

Wodecki, Andrzej

Article

The reserve price optimization for publishers on real-time bidding on-line marketplaces with time-series forecasting

Foundations of Management

Provided in Cooperation with:

Faculty of Management, Warsaw University of Technology

Suggested Citation: Wodecki, Andrzej (2020) : The reserve price optimization for publishers on real-time bidding on-line marketplaces with time-series forecasting, Foundations of Management, ISSN 2300-5661, De Gruyter, Warsaw, Vol. 12, Iss. 1, pp. 167-180, <https://doi.org/10.2478/fman-2020-0013>

This Version is available at:

<https://hdl.handle.net/10419/237013>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.



<https://creativecommons.org/licenses/by-nc-nd/4.0>

THE RESERVE PRICE OPTIMIZATION FOR PUBLISHERS ON REAL-TIME BIDDING ON-LINE MARKETPLACES WITH TIME-SERIES FORECASTING

Andrzej WODECKI

Warsaw University of Technology, Faculty of Management, Warsaw, POLAND

e-mail: andrzej.wodecki@pw.edu.pl

Abstract: Today's Internet marketing ecosystems are very complex, with many competing players, transactions concluded within milliseconds, and hundreds of different parameters to be analyzed in the decision-making process. In addition, both sellers and buyers operate under uncertainty, without full information about auction results, purchasing preferences, and strategies of their competitors or suppliers. As a result, most market participants strive to optimize their trading strategies using advanced machine learning algorithms. In this publication, we propose a new approach to determining reserve-price strategies for publishers, focusing not only on the profits from individual ad impressions, but also on maximum coverage of advertising space. This strategy combines the heuristics developed by experienced RTB consultants with machine learning forecasting algorithms like ARIMA, SARIMA, Exponential Smoothing, and Facebook Prophet. The paper analyses the effectiveness of these algorithms, recommends the best one, and presents its implementation in real environment. As such, its results may form a basis for a competitive advantage for publishers on very demanding online advertising markets.

Keywords: online marketing, real-time bidding, reserve price optimization, machine learning, forecasting.

JEL Classification: C53, C57, M37, M39.

1 Introduction¹

Internet advertising has become a multi-billion dollar industry in recent years. According to Zenith Agency (Austin, Barnard, and Hutcheon, 2019), global digital advertising spend in 2019 exceeded 290 billion US dollars, with more than 320 billion US dollars expected in 2020.

Advertisers may purchase the so-called ad slots both through direct contracts (non-programmatic) or by means of real-time auctions (programmatic marketing)². According to (Austin, Barnard, and Hutcheon, 2019), programmatic marketing constituted for 65% share of digital spend in 2019 and is expected to grow to 69% in 2020.

Real-time bidding (RTB), as a way of trading digital ads through real-time auctions, is the most popular mechanism of programmatic marketing. On the RTB market, one can, in a nutshell, distinguish between the demand side (organizations interested in purchasing

advertising space), the supply side (providing this space), the market organizer, and the companies supporting the transaction in various ways. In practice, in each of these segments, there are many actors with different business models and target functions. A simplified market diagram is presented in Fig. 1.

A typical transaction process on the RTB market is as follows. An Internet user visits one of the advertising spaces displayed by the Publisher, most often through a web browser or mobile application. The Publisher passes this information (so-called bid request: (1)) to the Supply Side Platform (SSP) enhanced by a description of the available advertisement inventories.

The SSP platform (there may be many of them on the market) transmits this information to Ad Exchange, which initiates the auction: it distributes information about inventories, enriched with user data (demography, behavioral profile) obtained from Data Management Platforms (DMP), among many Demand Side Platforms (DSP) waiting for the Advertisers' offers.

¹ This work was supported by Europejski Fundusz Rozwoju Regionalnego (European Regional Development Fund), grant number: RPMA.01.02.00-14-9523/17.

² An interesting overview of on-line advertising industry can be found, for example, in (Choi, et al. 2019).

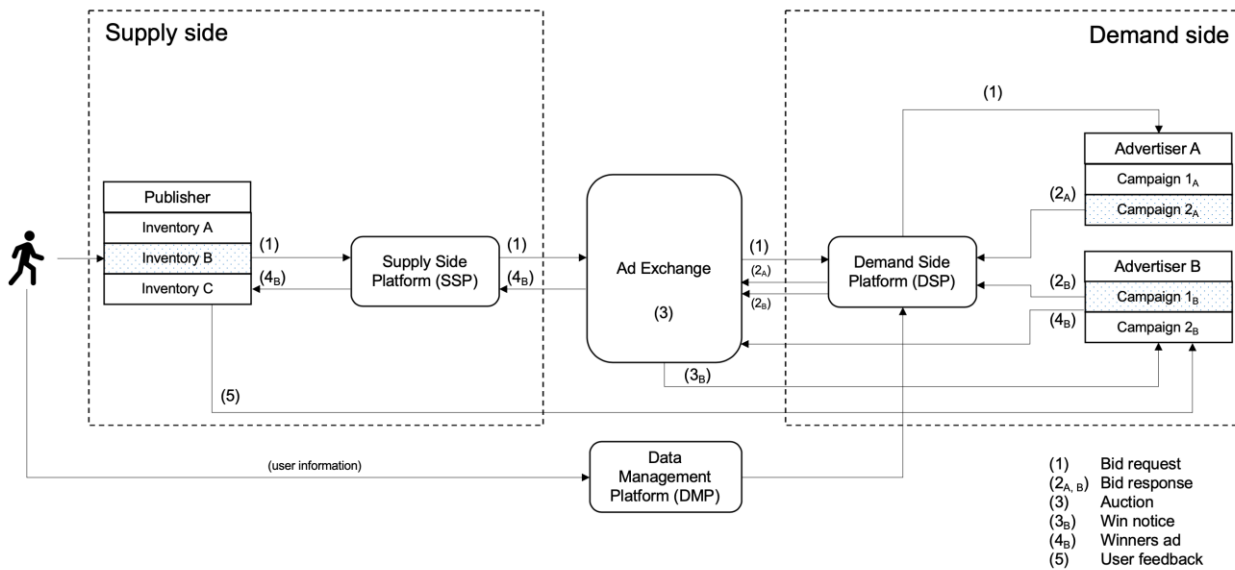


Figure 1. Online advertising ecosystem (*Source:* Own elaboration)

On the DSP, the bid request is initially adjusted to the previously registered space and user requirements (advertising campaigns). Advertisers receive bidding data and, most often using the algorithms implemented in their systems, specify the price they are willing to pay (so-called bid response: (2)). Bid responses are transmitted, via the DSP, to the Ad Exchange, where the auction is finalized (3), and the so-called win notice is transmitted to the Advertiser.

