Giza AdNetwork Agent Project report

Team giza: Guy Gomberg, Itay Waxman, Zohar Paskar, Amit Zitsman

1. Intro

Our main objective in the workshop was to create the most efficient and profitable advertising agent as possible. We soon came to the realization that the game is highly dynamical, and there is no one simple winning strategy. In order to survive the game we needed to take into account a variety of factors such as our competitor's behavior and strategy and react accordingly. Therefore we have developed several strategies for the bidding aspect of the game as well as the Bid-Bundle part of it. Some of these strategies had different motivation behind them such as Cost efficient, Minimum and Maximum strategies and came into play as the game progressed. We have gone through a long learning process in which we competed against ourselves and rival opponents and came to final and satisfying design to our agent, which we believe is a highly efficient one.

2. Strategies

Our strategy evolved during the development and practicing with other agents, we created an arsenal of strategies and behaviors so our agent can adapt to different scenarios. We also came to conclusion that the game has 3 parts – the beginning (first 7 days), the middle (day 8-45) and the end game (day 46-60).
The first 7 days are critical because every agent has an initial campaign and therefore the competition on impressions is very hard. Agents who do not finish the majority of their initial campaign impressions will get their rating down and they will have hard time to recover. Therefore our agent is more aggressive to get impressions and also our agent prefers to take campaign that has length larger the 5 days so we can use them to recover after the first 5 days. In the middle of the game we try to cost efficient and finish our campaign with low cost and also maintain a decent quality rating in order to survive to the end game. Agents who survive to the end game (got a decent quality rating) has great chance to increase the revenue since in the end game there are less agents who bid

early strat

mid strat

Segmenting strats:
- consider segmenting based on Qualit

(Some agents rating fall too low to bid) and therefore second bid prizes are higher and also the random effect is stronger and any agent has more chance to get campaign for a maximum bid. Our strategy is to give more percent of the opportunity campaign a maximum value and get lucky. If we get a campaign like that we have a huge budget for it and we can finish it easily and get great profit for it. We also learn between games if there are campaigns that were not finish because of low bids next game we will higher our bids. Also if we see we're having a difficult time to finish a campaign we will not take campaign that has low budget.

2.1 Campaign opportunity bid strategy

Each bid of opportunity bidding can be one of the following strategies. We give a probability for each to be used according to the pending campaign, the current situation and the day of the game.

Figure 1: Bid strategy used over game days

2.2.1 Cost efficient bid strategy

Our bid strategy goal, when applying the cost efficient model, is to offer bids that we estimate can guarantee us profit once the campaign is concluded.

Our final bid offer derives from a maximum mathematical function which consists out of two parameters:

They calculate bid to hit reserve price

$$math.max(bid, minBidValue)$$

minBidValue parameter is calculated by the amount of impressions we need to achieve for our ongoing campaigns multiplied by the formula for the minimum reserved price which is represented by the parameter- **qualityRatingOnMinBidFactor**.

$$qualityRatingOnMinBidFactora = 1/currentQualityRating$$

bid parameter is calculated by the following formula:

They estimate cost!

$$\begin{aligned} bid = {} & initialCost \\ & + (pendingCampaign.reachImps * estimatedImpCost) \\ & * (1 + profitPercent) * reachDensityFactor \\ & * activeCampaignFactor * loseStreakFactor \\ & * segmentSimilarityFactor * segmentRarityFactor \\ & * campaignLengthFactor * popularityFactor \end{aligned}$$

The above bid calculation takes into account key parameters in order to determine the desired bid level.

- **pendingCampaign.reachImps** as detailed above.
- **estimatedImpCost:**

  $estimatedImpCost = estimatedImpCost * 0.7 + avgImpCost * 0.3)$

  Is compiled of 70% our estimated cost for a single impression and 30% of the average cost of an impression.

? learning cost based on mixture of dynamic and modeled cost?

- **reachDensityFactor**

$$reachDensityFactor = (1.0 + (reachPerDay/800)/100$$

(? why 800

• seems like a lot of parameters like this must be learnt

A calculation of the amount of impressions we need to achieve on a daily bases, based on our active campaign list and the impression amount of each of them combined.

