International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

User-Driven Online Advertisement Auctions: A Game Theoretic Analysis

Simulated Stackelberg Game Trials with User Preferences — Neural Networks Based Vickrey Auction for Revenue Maximization

Jwalin Thaker

(Student/Author)
ECE Department Stevens Institute of Technology
Hoboken, NJ, USA
jthaker1@stevens.edu

Abstract

With a paradigm shift happening to the Internet, its architecture is transitioning from Web2.0 to Web3.0. Major highlights of this new system are key emphasis on user privacy, data security and decentralization. Centered around user privacy, this paper is a summary of research performed around online auctioning mechanisms, especially digital advertisements (ads), and how they would adapt to this change. Because the power to release data will shift towards the user, is it imperative to understand what changes an Ad agency or service provider would have to implement, to tailor personalized ads and keep website traffic afloat. Conventional advertisement auctioning is structured around the search engine provider (auctioneer) auctioning ad spots, which are bid in real-time by goods/service providers (bidders). The work here extends that model by introducing a third entity, the "user", its preferences, and how they drive the auction, by influencing bidder preferences. The goal is to model the problem under "Game theory", as a Stackelberg leadership game, and perform simulated trials of Vickrey auctions to record the trend on ad spot allocation and auction payments with changing user preferences.

Keywords: Web3.0, Auction, Advertisements, Game Theory, Neural Networks, Revenue Maximization, Constraints, Preferences, Vickrey Auction, Stackelberg Game, Security, Privacy, Decentralization

I. Introduction

Internet is arguably the best synthetic invention of the 21st century, one could say. It allows one to share resources and information across the globe within seconds. Most of the world's trade (of any goods and services) has switched to an online platform (marketplace and/or private website) housing under the e-commerce umbrella. In early 2000s, companies saw this behavioral paradigm as a unique opportunity to invest in advertisements over the Internet called digital marketing. Instead of bidding for physical space over placards, banners and billboards, now the competition for customer acquisition happens on your favorite search engines and/or web pages.

Digital advertisement is currently the top form of generating revenue online according to data collected by Statista resources. According to [1], Web 3.0 environments enable richer media formats. It happens by collecting user data (in the form of website footprints, trackers, cookie preferences and browsing history) and then drawing general consensus for behavioral segmentation and/or personalized, targeted advertisements.

Even though organizations have found comfort around running advertisements and generating revenue from user traction, the Internet architecture that supports it is changing. To understand that, here is an overview of different versions of the Internet so far -

1. *Web1.0* - This architecture supported only static website with no interaction or animation with the content — centralized and published by handful of organizations — no ads [Active: 90s]
2. *Web2.0* - Support for dynamic websites with interactive content — centralized, but content publication open to all — user data captured for digital ads [Active: 2000s - present]
3. *Web3.0* - Decentralized dynamic websites — focused on user privacy, security and data integrity, user-controlled content publishing and subscription — personalized ads [Active: 2020s- present] ("Fig. 1")

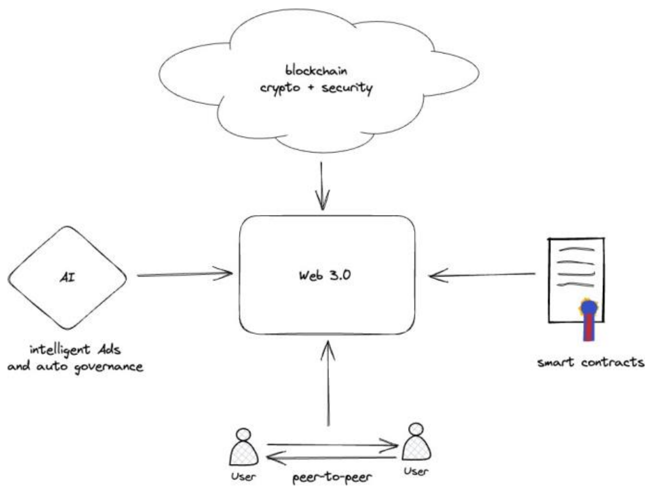

Fig. 1. Web3.0 unique aspects

With the current model of Internet (Web2.0) shifting to Web3.0, large efforts are being taken to adapt a decentralized approach towards fueling it (peer-to-peer), meaning each user's personal device would act as an individual node of publishing and consuming content, having most of the control over the information visible and released. With varied access of user information to online advertisers, it is imperative to see how user preferences might impact advertisements in Web3.0. Assuming that change of control will only lead to better personalized advertisements, we focus on the negatives that might surface from user preferences constraining the generic search engine ad spots positions. To understand this, we first define what an auction is, state several prevalent auctioning techniques, how most of the online auctioning of ad spots happens and summarize existing research around several proposed optimal auctioning mechanisms.