The Advertiser passes his advertisement (via DSP, Ad Exchange, and SSP) to the Publisher, who emits it on the indicated inventory. After the emission, its details, in particular information about the user feedback (5), are forwarded to the Advertiser. The transaction is finalized, the auction ends. The whole process (from the user entering the Publisher website to the display of the advertisement) takes about 0.1 sec.

The RTB market can be compared to a multi-player game, each with its own goal function. The Publisher wants to maximize profits from advertisements, which requires pricing to be as high as possible on the one hand, and competitive with other publishers on the other.

The Advertiser must also bid intelligently: offer prices that minimize the cost of acquiring a customer, but are higher than those of other advertisers.

This is hindered by the lack of information transparency in the market:

- 1) DSP and SSP platforms are its active participants: DSPs often buy space en masse from publishers and only bid on known prices;
- 2) advertisers do not have information about the auction sale price they have not won; and
- 3) publishers do not know all the prices offered by the advertisers at the auction.

No wonder that all market participants are actively looking for solutions (methods, algorithms, technologies) that would optimize their goal function. In this article we focus on the Publisher perspective, the Advertisers' one briefly described (for the reference in the literature review section).

The main goal is to identify the so-called reserve (minimal) price optimization strategy, which is not only efficient, but also possible to implement in the mid-size RTB marketing agency. Present strategies (presented in different research papers displayed below) in most cases try to forecast advertisers' strategies to maximize the profit from a given advertisement emission, what 1. makes it computationally challenging for small RTB agencies and 2. doesn't take into account other Publishers' goals like optimal ad space utilization (measured by so-called Coverage parameter).

In this paper, we recommend significantly different heuristics: instead of guessing the add value for the advertisers, we use advanced time-series algorithms to short-term (3 days) forecasting of the Coverage parameter and adopt reserve price to keep this parameter on the maximal possible level. Thus, it is not only computationally efficient (feasible to retrain the model on a daily basis), but also easy to understand and interpret by RTB agency consultants. The strategy presented below may increase competitive advantage of such agencies and increase its customers' (Publishers') profits.

After a brief presentation of the state of the research, in the main part of the work, we present the method of optimization of reserve prices at publishers using the modelling of time series of key space utilization parameters. We show the effectiveness of various models and heuristics created in cooperation with Yield Riser Ltd.: a company representing Publishers in the RTB market. The whole is summarized by recommendations for further research.

2 Literature review

Each of the participants of the RTB market described above has its own function of purpose, which it tries to maximize. Due to information asymmetry and the complexity of auction mechanisms, this raises many challenges. Access to data and the high practical importance of the problem motivates researchers to formulate and test hypotheses that help to achieve a competitive advantage in these markets. Below we will present an outline of current research challenges and the attempts to solve them both from the publishers' and advertisers' perspectives.

2.1 Advertisers' perspective

The main goal of advertisers is to buy the best possible advertising space at the lowest possible price. In most cases, this purchase is part of a broader advertising campaign having some specific goals (product, audience, etc.), budget, and lead time. The Advertiser, after receiving information about the currently available advertising space and the visiting client (bid request enhanced by user information data), must very quickly decide on the participation of the auction and

determine the price she will be willing to pay. It does so under conditions of incomplete information: it does not know both the minimum (reserve) price set by the Publisher and the prices offered by other Advertisers. In second-price auctions, only the winner of the auction knows the selling price - the other bidders only know that someone has offered a higher price, which is called right-censored data.

One of the most popular strategies to deal with such cases is the so-called landscape forecasting, which involves forecasting the probability distribution of prices offered by the market for each auction. There are many approaches to solving this problem. One group of methods based on survival analysis in medical science, taking into account only the losing logs (cf. Wang, et al., 2016; Wu, Yeh, and Chen, 2018; Wu, Yeh, and Chen, 2015). Ren, et al. (2019) in turn use survival analysis methods for censorship handling, and recurrent deep neural networks for modeling conditional winning probability with respect to each bid price.

Another approach, based on reinforcement learning (RL), is described in (Cai, et al., 2017). The Algorithm representing the Advertiser (called *Agent* in RL framework) at each timestep observes the campaign's parameters (remaining lifetime, budget) and the bid request for a specific ad impression (enhanced by user data). With such a set of information (called *State*), the *Agent* decides upon the bidding price and performs *Action*. After the auction, it receives *Reward*, which is derived from auction results and user feedback (in case of winning the auction). Finally, basing on tuple $State > Action > Reward$, *Agent* updates its *Policy*. After many iterations, *Agent* learns optimal policy for a given campaign. The main drawback of this approach lies in the fact, that in many cases, RL algorithms require thousands of iterations to learn optimal policy, which takes time and consumes a lot of computing power.

The most common approach to solve this issue is to simulate the RTB market with historical data, but that approach can't be implemented in real-life applications (*Agent* trained on historical data will probably not-optimally behave in new, real situations).

Another challenge addressed in contemporary research on the optimization of space purchase price de-

termination processes on RTB markets is the high dynamics of these environments. As a result, the process of making purchase decisions cannot be treated as a stationary optimization problem. This problem is addressed by many groups of researchers (Cai, et al., 2017; Lu, et al., 2019; Yang, et al., 2020), as a solution proposing various reinforcement learning options.

The perspective of advertisers presented above is often the basis for the creation of price optimization strategies by publishers, which will be briefly presented later in this work.