- **activeCampaignFactor:**

$$activeCampaignFactor = (1.0 + (activeCampaignCount - activeCampaignThreshold) * 0.01)$$

Takes into account the amount of active campaigns we possess. The less campaigns we have, the lower our bid gets. since we desire to achieve the campaign even more.

- **segmentRarityFactor:**

$$segmentRarityFactor = (1.0 + (segmentRarityThreshold - segmentProbabilty)/5)$$

A computation revolving the rarity of the segment.

We compute the rarity of a campaign's segment by uploading the segment probability table which is detailed in the specification for the game, and containing it in a specially made csv file. Once we have determined that the campaign is aimed towards a segmentation with a high probability we are willing to lower our bid in order to get it and vies versa, if the campaign is aimed to a rare segmentation we will increase a bid offer since we estimate it will be harder to gain a profit.

- **LoseStreakFactor:**

$$loseStreakFactor = \text{Math.max}(1.0 - (loseStreak - loseStreakThreshold) * 0.1, 0.7)$$

A certain percent is lowered from the bid offer until we reach a certain minimum.

The logic behind this is that the more games we lose in a row the more desperate we are to achieve a campaign so that the losing streak will end.

- **segmentSimilarityFactor**

$$segmentSimilartyFactor = 1.0 + (otherSameSegCamp - ourSameSegCamp) * 0.02$$

We examine each of our competitors' campaigns segments in order to determine how much we want to offer on a specific campaign. The more competition there is on a segment the less interested we are in winning that specific campaign since it will be harder for us to earn.

- **campaignLengthFactor** • This is an estimate of completion expectancy - can we make a better one
  We check whether the duration of the campaign is lower than 4 days or exceeds 8 days. If it is the latter then we lower our bid, since we have more time to complete the campaign and eventually earn a profit.
- **popularityFactor** • This is the concept of calculating and adjusting to the differences in sample from population of users
  We check the popularity factor and adapt accordingly. We count compute during the game how many users are there, if there are more than 60,000 we are willing to lower our bid, else we rise it.

2.2.1 Minimum bid strategy

Minimum strategy goal is basically to get a campaign at any prize. The game sets a reserve prize which we cannot bid lower than it. Therefore we will bid only epsilon above this reserve price.

Strategy bid calculation:

$$bid = 1 + \left(\frac{pendingCampaign.reachImps}{10}\right) * \left(\frac{1}{currentQualityRating}\right)$$

The bid value presented above is guaranteed to be accepted by the game server as we've a quality rating above ~ 0.25 .

2.2.1 Maximum bid strategy

Maximum strategy goal is to get a campaign only with the maximum prize allowed by the game's preserve high bound prize. Any bid in the game has a chance of 30% to win a campaign at random, regardless to the bid value. We're counting on that to happen to us when we bid a maximum value bid. If we success winning a campaign in that way we can achieve an advantage over other agents who struggle to finish low budget campaigns while we can finish a campaign with a large budget with a rewarding profit.

So seems like an important trick
I don't understand

Strategy bid calculation:

$$if\,(currentQualityRating \leq 1)$$

$$bid = (pendingCampaign.reachImps) * \left(\frac{1}{currentQualityRating}\right) - 1$$

