds (100-3000 for example). Finally, we simulate environments where agents draw their bids from disjoint consecutive ranges ([1-100], [101-200] and so on).

Quantitative result
If generally the bids are low (1-300), then the winning strategies are the ones that are close to the top of the range (in our simulation, winning interval is [268-283]). In other words, it makes sense to bid relatively high since you can win a lot of auctions and not exhaust the budget, meaning agents bidding high can participate in most auctions. Conversely, if the bids of the agents don't have an upper bound that is similar to current bid prices 1-300, then the winning strategy is the opposite of the above (winning agent is the one with the lowest mid-range [334-775]). Here, agents who keep bids towards the lower end of the other agents bids tend to do best. The reason is that the other high bidders now exhaust their budget much more quickly, leaving the low bidders free reign on the later auctions where they win and can get clicks with very limited competition. The third simulation, where the 50 agents each bid in relatively narrow, consecutive and non overlapping intervals ([0-20] [20-40] ...[980-1000]) shows another interesting dynamic. Here, it is worth bidding in the mid to high end of the range. This is explained by the need to keep balance between winning auctions by submitting competitive bids, while keeping enough budget to participate in later auctions.

We re-ran the simulations with 100 agents instead of 50, and while the gist of the strategies is the same, the fact that more agents are competing will generally push the 'good' intervals higher, as there is more money available for those auctions. So the higher the upper bound of potential bids, the higher the intervals for winning agents need to be in order to win any auctions.

4.2 Linear bidding strategy

We assume that a linear bidding agent estimates the click-through-rate (CTR) of an impression and uses it to determine a linear bid response. Thus, this section firstly discusses the CTR estimation problem and then the training of linear strategies (that are linear in the estimated CTR). For the CTR estimation, a benchmarking experiment with different binary classifiers was conducted using both the training and validation datasets. Then for the linear bidding strategy, the validation set was used for parameter calibration as the budget of 6250 does not apply to the training set and the test set does not contain the click values.

Decision problem description
Firstly, for the CTR estimation, the literature shows that many models are available (e.g. factorization machines, neural networks, etc.). The challenge here is deciding which of these models to use. All of them take some features as input and the binary click values as the target. The features include information about the ad itself (e.g. size of ad, url) and the user (e.g. time of bid, region of user) and are represented in one-hot encoded vectors. The challenge here stands in deciding which of these features to include as predictors of the click values. Then, following feature selection and model selection, the tuned optimal model is used to predict the binary values of clicks. Much like the literature, this paper defines the predicted CTR (pCTR) as the unconditional probability of having a click: $\Pr(\text{click} = 1)$, for each auction.

Secondly, the linear bidding strategy assumed in this paper takes the following form:

$$bid = base\_bid \times pCTR/avgCTR \tag{2}$$

where avgCTR is the implied empirical CTR of the training set (no of clicks divided by the no of impressions). The model parameter is `base_bid` and its calibration constitutes the model training challenge.

Proposed solution

In deciding which features are relevant, [6] was considered. The bid ID, URLs, bidding prices and paying rices were excluded as they are either unique for each auction or worthless for the classification problem. All other variables were kept and 1-hot encoded similarly to the literature methods and displayed in Table 1 in Section 3.2. This method was applied to the training, validation and test sets and resulted in features of 159 dimensions and lengths equal to the respective sizes of the datasets.

As for predicting clicks, the above features were used into a benchmarking experiment with a (i) logistic regression, (ii) a neural network and (iii) a random forest. The test set was excluded from all procedures as the target (click) is missing. Each model was trained and tuned on the training dataset using versions of stochastic gradient descent. The parameter grids for all models were set as flexible as possible without increasing computational time too much. The various model parameters are shown in the appendix. All models were compared out-of-sample on the validation set after the training and tuning them on the training set.