2.2. Publishers perspective

The main objective of the Publishers is to maximize the profit from the sale of their advertising space, both in off-line channels (e.g., via direct contracts) and on-line (e.g., on RTB markets). For each new issue, they must make an immediate decision on the distribution channel and the minimum price level. Too low a price generates a risk of loss of profit, too high - a risk of no emissions (Li, Ni, and Yuan, 2018). In second-price auction systems, the situation is further complicated by the fact that there are many advertisers in the market offering very low purchase prices. This motivates the creation of a strategy for setting reserve prices, protecting publishers from practically free access to their space.

The strategy of setting reserve prices is very important, not only for publishers (who are guaranteed a certain level of selling prices for advertising space), but also for the entire RTB ecosystem (determining the balance between supply and demand) (Li, et al., 2017). Publishers, like advertisers, do not know the behavior of their competitors (other publishers) and potential customers. The situation is complicated by the fact that advertising space is sold both on-line (currently dominant) and off-line (direct B2B sales). As a result, the Publisher must set separate reserve prices for both channels, but also decide very quickly which channel to offer given advertising space.

Researchers have for years been dealing with the issue of optimizing reserve prices in auction systems (Myerson, 1981; Riley and Samuelson, 1981). The most common current reserve selling price strat-

egies focus on the forecasting of the advertisers' pricing strategies (Li, et al., 2017). Other methods apply iterative price change in the direction of the gradient of an appropriately constructed Lagrangian relaxation (Radovanovic and Heavlin, 2012).

A similar, iterative, real-time approach to price optimization, was proposed by Austin et al. (Austin, et al., 2016). In turn, Li et al. investigated the theoretical and empirical methods of the impact of reserve price levels on the profits of publishers offering their space simultaneously in multiple channels (Li, Ni and Yuan, 2018).

3 Research problem

As can be seen, there are many strategies for determining the purchase and sale prices of advertising space in RTB markets. Most of them focus on "revenue" indicators, which can be calculated from parameters such as Ad Request eCPM or Estimated Revenue. It turns out, however, that effective management of Publisher's advertising space cannot focus only on profit: it is also very important that the space is used to its maximum extent, which is measured by the so-called Coverage ratio.

Next, strategies based on advertisers' pricing policies in most cases (especially when supported by Reinforcement Learning algorithms) consume significant computing power to (re)train algorithms and make predictions. Moreover, in our approach, we should take into account market dynamics. And, finally, every decision supporting system needs to be well understood by its users (in our case RTB consultants) which is not the case for advanced market simulations systems based on, e.g. deep learning algorithms.

Therefore, the research problem boils down to proposing such a strategy setting reserve prices for publishers, which satisfies the following conditions:

- 1) achieves high margins on particular ad emissions while taking care of the level of advertising filling of the whole site (measured by Coverage parameter);
- 2) treats and RTB market as a dynamic environment (real-time dynamics);
- 3) is easy to understand and interpret by RTB consultants responsible for setting reserve price levels;

4) is computationally efficient: a model for hundreds of inventories can be re-trained on a daily basis on reasonably priced servers.

The method of determining reserve prices described below was developed in cooperation with Yield Riser Ltd (YR). YR operates on a Supply Side of RTB Market, and manages its' customers (Publishers) portals via Google AdManager (GAM). Each day YR consultants monitor the efficiency of inventories (called AdUnits) analyzing a lot of available data, but focusing on three of the most important efficiency measures: Estimated Revenue (an estimation of revenue from a given AdUnit), Ad request eCPM (which measures the income from a given ad placement per 1000 displays) and Coverage (a measure of how efficiently a given inventory is utilized).

In their daily practice, YR consultants create the so-called pricing rules to set the optimal minimal bidding prices for their customers (Publishers) inventories, basing on Ad Exchange reports and personal experience. This "manual" process is time-consuming and prone to errors. The main goal of the project was to automate this process using state of the art, self-optimizing machine learning algorithms.

The main research problem is to formulate a method for determining the reserve prices of advertised space offered by publishers on the RTB market. In the heuristics proposed, time series forecasting of the Coverage coefficient turned out to be crucial, which determined the second of the research objectives: to identify the machine learning algorithm that best predicts this measure. The last goal was to indicate potential directions of research aimed at optimizing the economic efficiency of advertising space sales on the RTB market.

4 Research methods

This research project, implemented in the period from October 2019 to February 2020, according to CRISP-DM methodology (Chapman, et al., 2004), was divided into the following, about 3 months long, stages: *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling and Evaluation* and *Deployment*.

The *Business <> Data Understanding* stages were implemented in parallel. After an exploratory analysis of data from Google Ad Manager (GAM) reports, hundreds of A/B tests were carried out to verify the impact of various reserve price strategies on key indicators (mainly Ad request eCPM and Coverage). They confirmed the conclusions resulting from the game theory: in second-price auctions, the optimal strategy is to set the reserve price at the lowest possible level so that the purchase price is decided by the buyers themselves, through an auction (Wang, Zhang, and Yuan, 2017).

In mid-2019, Google announced a gradual transition from the second-price auction to the first-price auction, which had a strong impact on the strategy for reserve prices optimization. Coverage (a measure of how efficiently a given inventory is utilized) was identified as the key indicator for maximizing customer profits. Next, based on the analysis of GAM reports and YR consultants experience, a reserve price strategy was developed based on the prediction of this indicator for the next three days:

- 1) If the predicted Coverage is higher than 95%: increase the price;
- 2) If the predicted Coverage is between 90% and 94%: hold the existing price;
- 3) If the predicted Coverage is lower than 90%: decrease the price.

The practical implementation of this strategy required the creation of a Coverage prediction model. The first stage was to acquire, analyze, and prepare for modeling the appropriate time series (*Data Preparation*). In the next step (*Modelling and Evaluation*), various time series forecasting algorithms were selected, implemented, and evaluated.

The best one was implemented in the production system with a dedicated Business Intelligence (BI) dashboard, for a given inventory visualizing selected GAM parameters, many additional data aggregates, time series with forecasts and recommendations to reduce, maintain or increase the reserve price. Details of the methods used at these stages are presented below, and the results of research in the next part of this study.