$$else\ bid = pendingCampaign.reachImps - 1$$

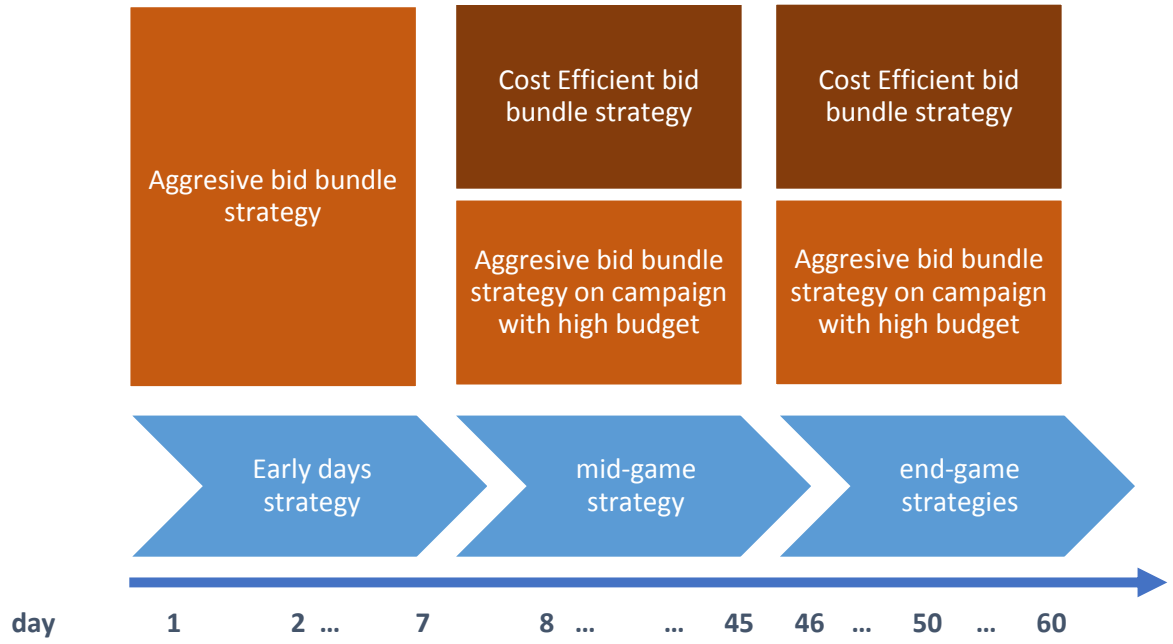
The bid value presented above is guaranteed to be accepted by the game server as we've a quality rating above ~ 0.25 .

2.2 Bid bundle strategy

At the first days of the game we use aggressive strategy because there's a hard competition on impressions (high demand because every agent has initial campaign) after those days we prefer to use cost efficient strategy on low budget campaign and aggressive strategy on high budget campaign (high budget is considered budget that we received from a maximum bid or minimum bid but the second bid was very high). Our mind set behind this strategy thinking is that if we have a low budget campaign we need to very cost efficient in order to make profit while high budget campaign have much more budget than it really requires to finish them, we want to be confident that we finish them and also damage other competitor agents campaign completion.

Don't really understand logic here. Don't you always want to be cost efficient?

Figure 2: Bid Bundle strategy used over game days



2.2.1 Cost efficient bid bundle strategy

Our bid bundle strategy goal is for every campaign to finish its impression reach while minimizing the cost. Further than finish it, if we can buy more impressions with low cost in we'll do it so we can increase our quality rating. Our strategy is based on the approach of competition environment awareness and campaign time space. In details, for each active campaign that still has targeted impression left, in our bid bundle for that campaign, we are taking into account various parameters that affect the demand for the campaign impression. Farther more, we also conclude from the time left for the campaign if we can try to send low value bids and increase the bid each day until all targeted impression are done.

Each day that a bid bundle is sent, for each one of our active campaigns, which mean that any one of our campaigns that meet the requirements of $campaign.dayStart \geq day + 1 \ \ and \ \ campaign.endDay \leq day + 1 \ \ and \ campaign.impToGo() > 0$ we'll calculate the bid for its impressions.

Bid bundle bid calculation function:

again important to recognise concepts of revenue/profit over just minimising CPC or imps

$$bid = 1000 * f(averageRevenuePerImp \uparrow, segmentRarityFactor \uparrow, randomFactor, minimumProfitPercent, impressionCompetitionFactor \uparrow, daysLeftFactor, populationFactor \downarrow, prevCampaignCompletionFactor)$$

** Factors are situational parameters that their value represents a percent.*

Properties and factors description and calculation methods:

- ***averageRevenuePerImp*** the average revenue for a single impression of a campaign. Each impression bid value should not extend the average revenue per impression to prevent any option for negative income for a fully finished campaign.

  $$averageRevenuePerImp = \frac{campaign.budget}{campaign.reachImps}$$

  Base estimation of impression value

  The thought behind this parameter calculation:

  We wanted some value to be the basic bid from which we will make our bid higher or give it a "discount".
- ***segmentRarityFactor*** is based on the distribution of the segment in the population. We define rare as relative to other segments probability to occur. It mainly calculated by the segment probability defined the game specification.

  $$segmentRarityFactor = (1 + (segmentRarityThreshold - segmentProbability))$$

  - $segmentRarityThreshold$ is the average probability.
  - $segmentProbability$is the segment probability.

