**Ad exchange game workshop**

**Tel-Aviv University 201**6

**Mor Chen**

**Ran Feder**

**Lee Hellow**

**Nofar Socher**

# **Introduction**

**Now days, advertisements are everywhere, and are one of the largest source of income sites and apps have.**

**As a result of the growing advertisement market, new technology called ad exchange was developed in order to find** efficient way for trading advertising opportunities.

Whenever a user visits a publisher's web page, the ad exchange conducts an auction for the ad among relevant ads from ad networks.

The winning ad is then displayed and the corresponding ad network is charged.

**In this workshop we have simulated advertising agent, which in each day of the game bids to win advertising campaign contracts, and submits a bidding strategy to the Ad Exchange, while the main purpose of the game is to gain the biggest profit.**

# **Game elements**

**The game has three main components:**

* **Campaign Opportunity auction – Our agent bids to win advertising campaign contracts, we need to bid low enough so that the publishers would agree to pay us and get campaigns, but not too low so we could make profit from the campaign..**
* **User Classification Service auction – Our agent bid to get the highest UCS level in order to get the best quality of matching, the higher our UCS level will be the more accurate information we get.**
* **Bid bundle auction – Our agent bid to get as many impressions he can in order to finish the campaign reach impressions and get high rating.**

## **Campaign Opportunity**

## **User Classification Service**

## **Bid Bundle**

### **Bid parameters**

For each bid bundle we collected data in order to build our bid. The parameters are:

* - This is the average revenue our agent get for each impression, most of the time we will use this parameter as our higher bound for our bid.
* Factor of the ad type and device coefs.
* – Factor of how many days left for this campaign - the less days left, the more we want to get impressions to finish the campaign so we bid higher.
* – This parameter will tell us about our state in the game, we calculate the ratio between the campaign impressions state and days left state.

If the ratio is low meaning our progress is good, otherwise our progress is not that good and we need to be more aggressive and get more impressions

* – Based on the User Population Probabilities table from the spec, this is factor of how popular the campaign market segment is.

If the market segment is big, meaning there are allot of potential impressions from this segment, so our bid will not change.

But if the segment is rare, meaning there are not many impressions from this campaign market segment so we bid will higher for each impression.

* – The bid random factor depends on the parameter. The parameter tells us about our progress in the game, if our progress is good then the is low, and the random factor will be lower, otherwise our progress is not that good and we give higher random factor to enlarge our bid and get more impressions
* - Factor of how many campaign are running at the current day and have the same market segment as our campaign - the competition we have for each impression.

### **Bid strategy**

We decided to create three strategies for each part of the game –

* Stable bid strategy– this is the strategy we use at most days of the game, we build the bid as a function of all the data parameters we collected.
* First day's strategy – at the first 12 days of the game we decided to be more aggressive and bid higher to get impressions in order to finish campaigns and get high rating. Our strategy is to calculate the stable strategy and enlarge it by multiplying it with random number bigger then 1.
* Last day`s strategy - At the last 8 days of the game we don't care anymore about our rating, and we try to get as much profit as we can, so we bid a lower than usual and try to get impressions at lower cost.

### **Bid Knn algorithm**

As a part of our strategy, we wanted to learn from previous games in order to build better bid bundle, so we implemented the K nearest neighbor's algorithm.

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions).

We collected data from previous games about our campaigns and bids, and based on this data we calculate the Knn factor which is the average bids from previous successful bid bundles.

If there are enough of good and similar previous campaign to our current campaign we calculate the knn factor and calculate it as 5% of our bid.

# **Architecture and Data collection**

# **Game results**