4.1. Data preparation and feature engineering

As mentioned above, the main source of data was Google AdManager (GAM), which offers many configurable reports (for all definitions, dimensions, and measures, see e.g., Ad Exchange historical report - Google Ad Manager Help, 2020). Sample dimensions are buyers (e.g., brands and their verticals), inventory segments (including pricing rules, sites, sizes, and channels), geography (countries), mobile devices details (device, operating systems) and time (days, weeks, and months, 2 years back). For each of these dimension, we can obtain many different measures, the most important being ad requests (all, matched, and CTR), ad request estimated Cost Per Mile (areCPM), matched eCPM, and Coverage.

Data may be obtained directly from GAM interface or via Application Programming Interface (API). In this analysis (univariate time series analysis and planning), we use a small subset of available data, literary one *Coverage* time series for 9 (of more than 600 managed by YR) AdUnits, a representative for low, medium and high coverage values.

Although data obtained from GAM is very clean, the most important and time-consuming part of this project was data preparation. That process was split into two phases: datasets preparation for time series forecasting and feature engineering for additional analysis, later on displayed in a dedicated BI dashboard.

4.2. Modelling

The main goal of the modelling phase was to identify the best possible time series forecasting algorithm for the Coverage parameter. In our analysis, we investigated a large spectrum of prediction algorithms - since it's impossible to present all of them in details; below we display only their most important aspects.

For an interesting study of these algorithms for time series forecasting, especially it's comparative analysis, see e.g. (Makridakis, Spiliotis and Assimakopoulos; 2018), and for implementation in Python, see e.g. (Brownlee, 2018b; and Brownlee, 2018a). In this project we apply only the so called *simple methods*, *autoregressive (ARIMA and SARIMA)*, *Exponential*

Smoothing and *Facebook Prophet* algorithms, limiting their application to univariate one-step forecasting. Below we shortly introduce this methods, with modelling evaluation results presented in the next section.

There are 2 different simple (naïve) forecasting strategies (for more details see e.g. (Brownlee, 2018a)): naïve, where past observations are used directly to forecast future values and average, where to forecast the future different statistics of past observations are used. A naïve forecast involves using the previous observations directly as the forecast without any change (so called a persistence forecast). In case we want to adjust this naïve forecast for seasonal effects, we can offset the historical data by a seasonal time lag. E.g., with seasonality = 7 days (which is our case), as a persistence forecast, we can take the observation with 7 timestep (days) back.

An average forecast strategy uses averaging prior value as predictions. All prior observations are collected and averaged, either using the mean or the median, with no other treatment to the data (Brownlee, 2018a). The number of prior observations averaged is controlled by the model parameter, which later on may be grid-searched in the model optimization procedure. Since *simple methods* are easy to understand and implement, they are perfect candidates for a benchmark model. In our case, we did a bit more by greed-search hyperparameter space of both naïve and average methods in order to find the best simple model - see Results section for more details.

Auto Regressive Integrated Moving Average (ARIMA) models proved to be very effective in modeling Coverage time series. In brief, *autoregression* means the model uses the dependent relationship between an observation and some number of lagged observations, *integrated* stands for the use of differencing of raw observations in order to make the time series stationary, and *moving average* means that the model uses the dependency between an observation and a residual error from a moving average model applied to the lagged observations.

Each of these aspects is reflected in ARIMA model parameters. With standard notation in the form ARIMA (p, d, q), where:

p - stands for the number of lag observations included in the model (also called *the lag order*),

d - is the number of times that the row observations are differenced (*degree of differencing*), and

q - is the size of the moving average window.

In ARIMA, a regression model is constructed (see e.g.: Brownlee, 2018b; Autoregressive integrated moving average, 2020) using the prepared data, specifically differenced to make a time series stationary. Differencing is a relatively simple transformation applied to a time series, where we eliminate trend and seasonal structures by computing the difference between consecutive observations:

$$y'_t = y_t - y_{t-1}$$

Sometimes this process may be repeated – the number of these iterations is called a *differencing order* (see d parameter).

SARIMA (Hipel and McLeod, 1994) extension supports the direct modeling of the seasonal component of the series. It uses differencing to remove additive seasonal effects. In that case, four additional seasonal parameters are introduced:

P - seasonal autoregressive order,

D - seasonal difference order,

Q - seasonal moving average order, and

m - the number of time steps for a single seasonal period.

These seasonal parameters may be identified by a careful study of autocorrelation plots (correlations of recent time steps).

Exponential smoothing is a time series forecasting method for univariate data that can be extended to support data with a systematic trend or seasonal component (Brownlee, 2018a). In these methods, the prediction is a weighted linear sum of recent past observations but, in opposite to Box-Jenkins ARIMA family of methods, the model explicitly uses an exponentially decreasing weight of past observations. In this research, we implement a *Triple Exponential Smoothing (ETS)* extension of ESM, also called *Holt-Winters ES*. For a detailed description and list of model hyper-parameters see e.g. (Exponential smoothing, 2020).

Facebook Prophet is an open-sourced project focusing on time-series forecasting, based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. Since the detailed description of methods available in this library is beyond the scope of this report, we encourage an interested reader to study papers and manuals presented in “Prophet”.

The general procedure for all the algorithms implemented was the following. After loading the dataset, we prepared the test harness: split the dataset in train and test dataset in proportion 9:1, and applied a walk-forward validation procedure (very efficient in case of time series forecasting model evaluation). Next, the model was defined, and appropriate (model-dependent) hyperparameter space configured and grid-search performed. Finally, models were evaluated (using RMSE as a quality measure), and the best performing model configuration was identified.

4.3. Model evaluation

After the basic time-series analysis, including autocorrelation and stationarity checks, in order to identify the best forecasting model, we implemented the score and evaluate models described above. With Root Mean Square Error (RMSE) as a forecast quality measure, our goal is to identify the best one. In case we get satisfactory forecasting (prediction error at the level of 10% of Coverage or lower), we will suggest this as a first, simple method of setting up a pricing rule.

As an evaluation metric for time series forecasting models, we propose RSME (Root Mean Squared Error) as easy to understand measure of the prediction quality:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{j=1}^n (y_j - \hat{y}_j)^2} \quad (1)$$

where y and \hat{y} are observed and predicted values of time series. Thus, RMSE being a square root of the average of squared differences between a prediction and actual observation measures the average magnitude of error.