The model with the best out-of-sample score was then used for CTR prediction (pCTR = Pr(click=1)) on the validation set. The pCTR was then inputted into equation (1) on which the optimal `base_bid` was found by running a line search of 400 equidistant points between 25 and 350. Much like the previous sections, the `base_bid` that yields the highest expected number of clicks is declared optimal. We drew 500 random samples from the winnable auctions set to compute the expected number of clicks. 500 is a number determined experimentally to ensure that the sample empirical CTR is comparable to the full validation set's empirical CTR.

Quantitative result

With the pre-processed features, the optimal parameters for the logistic regression were a regularization coefficient of 0.001 and an L1 regularizer. As for the neural network: a relu activation, a regularization term of 0.001 and a design with 3 hidden layers composed of (50, 100, 50) neurons were optimal. Lastly, the random forest was run as part of the CatBoostClassifier library which was built to deal with large datasets of 1-hot encoded vectors. Its optimal parameters were: 100 trees of depth 16 and a regularization coefficient of 3.

All models were tuned with 2-fold cross-validation and benchmarked with out-of-sample roc-auc as the objective function. The logistic roc-auc score was the lowest at 61.9% while the neural network and random forest both had scores close to 70%. Regarding other metrics, the accuracy scores are similar, but the confusion matrices are by far the most informative. The neural network and random forest both predict 0 for all the true positives (click=1) observations, indicating that they both predict the negative class all the time. In this aspect, both models perform just as good as a dummy classifier that always predicts the most occurring class. Still, for the CTR estimation, we need the Pr(click = 1) and not the binary class predictions. Thus, following the literature, the model with the best roc-auc score was chosen: the CatBoost random forest.

Our results are far from perfect due to the very large imbalance in the datasets. The training set contains 0.0737% of the positive class while the validation set contains 0.0665% of the positive class. This issue could be addressed by using an under-sampling of the most abundant negative class within cross-validation. Alternatively, a package like SMOTE "Synthetic Minority Over-sampling Technique" could be also be used to over-sample the minority class. To achieve a similar purpose, the models' loss function (log-loss) for the pozitive class (click=1) were weighted by the ratio of sum(negative examples)/sum(pozitive examples) = 1503.6 (computed on the training set) while the loss of the negative class had a weight of 1 (this weighting follows the CatBoost developers' recommendation for highly imbalanced datasets). However, the CatBoost results with a weighted loss are in line with the neural network and catboost without weighted losses: the accuracy is similar as the roc-auc values. The confusion matrix shows that, this time, 108 of the 202 validation clicks were accurately predicted. Also, there are only 94 false negative predictions (click classified as non-click) while there are 64,506 false pozitive predictions (non-click classified as click). As a result, the CatBoost model is much more likely to confuse a non-click for a click and was thus not used further. An implication of this is that for a linear bidder, high pCTR values of false negatives will lead to high bids on ads with a low probability of click and exhaust our budget on irrelevant ads.

One last attempt to improve our CTR estimation stands in stacking the best performing models (neural network and random forest). The meta-learner that is trained with the pCTRs on the validation set is another CatBoost model. This is in turn also cross-validated on the validation set only. Optimal parameters are a depth of 5 with 50 trees an L2 regularization coefficient of 3. For this model, the log-loss of the positive examples was weighted by sum(negative examples)/sum(pozitive examples) on the validation set. The accuracy and roc-auc scores are similar to all previous models (around 70 percent). Interestingly, 154/202 clicks are correctly predicted (true positives) while 93,488 non-clicks are predicted as clicks (false positives). As a result, this method is also abandoned since the linear and non-linear bidders would end up over-bidding on ads with a low probability of clicks, having so many false positives.

The linear search for the optimal base bid yielded a slope value of 55.14 and the optimal implied bids are expected to win 124 clicks on average, while the expected implied CTR is 0.08%. The full budget was used.

4.3 Non-linear bidding strategy

This section is in the spirit of the previous one, but a non-linear bidding function is considered this time.