  The thought behind this parameter calculation:

- ***randomFactor*** a random effect to bid. A number between 0 to 1. The random factor lower bound (which limits the random number from the bottom) is change accordingly to situation.

  x They don't explain their logic behind including random factor.

  ↳ in a game where impressions are bid on seperately I understand this more as an minimise risk & increasing variance } BUT I feel this is of much more limited value when there are only 60 bids.

$$randomFactor = \max\left(0.8 - \left(\frac{segmentProbability}{4}\right), random(0,1)\right)$$
$$if\,(daysLeft = 2)\ and\ (campaign.impToGo() > 0.5 * campaign.reachImps)$$
$$randomFactor = \text{random}(0.9,1) \quad \text{// limit the random affect}$$
$$if\,(daysLeft = 1)\ and\ (campaign.impToGo() > 0.5 * campaign.reachImps)$$
$$randomFactor = 1 \quad \text{// we make the random as constant}$$

- ***minimumProfitPercent*** is a percent that represent the minimum profit percent that we're aiming not profit less from. ?
  - As I understand it this is their concept of risk management.
- ***impressionCompetitionFactor*** represents how much impressions are required by active campaigns owned by rival competitors.

$$impressionCompetitionFactor = 1 + 0.01 * (impCompetitionCount * 1000)$$

- ***daysLeftFactor*** represents the effect of days left for the campaign to finish. The more days that are left the factor value is lower but when the campaign is about to finish and there's only few days its value goes high.

$$daysLeftFactor = 1 - \left(daysLeft - \underbrace{dayLimitPresureThreshold}_{=2}\right) * \underbrace{dayReliefPercent}_{=0.1}$$
$$if\left(daysLeft - \underbrace{dayLimitPresureThreshold}_{=2} > 9\right)$$
$$daysLeftFactor = 0.1$$
$$if\,(daysLeft = 2) \text{ // last days we want to buy any impressions left}$$
$$daysLeftFactor = 2.5$$
$$if\,(daysLeft = 1)$$
$$daysLeftFactor = 3$$

- Not sure about this, although it is valuable to increase bid as it looks less likely campaign will succeed it seems like basing that concept on only days left is not ideal.

- ***populationFactor*** represents the effect of the population in the game. During the development of our agent the server parameters change and the

- This is the idea that since competition is for a pool of users the size of the pool matters.

population decreased in a drastic manner. We decided to check the size of the population as it is received in the reports and if we notice a low population size we increase our bid with the population factor.

- $\boldsymbol{prevCampaignCompletionFactor}$ represent the effect of the average of previous campaign completion percent. If we're not fulfilling campaigns impression reach than we're adapting and increasing the bid bundle bids.

$$if\,(prevCampaignCompletionPercent < \underbrace{badCompletionThreshold}_{=1})$$

$$prevCampaignCompletionFactor = 1 + (1 - prevCampaignCompletionPercent) * 2$$

This seems inelegant but useful.

Maximum bid limit

prevent weirdness

We're aware that our bid may go high more than expected. Therefore we're regulating it with a maximum value. In the first days of a game we allow a bid to more aggressive and as the days go by we limit it to a more profitable maximum bid (for example not more than the average impression revenue).

2.2.1 Aggressive bid bundle strategy

Idea that ideal parameter values are time dependent.

The same as cost efficient bid bundle strategy except there's no maximum bid and also some fixed parameter are higher in order to take any impression we need to finish the campaign reach and even more in order to increase our quality rating.

We use this strategy in the early days of the game (until day 10~). In the beginning of the game any agent gets an initial campaign and agents who do not complete their initial campaign find themselves with a low quality rating which prevents them from getting new campaign in the future of the game. We decided it's our first priority to finish the first days of a game with a full campaign and use this advantage to get more campaign than other agent. If the cost of the first campaign is not profitable, we can recover by taking more campaign because we have higher quality rating.

2.3 USC bid strategy

2.3.1 Regression USC bid strategy

This class is in charge with finding the best price to bid in order to win a desired UCS level for our agent.

In order to find the best bid price, this class uses data that was collected on the UCS by the class `UCSLevelDataHistory` and calculates the averages of the Prices to bid and UCS levels from the past games and days. These actually gives us the Expectation of those variables.

Definitely runs multiple games then.