Compared to an alternative measure - Mean Absolute Error (MAE):

$$\frac{1}{n} \sum_{j=1}^n |y_j - \hat{y}_j|$$

RMSE gives a relatively high weight to large errors (errors are squared before averaging). Since in our analysis large errors are undesirable, we choose RMSE as a metric to score forecasting models.

To evaluate the model a *walk-forward validation procedure* is applied (see e.g.: Brownlee, 2018b). Firstly, the dataset is split into train and test datasets. Next, starting at the beginning of the time series, the minimum number of samples is used to train a model, prediction for the next time step using a trained model is made, and the predictions are evaluated against the known value. Finally, the time window is expanded to include the known value, and the process is repeated.

5 Results

As first, the exploratory data analysis (EDA) was performed. The dataset, with Coverage time series

of app. 600 AdUnits for the period 01.2019-11.2019, was divided into three tertiles containing low, medium and high Coverage ones.

Next, we randomly picked three representatives for each category. Exploratory visualization for these nine representative AdUnits is presented in Fig. 2.

Additionally, we performed Coverage time series decomposition - an example for the same is displayed on Fig. 3

As can be seen, on average, it's hard to identify any trends or seasonality (which is obvious taking into account that AdUnits belong to very different website categories). Some obvious outliers are present in most of the sample AdUnits. To check time-series stationarity, we performed a series of AdFuller (ADF) tests. In most, they appeared to be stationary: for four (out of nine), ADF Statistics are larger than the critical value at 1%, thus these datasets seem NOT to be stationary. But at 5% level, most of the time series are stationary. Autocorrelation plots showed that, with some small exceptions, we don't observe a significant autocorrelation.

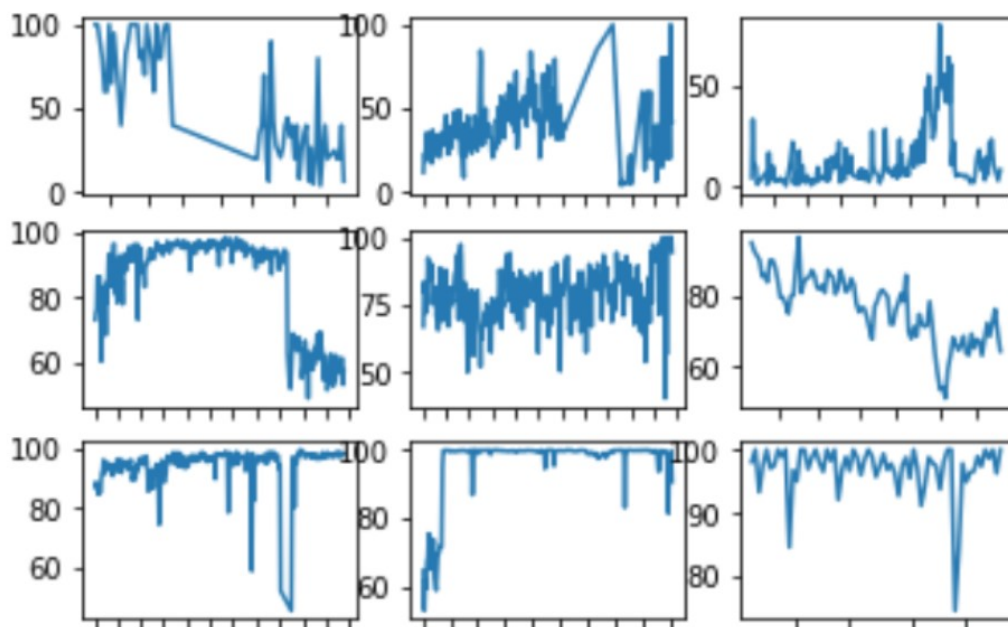


Figure 2. Line plot of Coverage time series for low, medium and high Coverage AdUnits (Source: Own computations)

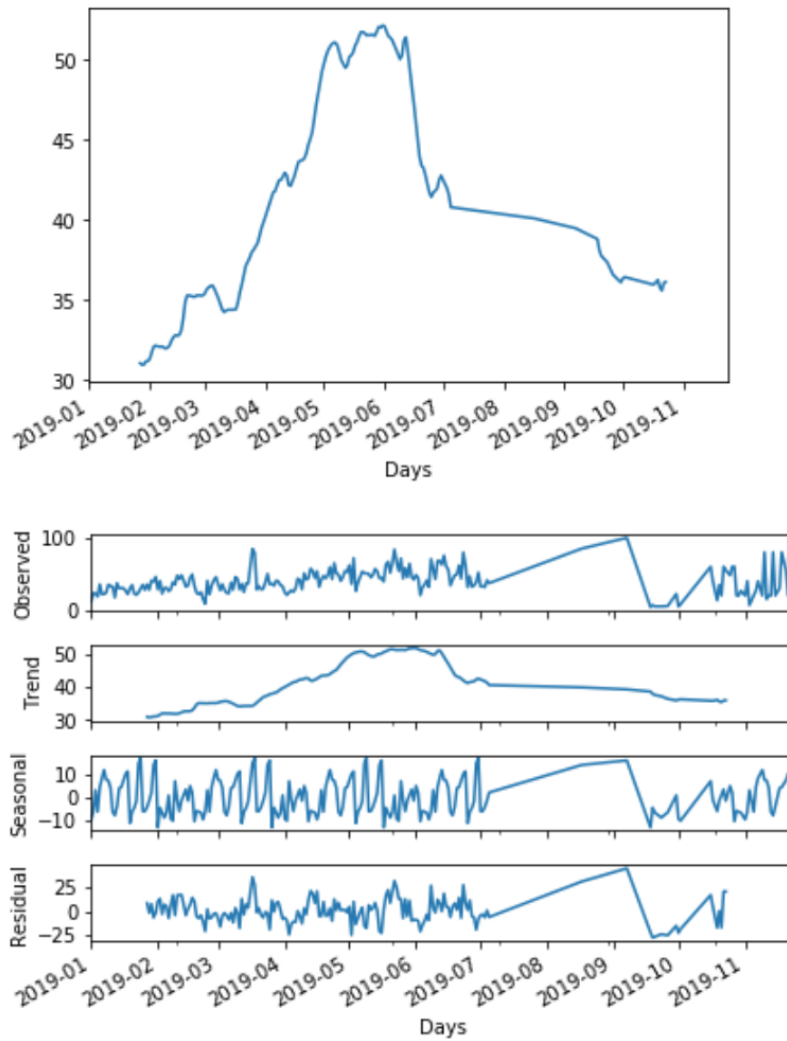


Figure 3. Coverage time series decomposition for a sample AdUnit
(Source: Own computations)

Having analyzed the data and prepared for modelling, according to our research methodology, we started with the benchmark model identification. For that we chose the persistence forecast model: a very simple heuristics, where the value from step “t” is used as a prediction for step $t+1$. The RMSE for this baseline model, averaged over 9 AdUnits = 11.20.