Decision problem description

Much like the linear bidding, the non-linear bidding agent must calibrate a non-linear function in pCTR. The literature presents many such non-linear functions and this paper

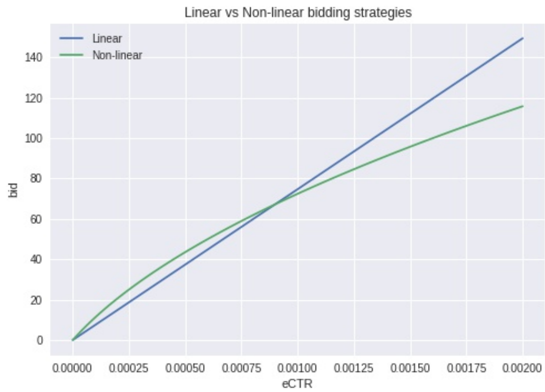

Figure 4: Bids as a function of pCTR where the bidding strategies are linear and non-linear

follows [16]'s The ORTB1 function:

$$bid = \sqrt{\frac{pCTR \times c}{\lambda} + c^2} - c \tag{3}$$

where the model parameters are λ and c. Their calibration represent this section's challenge.

Proposed solution

A grid search over different values of λ and c is done to maximize the expected number of clicks and exhaust the budget. According to [16], their optimal λ was 5.2 $\times 10^7$ while values of c between 50 and 80 show a comparable scale for their bids to our possible ones (see [16] Figures 3 and 4). As such, our c grid contains 50 points between 50 and 100 and the λ grid contains 100 points between 10^{-7} and 10^{-5}.

Quantitative result

Given the already estimated pCTR, the grid search for the optimal parameters yielded a c of 50 and a λ of 4 $\times 10^6$ again obtained with 500 random samples from the set of winnable auctions. The implied optimal bids are expected to win 117 clicks while the implied CTR is 0.09%. The difference from the linear bidder is that the non-linear bidder, although it is expected to win less clicks in a multi-agent framework, it will also win less auctions but have a higher CTR. Thus, if maximizing CTR is the goal, the non-linear bidder performs best.

Much like [16]'s results (e.g. Figures 8 and 10), our results are similar. [16] show that the higher the budget is, the more similar the linear and non-linear bidding strategies will perform (when the number of clicks won is the KPI). Figure 4 shows that the non-linear bids are higher than the linear ones when the estimated CTR is lower than 0.00085. This indicates that the non-linear bidder will allocate more budget to low-CTR ads and spend less per ad. But at the same time, the probability of having a click in the low-CTR cases is implicitly very low, indicating that the potential benefits of the non-linear bidder are not very clear in this simplified static environment.

4.4 Multi-agent bidding strategy

The previous sections developed progressively more sophisticated approaches to click through rate estimation. The time has come to draw on the insights gained there to develop a strategy for a genuinely competitive setting. This section therefore focuses on Criterion 2. For Criterion 1, we submitted our best-performing bid on the leaderboard test, for which we estimated the pCTR via the stacked model described above, excluded the bottom 75% of pCTR impressions and optimised a linear base bid on the remaining impressions' pCTR values.

Decision problem description

We again receive a dataset of impressions with features and prices but where we could optimise various bidding strategies through evaluation on the validation set, the payprice in the data no longer determines the second highest bid. Instead, a realistic auction format with other bidders comes into play. We take insights from auction theory, game theory, economics, the previous sections, empirical evidence from the leaderboard and our own simulator into account in formulating a suitable strategy.

Proposed solution

We consider each major aspect of the situation in turn, starting with a simulator we built for this section.

Multi-agent simulator All groups have access to the 3 strategies we analysed in the previous sections: Random Interval, Linear and Non-linear biding. A major driver of performance is the distribution of the strategies between groups, i.e. how many groups elect which strategy. Each such competitive states leads to different performance for a given strategy. Our simulation takes as inputs the number of groups using each of the 3 strategies as well as their bids, runs through all auctions in the validation set, and returns the average (expected) number of clicks per agent. [2] First we analyse three extreme states, in which the vast majority of groups use a dominant strategy, and compare that to a minority of two groups using the other two. For instance, there would be 25 submissions based on the linear bidding strategy, 2 based on random interval and 2 on non-linear. These extreme cases might not be realistic, yet they allow us to uncover the relative performance of our agents in different scenarios. We then analyse less extreme distributions including an equal split. For the random agents, we run the simulations for multiple possible random intervals of width 250 from 0 to 2500([0,250] then [250,500]...).