The formal definition of an **Auction** is "An event held by the auctioneer (willing to sell or mediator) for the sale of item(s) amongst the bidders (willing to buy)". Some of the primitive and modern auctioning techniques still widely used are -

1. *English Auction*: In such auctions, the bidders openly announce their bids, and the auctioneer increases the price until no higher bid is made. The winner of the auction is the bidder with the highest valuation

and is obliged to pay the bid amount to the auctioneer.

2. *Dutch Auction*: Opposite to the English auctions, the auctioneer starts at a very high price and gradually lowers the price. The winner of the auction is the first bidder which accept the current price and pays it out to the auctioneer.
3. *First-Price Sealed-Bid Auction*: Unlike open bid auctions, the bidders submit their bids secretly, and the highest bidder wins the auction, paying the price they bid. The auctioneer is trusted and expected to be impartial throughout the process.
4. *Second-Price Sealed-Bid Auction (Vickrey Auction)*: Similar to the first-price sealed-bid auction, bidders submit their bids secretly. However, the highest bidder wins the auction but pays the second-highest bid price. These auctions are famous because they provide incentive to bidders to stay true to their valuations.

For online advertisement real-time bidding scenarios, "Vickrey Auction/Second-Price Sealed-Bid Auction" is the most widely used technique as it ensures robust, secure and impartial auctioning throughout the process. A typical flow for such an interaction looks like this -

- A user goes to a search engine provider (auctioneer) to get results for a search prompt
- Auction starts in the background for ad spots on ad exchanges (servers) hosted by the auctioneer
- Bidders relevant to the search prompt keywords compete in real-time for the advertisement spots by providing valuations of each advertisement spot
- Auctioneer applies its proprietary logic (either to optimize allocation of ad spots or to maximize revenue) to calculate scores based on several bidder (including their valuation) and self preferences and constraints. With these scores, the bidders are ranked and assigned the ad spot accordingly This aligns with [2] who demonstrated that Vickrey auctions are the most efficient and robust mechanism for revenue maximization in the presence of budget constraints. The budget constraints analysis builds on [3] who discussed multi-unit auctions with budget-constrained bidders.

Since the second price auctions have been introduced, several researchers have aligned interests to analyze them and design methodologies to optimize them. There have been research papers mentioning modeling auctions under game theory, as a game between the bidder and auctioneer to find the Nash equilibrium, resulting in optimal strategy for both at each item bidding. Some other modifications and recent research areas involve introducing concepts of ML/AI in the form of neural networks to optimize the allocation and payment efficiency out of the auction. With such accomplished efforts, we plan to extrapolate these findings and incorporate game theory via a Stackelberg leadership model starting from the user and its preferences signalling and governing bidder preferences, and then implementing Vickrey auctions using neural networks to promote revenue maximization of the auctioneer. With this exercise, we want to simulate auction trials and gain insights into impact of user induced stringent data practices (which might come with Web3.0) in the form of preferences on the overall auction, with probable suggestions for online advertisement companies to accommodate any changes to come.

The remainder of the paper is organized as follows:

- *Section II* talks about formally defining the problem statement and modeling it mathematically to set up foundation for algorithmic implementation
- *Section III* titled "Stackelberg-Vickrey auction trials with neural network" elaborates about the implementation of the proposed solution, wherein Stackelberg-Vickrey trials with entity level preferences and constraints are plugged with simple payment and allocation neural networks to fuel

numerical analysis and result amalgamation

- *Section IV* highlights empirical results from the trials and several metrics of comparison and solution evaluation
- *Section V* concludes the paper with any potential insights for advertisement agencies or bidders to counteract any Web3.0 user preferences induced traction loss

II. Problem Statement

To mold our digital advertisement scenario into auctioning jargon, we associate certain terms and introduce entity level preferences (“Fig. 2”) -

1. Users - Entities that initiate the auction by search a prompt relevant to some goods/service they need.
2. Bidders - Entities that participate in the auction bidding by placing their ad spot valuations — Marketing agencies and/or organizations providing goods/services — they compete to lure customer to their websites/products.
3. Auctioneers - Entities that host/organize the auction and govern them — Search engine providers or private advertisement tool/exchange server owners.
4. Items - Advertisement spots — bidders contend for them and users click the content they showcase.