In order to find the best bid price this class offers 2 options:

This is what I want to do too, have a target wins from the campaign agent and other agents try to reach that target

1) Recommended price: given a UCS level as a target the method `getRecommendedPrice` will return the multiplication of the current average price by the factor of 1 plus the difference between the expectation of the UCS level and the required UCS level target. In this way the bigger the difference between them, the further we bid from our average offer.

2) Price from regression: this option proved favored after test runs and in trainings. The method `getPriceFromRegression` is given a UCS level as a target and according to the current day, and game number returns the best price in one of 2 ways:

a. Constants – in the first game (marked game 0 in our data list) sufficient data was not yet collected until at least 20 days have passed. There are not enough observations to perform a proper regression. In this case we use constant prices we found during training (mostly against dummy agents). For each UCS level (they are set by powers of 0.9 in this tournament) we found a suitable price after running that level as target for multiple games. Those constant we be used as best Prices in the beginning of the tournament.

Weights the constants from previous games vs. variable values from this game.

b. Linear regression: our model $USCLevel = \beta_0 + \beta_1 Price\_Bid_i + \varepsilon_i$

In each game (except the first) if 20 days have not yet past, the strategy is to use the averages calculated the previous game (otherwise use the current game's averages). The method finds estimators to β_0 (marked in the class by a) and β_1

(marked in the class by b) by calculating variance and co-variances for the variables. In this way out model explains the how the bid price affects the USC level. In order to predict the best bid price that will achieve the requested UCS target level we use $USCTarget = \beta_0 + \beta_1 * PriceToBid$ and return the best price from regression: $PriceToBid = \frac{USCTarget - \beta_0}{\beta_1}$

2.3.1 Custom USC bid strategy

Before we've enough observations in order to use the regression we use a heuristic algorithm.

$$ucsBid = \min\Big(0.12 + 0.19 * ucsTargetLevel + (0.1 - day * 0.016), 0.12 - day * 0.01 + prevUcsBid * \big(1 + (ucsTargetLevel - ucsLevel)\big)\Big)$$

2.4 USC Target strategy

We set our UCS target according to our need and the situation. Based on the parameter of how much impression are needed for our active campaigns and also depend on the game day.

```
if(liveImpToGo == 0)
{
        ucsTargetLevel = 0;
}
else if (liveImpToGo < 5000)
{
        ucsTargetLevel = 0.65;
}
else if (liveImpToGo < 15000)
{
        ucsTargetLevel = 0.75;
```

```
}
else
{
        ucsTargetLevel = 0.85;
}

if((day+1)<=5) // in the first 5 days the comptetion is huge. Any agent has a campaign.
{
        ucsTargetLevel = 0.95;
}
```

3. Performance results & analysis

Agent Giza showed high performance during the competition. We delivered an agent that can adapt to many situations, rivals and random effects.
We learned through the agent development and while practicing that there are several strategies that are at the top chain and each kind of these tactic has a counter or it fits to a specific situation or rivals.

Verg Important !!!

We collect any data sent and receive in the game and at the end of any game we output it into a file 'game[Number].txt' who is organized for a human eye to understand how the game went for our agent at any aspect of the game. In the file all the campaign we received with details of their budget, cost, targeted impression we got out of the impression reach required. All the days bank status, quality rating level, USC level and much more. It helped us a lot analysis our performance. We had 2 demo competitions versus other teams from TAU and 3 TAC international competitions; we'll review those competitions and explain what was our strategy, what we conclude from each competition and what was the performance results.

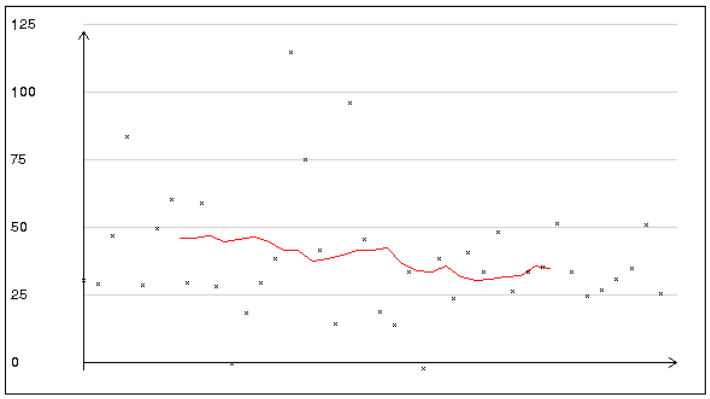
Demo competition 1

In this competition our main strategy was to be cost efficient both in bid on new campaign opportunities and on bid-bundle bids. We succeeded to take the majority of the campaign in each game and finish their impression reach with low