Next, we applied different time series modelling techniques to nine representative AdUnits. We started with autoregressive models. Following results of our previous research (not covered in this paper), we focused on Seasonally adjusted ARIMA. We searched over the following parameter values:

$$p = (0, 1, 2),$$

$$d = (0, 1),$$

$$q = (0, 1, 2),$$

$t = 'n'$ (no trend), ‘c’ (constant), ‘t’ (linear) and ‘ct’ (constant with linear trend),

$$P = (0, 1, 2),$$

$$D = (0, 1),$$

$$Q = (0, 1, 2), m = (0, 7).$$

The grid search over this parameter space showed that the best model has the following configuration: Model (2, 1, 2), (0, 0, 2, 7), 'n'], which reads:

1) for trend elements:

- a) $p = 2 <$ trend autoregression order,
- b) $d = 1 <$ trend difference order,
- c) $q = 2 <$ trend moving average order;

2) for seasonal elements:

- a) $P = 0 <$ seasonal autoregressive order,
- b) $D = 0 <$ seasonal difference order,

- c) $Q = 0 <$ seasonal moving average order,
- d) $m = 7 <$ the number of time steps for a single seasonal period,
- e) $t = 'n' <$ no trend parameter.

It's RMSE (averaged over 9 AdUnits, excluding one outlier) equaled to 8.48. In conclusion: SARIMA model significantly improves our benchmark, although computations were extremely time consuming. Next, we evaluated Exponential Smoothing (ETS) models. Since ETS computations are also extremely time consuming, we didn't perform a systematic grid search, but tried different manual configurations. Approximately 10 experiments over these models showed resulted in the best model with the following hyper-parameter set-up:

- T = None (trend is dropped),
- D = False (trend shouldn't be damped),
- s = Additive (seasonal component is additive),
- p = 7 (seasonal period = 7 time steps), and
- b = False (we do not perform power transform of the series).

It's (averaged over 9 AdUnits) RMSE = 8.37, which is better than our benchmark and SARIMA models. Other model setup's, with a bit worse (but comparable) RMSEs, were with $t = 'add'$ (additive trend), $s = 'mul'$ (multiplicative seasonality), and different values of seasonal period p (1, 2, 20).

Conclusion:

ETS model significantly improves our benchmark, and it's quality is comparable to the best SARIMA model.

Finally, we evaluated a Facebook Prophet library. Interestingly, its basic configuration resulted in RMSE (averaged over 9 AdUnits) = 10.6, what is worse than ETS and SARIMA, with low-Coverage AdUnits representatives being significant "outliers" (in terms of RMSE). Computation for medium and high coverage representatives, combined with *fbprophet* library native diagnostics (with cross-validation procedure, described in (Prophet Diagnostics, 2020) gave more promising results: it resulted with RMSE = 6.71.

The evaluation of all the time series forecasting models evaluations is presented in Table 1.

Table 1. Evaluation of time series forecasting models
(Source: Own elaboration)

| Model | RMSE |
|---|-------|
| Baseline (persistence) | 11.20 |
| SARIMA | 8.48 |
| ETS | 8.37 |
| Facebook Prophet (Medium and High Coverage) | 6.71 |

Since, in 1. YR focused mostly on high Coverage AdUnits, 2. it's heuristics for setting up the minimal bidding prices takes into account high (>90%) Coverage values and 3. Facebook Prophet model training was significantly faster than SARIMA and ETS, we have identified Facebook Prophet as the most promising set of models.

To support YR consultants in their daily practice, a dedicated, interactive BI dashboard has been implemented.

For a given AdUnits, it displays:

- 1) basic statistics;
- 2) time series forecasting for key parameters: Coverage, AreCPM, and Estimated Revenue; and
- 3) a set of advanced AdUnit features, useful in in-depth analysis.

An example screen of this dashboard is presented in Fig. 4.