Majority(86%)	Rand perf	Linear perf	Non-linear perf
Random	3.6	27.6	29.1
Linear	9.3	6	1
Non-Linear	10.6	27.1	4
Average	7.8	20.2	11.4

Table 4: Average number of clicks per agent across random bidding intervals if highly frequent strategy

Table 4 summarises the results as average expected number of clicks per agent across these random bidding intervals.

[2] We assume there are 29 groups competing, each with a budget of 6,250. We also add random noise to the linear and non-linear bidding strategies to account for small variations in CTR prediction models between groups.

The numbers in between brackets represent the typical range of number of clicks for each agent, depending on the strategy they use, for all permutations of states. The diagonal elements are all low, showing that if a strategy is used by the vast majority of participants, a contrarian choice is generally better than following the herd. On average the linear bidding strategy does best across these three scenarios. We run the same simulation for a less extreme scenario where there is one strategy used by 50% of participants, while other two equally split around 25%each. The results table is shown in the appendix as Table 5.

The heuristic stipulating that an agent is generally better off not using the majority strategy remains true for random and non-linear, but is no longer true for linear. Interestingly, now that there are many agents playing the same strategy, and competing for high pCTR auctions in the case of linear and non-linear agents, the performance per agent has understandably dropped markedly from the more extreme cases where in some instances these agents had these auctions without much competition. In this case, the linear strategy dominates the other 2, no matter what strategy garners a majority of 50% of groups. Finally, we also ran the simulation for an equal split across groups using each of the 3 strategies (Table 6 in the appendix). It indicated that the linear strategy performs best in this scenario.

Having seen the above, we can argue that if competitors strictly adhere to linear and non-linear bids learned from the validation set, no matter what the random agents bid, on average it is often better to use a linear bidding strategy to maximise the number of clicks. In fact, not only is this true on average, it is in fact true as long as the proportion of agents using the linear strategy is not extremely high (86% or higher in our examples). Where does this leave us?

If we believe competitors are "naive", then we should just bid according to a linear bidding strategy, as it has the best performance in our simulations. However, we believe our competitors would have done similar analysis and would probably come to a similar conclusion. This means that it is likely we will compete in an environment where most participants will have some preference for using the predictive power of a linear model to submit bid. The key weakness in our analysis so far is that our bids in a linear bidding strategy are based off a "base bid" that was optimised on the back of pay prices from validation data (criterion 1) while in this competition, linear bidding strategies will be evaluated using criterion 2. As such, there is likely an inflation on pricing, meaning that most teams will revise their base bids higher while keeping the pCTR from the estimated models.

Auction theory, game theory and economics Having simulated strategies as a basis for our approach, we complement the analysis with other observations. To start, auction theory highlights that the generalised second price auction format does not have a stable equilibrium and that bidders have no incentive to bid their reservation price [3]. Since there is no equilibrium, it is impossible to find a 'correct' strategy - the best one will depend on others' bids in any scenario.

Secondly, we can infer from the section 2 on competitive random bidding that more bidders means higher optimal prices. Since there are 29 groups, we know that bids need to be higher for a winning strategy. But since there is no equilibrium, it's not clear how high it needs to be. Simple economic reasoning suggests an average payprice of 600 if all 29 groups are to spend their entire budget and all impressions are sold - we could call this the 'auction-clearing average bid'. While unlikely to be attained can treat this as an upper bound on the average price level.