cost bids. We took only campaigns that were to fit our prize policy and only profitable ones. This strategy is weak against aggressive minimum bidders (bids most of the time minimum bid) but strong against other strategies such as maximum bidders (bids most of the time maximum bid). We are proud to receive high performance in this competition and get to win it as #1 agent.

Position	Agent	Average Score	Games Played	Zero Games
1	giza	35	28	0
2	tau	30	28	0
3	ibm	11	28	0
4	agent00	3	28	8
5	Agent2	1	28	2
6	OOS	-6	28	23

Game	Score	Game	Score	Game	Score	Game	Score
335	24.35	336	28.88	337	27.07	338	-0.09
339	6.73	340	20.22	341	35.36	342	48.35
343	33.56	344	37.98	345	28.40	346	26.59
347	14.39	348	34.60	349	43.75	350	51.20
351	10.21	352	72.34	353	19.80	354	34.68
355	48.30	356	76.68	357	163.37	358	7.85
359	55.25	360	36.91	361	18.32	362	1.65

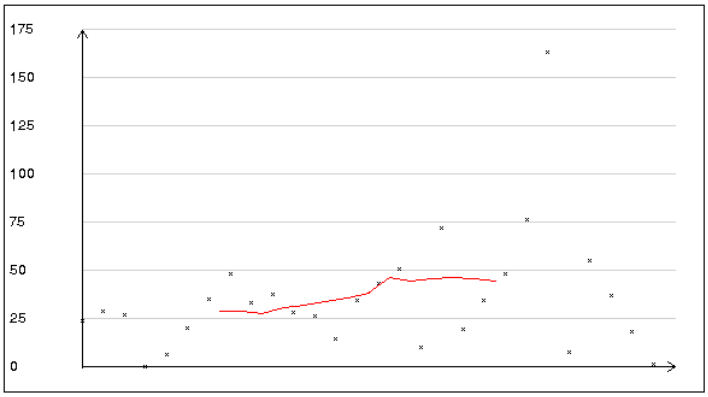


Demo competition 2

In this competition our main strategy was to be aggressive with minimum bids and also aggressive with getting a lot of impressions with low cost (in order to raise our rating as high as we can). Our set of mind is, we won't to take as much as we can campaigns and starve our opponents without any campaign to profit from. In order to overcome other minimum bidder we try to raise our rating as much as possible giving us the opportunity to bid on campaign lower than they do. This strategy is weak against aggressive bid bundle bidders who will set high bids on impression – it will cause anyone who take campaign with minimum bid to have hard time to complete them and in time the rating will decrease and they wouldn't be able to set a bid in the bounded prizes. We are glad our strategy worked very well and received high performance in this competition and got us to win it as #1 agent.

Position	Agent	Average Score	Games Played	Zero Games
1	giza	38	40	0
2	tau	20	40	5
3	Agent2	15	40	0
4	OOS	6	40	17
5	ibm	5	40	0
6	livadx	0	40	40
7	agent00	0	40	40

Game	Score	Game	Score	Game	Score	Game	Score
563	30.47	564	29.23	565	46.91	566	83.93
567	28.90	568	49.85	569	60.61	570	29.75
571	59.11	572	28.43	573	-0.62	574	18.54
575	29.59	576	38.63	577	114.99	578	75.15
579	41.83	580	14.46	581	96.37	582	45.86
583	18.98	584	13.86	585	33.78	586	-2.48
587	38.76	588	23.85	589	40.87	590	33.79
591	48.65	592	26.55	593	33.85	594	35.44
595	51.54	596	33.52	597	24.57	598	27.14
599	31.11	600	35.18	601	51.27	602	25.60