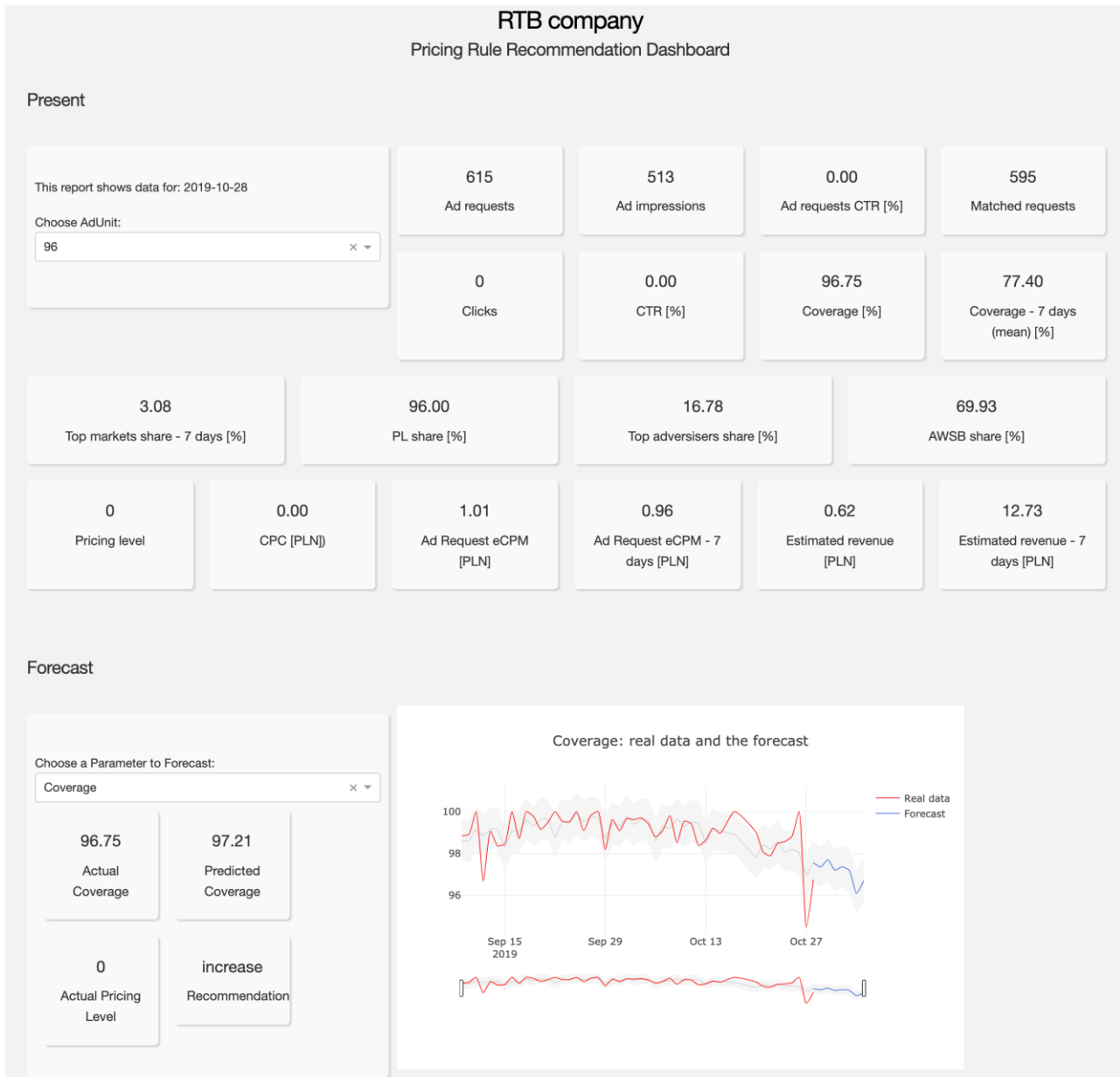


Figure 4. An interactive dashboard with recommendations based on Coverage forecasting
(Source: Own elaboration)

6 Conclusions and future works

The main goal of the research was to recommend a method for setting minimal prices for Publishers' inventories on RTB markets focusing on both profit and Coverage optimization. Based on the research presented in this paper, hundreds of experiments and YR consultants' experience, we derived the following heuristics:

1) Set up app. 10 so called pricing baskets for minimal prices, the first one being "no minimal price",

the 10th one being a maximum reasonable minimal price;

2) For a given AdUnit, predict the Coverage value for next three days using the best possible prediction algorithm;

3) If the predicted Coverage is:

a) higher than 95%: increase the price (move the reserve price to a "higher" basket, if possible),

b) between 90% and 94: hold the existing price,

c) lower than 90%: decrease the price (move the reserve price to a "lower" basket, if possible).

The quality of minimal price recommendations is driven by the quality of forecasting algorithms. Thus, the second main goal of this project was to identify the best possible forecasting model for *Coverage* parameter for 9 representative AdUnits of the Publisher active on the programmatic marketing market.

The dataset used for a procedure was 1-years' historical *Coverage* time series (app. 200 on average observations). Augmented Dickey-Fuller test proved the series is stationary, giving hope for future successful modeling.

As a measure of model accuracy, we choose Root Mean Squared Error (RMSE). Facebook Prophet algorithm has been found as the most promising one: not only did it scored the minimal RMSE error (for medium and high values *Coverage* representatives), but it was also the fastest (in terms of computing time) one.

The third goal of this research was to lay the foundations for future recommendation algorithms. The most natural approach would be to use Reinforcement Learning (RL). As shown in the Literature Review part of this paper, RL algorithms are being commonly applied in the bidding price optimization problem on RTB markets.

The main concern with such approaches lies in fact that most (if not all) approaches are based on historical data as a foundation of RL environment simulation, while to prove RL efficiency in practice, it should be applied to real-life situations. This is possible to implement, but since market feedback to price changes are counted in hours, there is a risk that it will take years (thousands of interactions needed) literally to train the final algorithm.

To sum up, the study proposed a new method of optimizing the prices of minimum advertising space offered by the publishers on the RTB markets, focusing not only on the profits from individual issues, but also on ensuring maximum filling (*Coverage*) of advertising space.

The latest machine learning algorithms were used for this purpose, identifying the *Facebook Prophet* library not only as one of the best (in the sense of RMSE), but also as the most computationally efficient. As the next step in the research, it was recommended to use teaching methods with enhancement

with *Coverage* as a vital component of the reward function.

7 References

- [1] Anon, 2020a. *Ad Exchange historical report - Google Ad Manager Help*. [online] Available at: <<https://support.google.com/admanager/table/7612037?hl=en>> [Accessed 10 Jun. 2020].
- [2] Anon, 2020b. Autoregressive integrated moving average. In: *Wikipedia*. [online] Available at: <https://en.wikipedia.org/w/index.php?title=Autoregressive_integrated_moving_average&oldid=950625193> [Accessed 1 Jun. 2020].
- [3] Anon, 2020c. Exponential smoothing. In: *Wikipedia*. [online] Available at: <https://en.wikipedia.org/w/index.php?title=Exponential_smoothing&oldid=957231778> [Accessed 1 Jun. 2020].
- [4] Anon, 2020d. *Prophet*. [online] Prophet. Available at: <<http://facebook.github.io/prophet/>> [Accessed 1 Jun. 2020].
- [5] Anon, 2020e. *Prophet Diagnostics*. [online] Prophet. Available at: <<http://facebook.github.io/prophet/docs/diagnostics.html>> [Accessed 1 Jun. 2020].
- [6] Austin, A., Barnard, J. and Hutcheon, N., 2019. *Programmatic Marketing Forecasts 2019*. [online] Zenith. Available at: <<https://s3.amazonaws.com/media.mediapost.com/uploads/ProgrammaticMarketingForecasts2019.pdf>>.
- [7] Austin, D., Seljan, S., Monello, J. and Tzeng, S., 2016. Reserve Price Optimization at Scale. In: *2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). pp.528–536.
- [8] Brownlee, J., 2018a. *Deep Learning for Time Series Forecasting*. [online] Available at: <www.machinelearningmastery.com>.
- [9] Brownlee, J., 2018b. *Introduction to Time Series Forecasting in Python*. [online] Available at: <www.machinelearningmastery.com>. Cai, H., Ren, K., Zhang, W., Malialis, K., Wang, J., Yu, Y. and Guo, D., 2017. Real-Time Bidding by Reinforcement Learning in Display Advertising. *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining -*