A. *User Preferences*

For this new entity being introduced in the auction, we define three preferences as features that govern the amount of privacy that the user wants to retain while initiating and receiving results from the auction -

1. *Trust score* - Emphasizes the trust on a particular bidder (goods/service provider) on advertisement (average score sentiment across all user-base observations) — higher the better
2. *Relevance score* - Gives an associated score to the search prompt result generation (average score across historical user observations and assumed to be less fluctuating) — higher the better
3. *Privacy score* - Denotes the resistance to data release by the user to bidders' services (individual user belief) — lower the better

This preference learning mechanism follows [4] who estab- lished frameworks for learning user preferences in mechanism design.

B. *Bidder Preferences*

Some of the features tied to the bidder entity encompass its preferences and constraints such as -

1. *Click-through rate* - The measure of probability per item (ad spot) that it is likely to be clicked by the user, directly formulated using user preferences (see in Section III) — higher the better
2. *Max budget* - A bidder's maximum budget per item during auction participation — higher the better
3. *Auction participation cost* - Denotes the fixed participation cost incurred for partaking in an auction iteration — lower the better

C. *Auctioneer Preferences*

Some of the preferences tied to the auctioneer entity are -

1. *Minimum bid* - A constrain/preference levied by the entity to ensure that all bids are positive and that

even the worst case of auction iteration results in minimal loss — higher the better

2. *Auction hosting cost* - Denotes the fixed hosting cost incurred from the auction in each iteration — lower the better

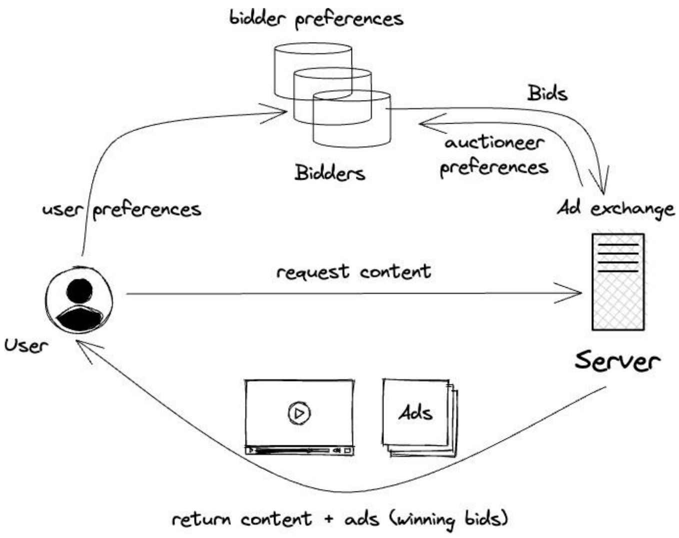

Fig. 2. Online advertisement auction with user preferences

For the scope of this paper and the game that we would set out to model ahead, we focus on the Vickerey auctions, the reason being its ease of modeling and understanding and that its innate structure gives no incentive to any bidder to inflate or understate their item valuations and stay true to them. Before we mathematically formulate the problem of user preferences driven ad spot auctions, here are some of the rational assumptions around the problem -

- For each advertisement spot, there is a different valuation, based on the click-through-rate
- Each user is putting a similar prompt in the search engine to have commonality in bidder preference selection
- The bidder aims for securing the maximizing return of investment (ROI) and always places a positive bid
- The auctioneer aims at maximizing the revenue from the auction
- The preferences set by the user, bidder and auctioneer are purely based on self-interest and are just
- All mathematical values mentioned in the paper are in the range of 0-1

III. Stackelberg-Vickrey auction trials with neural network

Against the problem stated so far, the solution we put forth involves two popular game theory models - “Stackelberg leadership game”, which defines a game between 2 players, where player-1 leads the game either by signaling or playing a strategy, and the player-2 responds back sequentially, and “Vickrey Auction”, which defines an auctioning game between bidder and auctioneer, where sealed bids are placed by bidders, staying true to their item valuation and the highest bidder wins the auction with the payout of second highest bid. We plan to borrow ideas from both these models and stack them in such a manner -

1. *Stackelberg leadership game* - User initiates the auction, sets preferences based on historical or current beliefs and signal them to the bidder, in-turn affecting some of their own initial constraints/preferences.
2. *Vickrey auction game* - With the updated bidder preferences, an auctioning game is modeled between auctioneer and the bidder, using simple neural networks for payment and allocation, the goal being to allocate probabilities of single or multiple ad spots (items) across the participating bidders.