TAC competition

As we practice versus other agents from abroad we came to realization that we need to be more stable and adaptive towards various types of strategies. On one hand we want to finish campaigns and on the other hand if we insist to finish a campaign at any cost we may get our profit to a negative number which we may not recover. We also came to conclusion that the game has 3 parts – the beginning (first 5 days), the middle (day 6-40) and the end game (day 41-60). The first 5 days are critical because every agent has an initial campaign and therefore the competition on impressions is very hard. Agents who do not finish the majority of their initial campaign impressions will get their rating down and they will have hard time to recover. Therefore our agent is more aggressive to get impressions and also our agent prefers to take campaign that has length larger the 5 days so we can use them to recover after the first 5 days. In the middle of the game we try to cost efficient and finish our campaign with low cost and also maintain a decent quality rating in order to survive to the end game. Agents who survive to the end game (got a decent quality rating) has great chance

to increase the revenue since in the end game there are less agents who bid (Some agents rating fall too low to bid) and therefore second bid prizes are higher and also the random effect is stronger and any agent has more chance to get campaign for a maximum bid. Our strategy is to give more percent of the opportunity campaign a maximum value and get lucky. If we get a campaign like that we have a huge budget for it and we can finish it easily and get great profit for it.

The results:

We are very proud to finish TAC as #2 agent! ☺

TAC Competition 1

Position	Agent	Average Score	Games Played	Zero Games
1	anl	16	40	0
2	Agent2	14	40	3
3	giza	13	40	0
4	blue	4	40	1
5	livadx	3	40	0
6	WinnieTheBot	0	40	0
7	Amunra	-1	40	38
8	tau	-1	40	1

Game	Score	Game	Score	Game	Score	Game	Score
870	7.63	871	2.38	872	17.81	873	3.17
874	0.80	875	4.25	876	16.04	877	9.81
878	-0.40	879	16.57	880	36.12	881	-3.52
882	3.16	883	11.49	884	2.50	885	10.79
886	29.44	887	0.30	888	-14.04	889	-24.21
890	-6.03	891	-1.02	892	-2.82	893	-0.06
894	3.46	895	91.15	896	32.60	897	5.95
898	87.64	899	27.09	900	-12.81	901	16.31
902	18.45	903	34.16	904	-6.77	905	7.90
906	13.59	907	34.81	908	17.52	909	30.87

TAC Competition 2

Position	Agent	Average Score	Games Played	Zero Games
1	anl	29	40	0
2	tau	15	40	0
3	giza	13	40	0
4	livadx	5	40	3
5	Agent2	5	40	0
6	WinnieTheBot	4	40	0
7	blue	2	40	3
8	Amunra	0	40	39

Game	Score	Game	Score	Game	Score	Game	Score
975	19.73	976	22.15	977	8.87	978	12.18
979	6.44	980	28.31	981	10.92	982	19.48
983	5.23	984	13.77	985	94.96	986	4.73
987	-8.14	988	14.61	989	-15.48	990	7.86
991	31.97	992	7.29	993	28.87	994	2.95
995	2.93	996	34.61	997	1.22	998	-2.97
999	-2.38	1000	30.88	1001	-8.67	1002	-3.30
1003	36.62	1004	-8.35	1005	-9.54	1006	-2.96
1007	-2.72	1008	47.05	1009	29.06	1010	25.13
1011	23.56	1012	11.99	1013	18.07	1014	17.47

TAC Competition 3

Position	Agent	Average Score	Games Played	Zero Games
1	anl	31	40	0
2	giza	21	40	0
3	Agent2	16	40	2
4	tau	13	40	1
5	livadx	5	40	1
6	blue	1	40	23
7	WinnieTheBot	0	40	33
8	Amunra	0	40	40

Game	Score	Game	Score	Game	Score	Game	Score
1069	35.67	1070	3.89	1071	25.87	1072	44.19
1073	-12.38	1074	24.38	1075	120.82	1076	61.90
1077	-1.34	1078	13.37	1079	18.42	1080	26.79
1081	-3.93	1082	4.28	1083	-20.24	1084	8.54
1085	1.71	1086	33.41	1087	29.13	1088	51.72
1089	32.18	1090	-9.80	1091	3.39	1092	34.24
1093	1.90	1094	17.20	1095	39.72	1096	15.39
1097	23.52	1098	9.16	1099	115.36	1100	16.43
1101	6.27	1102	7.58	1103	-7.01	1104	-0.48
1105	33.25	1106	-23.64	1107	25.40	1108	46.06