Thirdly, if we are risk averse - and we do care more about winning some clicks than being indifferent between winning most or winning none - then we should run a strategy that is likely to perform well under various scenarios. The simulator indicates that the linear strategy does best on average, but can perform poorly in some settings. A mixed strategy may be best in this setting.

Empirical evidence The leaderboard gives us an indication of the bids of other teams. Since there is no equilibrium in strategies, bids lack a focal point around which to cluster. Testing out several levels up to the limit implied by the auction-clearing average bid of 600, we can calibrate our strategy somewhat, but need to bear in mind that the final outcome will be different, and likely higher as groups aim to outbid each other. In addition, we note that there are very few clicks for ads with very low predicted predicted clicks, which we can use to limit the impressions that we bid on.

Quantitative result

As outlined above, an exact quantitative optimum is not attainable in this auction format. However, we take the results of our simulator, the relevant theory and the empirical evidence into account in our strategy. In particular, due to our risk aversion we run a mixed strategy, calibrated via the leaderboard to an average bid around 200 for the linear and non-linear strategy. For the constant bids, we use a level of 500, which gave us second spot in an earlier try. The weights of the strategy mix implied by Table 4 lead us in 50% of auctions to choose a linear strategy, in 30% a non-linear one and in 20% a constant one. In our most common, linear strategy, we furthermore exclude the 20% of impressions with the lowest predicted click values, as these have little hope of being clicked on, and the top 5%, as we expect excessive competition for these ads and thus a winner's curse. The exclusion of the low bids is based on an analysis of the click-through range of our bids, which shows that very low pCTRs have little potential (see Figure 5 in the appendix).

5. CONCLUSION

The project offered a fascinating opportunity to explore ideas from machine learning and multi-agent systems in a life-like application. The lack of a stable equilibrium makes for a particularly challenging problem. The availability of the leaderboard provided some insights, but the likelihood of other teams obscuring and adjusting their favoured strategy makes any attempt at direct inference as to the success of one's own approach doubtful. The result that we found perhaps most interesting and surprising is that a simple linear bidding strategy can perform extremely well. In our simulator, such a strategy could often win most clicks when set at the right level. Similarly, it did very well on the leadboard on several occasions. Another remarkable aspect is that it is possible to determine a bidding strategy that manages to buy the majority of clicks despite the limited budget and low number of clicks. Although we were moderately successful, a number of possible improvements come to mind which are discussed below.

The bidding approach we have used could be enhanced in a number of ways. To begin with, we selected variables from the dataset such that we could run the estimation on our limited computational resources. The use of more variables would be a fairly likely way to improve the estimation. Similarly, feature engineering might yield good results without increasing the number of variables much. A related idea would be to consider separate models for each ad campaign, or a similar hierarchical structure in which the CTR prediction model is conditioned on impression features.

More generally, other machine learning architectures might lead to better results than those we presented. For instance, deeper neural networks would be an option if resources for hyperparameter tuning were less limited. A multi-agent reinforcement learning simulator would be another interesting approach. Finally, direct feedback on a bidding strategy is limited by the current set-up, notwithstanding the limited information available from the leaderboard. An interesting research direction would be to explore dynamic optimisation of bids in light of real-time feedback. While reinforcement learning seems particularly relevant in this regard, Bayesian method would offer a compelling alternative.

References

[1] Alexandre De Corniere. "Search advertising". In: *American Economic Journal: Microeconomics* 8.3 (2016), pp. 156–88.

[2] Benjamin Edelman and Michael Ostrovsky. "Strategic bidder behavior in sponsored search auctions". In: *Decision support systems* 43.1 (2007), pp. 192–198.

[3] Benjamin Edelman, Michael Ostrovsky, and Michael Schwarz. "Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords". In: *American economic review* 97.1 (2007), pp. 242–259.

[4] Przemyslaw Jeziorski and Ilya Segal. "What makes them click: Empirical analysis of consumer demand for search advertising". In: *American Economic Journal: Microeconomics* 7.3 (2015), pp. 24–53.