- WSDM '17*, pp.661-670.
- [10] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R., 2004. *CRISP-DM 1.0*. SPSS.
- [11] Choi, H., Mela, C.F., Balseiro, S. and Leary, A., 2019. Online Display Advertising Markets: A Literature Review and Future Directions. *SSRN Electronic Journal*. [online] Available at: <<https://www.ssrn.com/abstract=3070706>> [Accessed 4 May 2020].
- [12] Hipel, K.W. and McLeod, A.I. eds., 1994. Chapter 12 Seasonal Autoregressive Integrated Moving Average Models. In: *Developments in Water Science, Time Series Modelling of Water Resources and Environmental Systems*. [online] Elsevier.pp.419–462. Available at: <<http://www.sciencedirect.com/science/article/pii/S0167564808706737>> [Accessed 10 Jun. 2020].
- [13] Li, J., Ni, X. and Yuan, Y., 2018. The Reserve Price of Ad Impressions in Multi-Channel Real-Time Bidding Markets. *IEEE Transactions on Computational Social Systems*, 5(2), pp.583–592.
- [14] Li, J., Ni, X., Yuan, Y., Qin, R., Wang, X. and Wang, F.-Y., 2017. The impact of reserve price on publisher revenue in real-time bidding advertising markets. In: *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [online] 2017 IEEE International Conference on Systems, Man and Cybernetics (SMC). Banff, AB: IEEE.pp.1256–1261. Available at: <<http://ieeexplore.ieee.org/document/8122785/>> [Accessed 7 May 2020].
- [15] Lu, J., Yang, C., Gao, X., Wang, L., Li, C. and Chen, G., 2019. Reinforcement Learning with Sequential Information Clustering in Real-Time Bidding. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. [online] CIKM '19: The 28th ACM International Conference on Information and Knowledge Management. Beijing China: ACM.pp.1633–1641. Available at: <<https://dl.acm.org/doi/10.1145/3357384.3358027>> [Accessed 4 May 2020].
- [16] Makridakis, S., Spiliotis, E. and Assimakopoulos, V., 2018. Statistical and Machine Learning forecasting methods: Concerns and ways forward. *PLOS ONE*, 13(3), p.e0194889.
- [17] Myerson, R.B., 1981. Optimal Auction Design. *Mathematics of Operations Research*, 6(1), pp.58-73.
- [18] Radovanovic, A. and Heavlin, W.D., 2012. Risk-aware revenue maximization in display advertising. In: *Proceedings of the 21st international conference on World Wide Web - WWW '12*. [online] the 21st international conference. Lyon, France: ACM Press.pp.91-100. Available at: <<http://dl.acm.org/citation.cfm?doid=2187836.2187850>> [Accessed 5 Jun. 2020].
- [19] Ren, K., Qin, J., Zheng, L., Yang, Z., Zhang, W. and Yu, Y., 2019. Deep Landscape Forecasting for Real-time Bidding Advertising. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '19*. [online] the 25th ACM SIGKDD International Conference. Anchorage, AK, USA: ACM Press.pp.363–372. Available at: <<http://dl.acm.org/citation.cfm?doid=3292500.3330870>> [Accessed 4 May 2020].
- [20] Riley, J.G. and Samuelson, W.F., 1981. Optimal Auctions. *The American Economic Review*, 71(3), pp.381–392.
- [21] Wang, J., Zhang, W. and Yuan, S., 2017. Display Advertising with Real-Time Bidding (RTB) and Behavioural Targeting. *arXiv:1610.03013 [cs]*. [online] Available at: <<http://arxiv.org/abs/1610.03013>> [Accessed 11 May 2020].
- [22] Wang, Y., Ren, K., Zhang, W., Wang, J. and Yu, Y., 2016. Functional Bid Landscape Forecasting for Display Advertising. In: *European Conference on Machine Learning and Knowledge Discovery in Databases - Volume 9851*, ECML PKDD 2016. [online] Riva del Garda, Italy: Springer-Verlag.pp.115–131. Available at: <https://doi.org/10.1007/978-3-319-46128-1_8> [Accessed 3 Jun. 2020].
- [23] Wu, W., Yeh, M.-Y. and Chen, M.-S., 2018. Deep Censored Learning of the Winning Price in the Real Time Bidding. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18. [online] London, United Kingdom: Association for Computing Machinery.pp.2526–2535. Available at: <<https://doi.org/10.1145/3219819.3220066>> [Accessed 3 Jun. 2020].

- [24] Wu, W.C.-H., Yeh, M.-Y. and Chen, M.-S., 2015. Predicting Winning Price in Real Time Bidding with Censored Data. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15*. [online] Sydney, NSW, Australia: Association for Computing Machinery. pp.1305–1314. Available at: <<https://doi.org/10.1145/2783258.2783276>> [Accessed 3 Jun. 2020].
- [25] Yang, C., Lu, J., Gao, X., Liu, H., Chen, Q., Liu, G. and Chen, G., 2020. MoTiAC: Multi-Objective Actor-Critics for Real-Time Bidding. *arXiv:2002.07408 [cs]*. [online] Available at: <<http://arxiv.org/abs/2002.07408>> [Accessed 5 May 2020].