Now, we define the problem mathematically, stating constraints/preferences, how they are sampled for the simulation and utility of each entity -

A. ***User***

For entity type user, we define the following -

- s_t - Trust score

As this preference denotes the average trust score towards a bidder across all user-base with the similar search prompt, we believe that most users will start with 50-50 trust in the bidder and then with iterations of simulation, deviate towards a better or worse trust for a bidder. Thus, we sample this from a standard normal distribution with mean at 0.5 and standard deviation of 1 (orange line in Fig. 3)

- s_r - Relevance score

This preference measure the relevance of returned results based on a search prompt, directly tied to affinity to advertisement visible to the user. As our assumption states users' nature to perform same search queries, and the auctioneer being consistent with the auction, we sample these preference from a uniform distribution with clipped deviation of 0.15 (blue line in Fig. 3)

- s_p - Privacy score

One of the most important user preferences that dictates personal resistance to amount of data being released out in the Internet. To model it in our simulations, we sample it from a beta distribution with parameters (3, 2) so that it provides left skewed distribution, following the trend of how users start with a higher privacy threshold but with time, reduces it (green line in Fig. 3)

- Utility -

The utility of a user is maximizing trust and relevance score, whilst minimizing the privacy score (threshold). It can be denoted as:

$$u_u \propto \max(s_t), \max(s_r), \frac{1}{\min(s_p)} \qquad (1)$$

$$where, s_t, s_r, s_p > 0 \qquad (2)$$

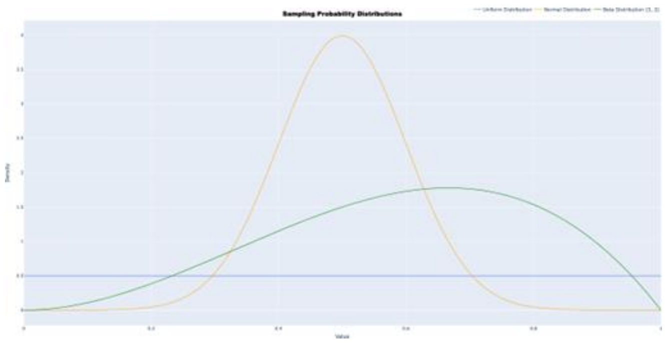

Fig. 3. Sampling distribution for preferences

B. *Bidder*

For entity type bidder, we define the following -

- γ - Gamma

Bidders own belief and confidence on ad quality, its relevance etc. It is a random number sampled from a uniform distribution to have similar bidders possess similar γ values.

- ctr - Click-through rate

We model the click through rate based on the gamma value (to introduce variance) and the user preferences. The formula is created to encapsulate the relative change that we assume each user preference to hold against the actual click-through-rate of the bidder.

$$ctr = (s_t * s_r * \gamma)/\sqrt{s_p} \qquad (3)$$

- m_b - Max budget

Bidder's preference that states a budget cap for each item, basically the maximum willingness to pay for an item (Ad spot). For the case of our simulations, as most of the item allocation and payment are expected to be between 0-1, we set this value to be fixed at 1.

$$m_b = 1.0 \qquad (4)$$

- c_b - Auction participation cost
  For the sake of this exercise, we keep this fixed at 0.05.

$$c_b = 0.05 \qquad (5)$$

- ω_b - Valuation per item
  A bidder's valuation of an item i is defined as -

$$\omega_b = ctr_i * m_{bi} \qquad (6)$$

- Utility -
  The bidder's utility is the valuation it states for an item minus the fixed cost of participating in the auction, denoted as

$$u_b = \omega_b - c_b - p_b \qquad (7)$$

where, p_b is payout for winning the auction and greater than 0, else 0.

C. *Auctioneer*

For entity type auctioneer, we define the following -

- c_a - Auction hosting cost
  For the sake of this exercise, we keep this fixed at 0.05.

$$c_a = 0.05 \tag{8}$$

- m_a - Minimum bid

The auctioneer sets a constraint in the auction to have a reserved minimum bid so that they get minimal losses from hosting it. This is again drawn from a uniform distribution to keep it unbiased across simulation runs.