[5] Junqi Jin et al. "Real-time bidding with multi-agent reinforcement learning in display advertising". In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM. 2018, pp. 2193–2201.

[6] Hairen Liao et al. "iPinYou global rtb bidding algorithm competition dataset". In: *Proceedings of the Eighth International Workshop on Data Mining for Online Advertising*. ACM. 2014, pp. 1–6.

[7] Kan Ren et al. "Bidding machine: Learning to bid for directly optimizing profits in display advertising". In: *IEEE Transactions on Knowledge and Data Engineering* 30.4 (2018), pp. 645–659.

[8] Yoav Shoham and Kevin Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

[9] Simon Stiebellehner, Jun Wang, and Shuai Yuan. "Learning Continuous User Representations through Hybrid Filtering with doc2vec". In: *arXiv preprint arXiv:1801.00215* (2017).

[10] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[11] Jun Wang, Weinan Zhang, Shuai Yuan, et al. "Display advertising with real-time bidding (RTB) and behavioural targeting". In: *Foundations and Trends® in Information Retrieval* 11.4-5 (2017), pp. 297–435.

[12] Jian Xu et al. "Lift-based bidding in ad selection". In: *Thirtieth AAAI Conference on Artificial Intelligence*. 2016.

[13] Chaoqi Yang et al. "MoTiAC: Multi-objective Actor-Critics for Real-time Bidding in Display Advertising". In: ().

[14] Shuai Yuan, Jun Wang, and Xiaoxue Zhao. "Real-time bidding for online advertising: measurement and analysis". In: *Proceedings of the Seventh International Workshop on Data Mining for Online Advertising*. ACM. 2013, p. 3.

[15] Weinan Zhang, Tianming Du, and Jun Wang. "Deep learning over multi-field categorical data". In: *European conference on information retrieval*. Springer. 2016, pp. 45–57.

[16] Weinan Zhang, Shuai Yuan, and Jun Wang. "Optimal real-time bidding for display advertising". In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2014, pp. 1077–1086.

[17] Weinan Zhang et al. "Real-time bidding benchmarking with ipinyou dataset". In: *arXiv preprint arXiv:1407.7073* (2014).

APPENDIX

Model hyperparameters for linear and non-linear bidding:

(i) Logistic regression:
- regularizer : [L1 norm, L2 norm]
- regularization: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]

(ii) Neural Network:
- hidden layer sizes: [(100,200,20), (100,50, 10), (50,100,50)]
- regularization: [0.0001, 0.001, 0.01, 0.1, 1, 10, 100]
- activation: [relu, tanh, logistic]

(iii) Catboost Random forrest:
- number of trees: [50, 100, 150, 200]
- depth: [5, 8, 10, 12, 16]

Tables for Section 5, strategy simulator:

Majority(50%)	Rand perf	Linear perf	Non-linear perf
Random	4.3	16.4	1.5
Linear	5.4	9.2	0.6
Non-Linear	5.5	19.2	1
Average	4.9	14.9	1

Table 5: Average number of clicks per agent across random bidding intervals if 50/25/25% split

Equal split	Rand perf	Linear perf	Non-linear perf
rand/lin/non-lin	5	14	0.9

Table 6: Average number of clicks per agent across random bidding intervals if equal split

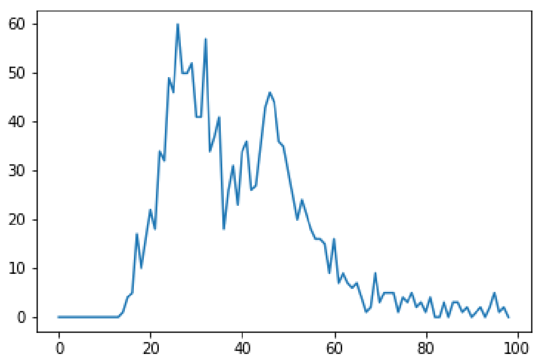

Figure 5: Clickthrough rate by pCTR bin for the lower pCTR ranges containing most impressions)