- Utility -

The auctioneer's utility is the revenue generated from the auctions i.e. payout from the winning bidder (assuming the second price auctions) minus the fixed cost of organizing it -

$$u_a = p_b - c_a \tag{9}$$

D. ***Algorithm***

Before actually building and plugging-in a neural-net architecture of our own, we decided to learn and adopt an existing deep-learning based optimal auctioning technique and perform a proof of concept off of their work by layering user preferences in it. With promising results showing some sort of insight into the trend that user preferences, especially the privacy threshold showed with allocation and payment, we decided to come up with a simple neural network named "UserNet". Due to computational limitations we saw while initially piloting against the previously mentioned author's approach, we structured the network without any deep learning framework or complex layering. Below we have the pseudo code showing the payment (1) and allocation (2) algorithm based on a simple neural

Algorithm 1 Payment Network

Require: Payment network parameters

Initialize the payment network with the following parameters:

- Input size: Length of bidder preferences
- Output size: Length of auctioneer preferences
- Learning rate: A scalar that controls the step size of weight updates during backpropagation
- Regularization rate: A scalar that controls the strength of L2 regularization

for each bidder in bidders **do**

Compute bidder's preferences by forwarding user prefer- ences through the allocation network

Calculate the predicted auctioneer preferences by for- warding bidder preferences through the payment network

Compute the gradient output as 2 * (predicted auctioneer preferences - actual auctioneer preferences)

Update the payment network weights using backward pass with bidder preferences and gradient output

end for

Algorithm 2 Allocation Network

Require: Allocation network parameters

Initialize the allocation network with the following parameters:

- Input size: Length of user preferences
- Output size: Length of bidder preferences
- Learning rate: A scalar that controls the step size of weight updates during backpropagation
- Regularization rate: A scalar that controls the strength of L2 regularization **for** each bidder in bidders **do**

Compute bidder's preferences by forwarding user preferences through the allocation network

Calculate the predicted auctioneer preferences by for- warding bidder preferences through the payment network

Compute the gradient output as 2 * (predicted auctioneer preferences - actual auctioneer preferences)

Update the allocation network weights using backward pass with user preferences and gradient bidder preferences

end for

network, with forward and backward functionalities mimicking gradient descent optimization technique inherently - To summarize the above mentioned pseudo codes, for allocation network, the initial input and output weights are based on the user and bidder preferences sizes respectively, while for payment network, the weights are bidder and auctioneer preference sizes respectively. With forward and backward functions we calculate the regret and gradient error in the system at any point with the formula -

$$r = p_{pred} - p_{actual} \quad (10)$$

$$e_{grad} = 2 * r \quad (11)$$

where r is the regret and e_{grad} is the gradient error. Minimizing the gradient error is the goal of the model while training, in order to adhere to the auctioneer's strategy of "revenue maximization". For preferences that we have already defined for each entity, an example ANNVizualizer output looks like this for both the networks Plugging these networks under a Stackelberg game with each iteration, the entire pseudo code looks as mentioned in algorithm 3-

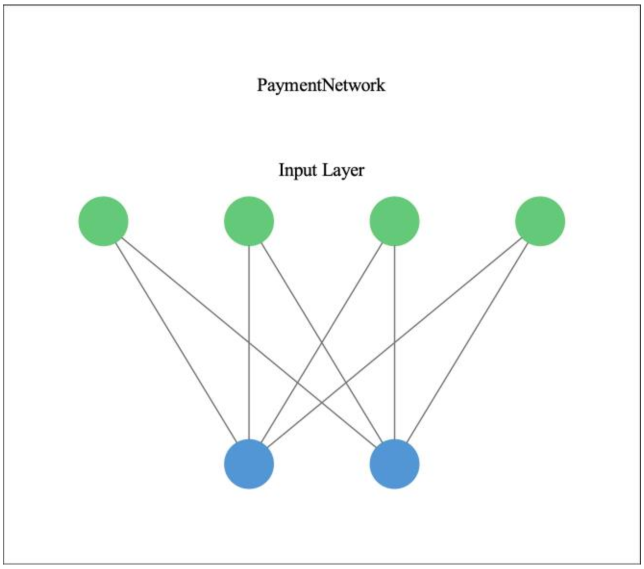

Fig. 4. Payment Network

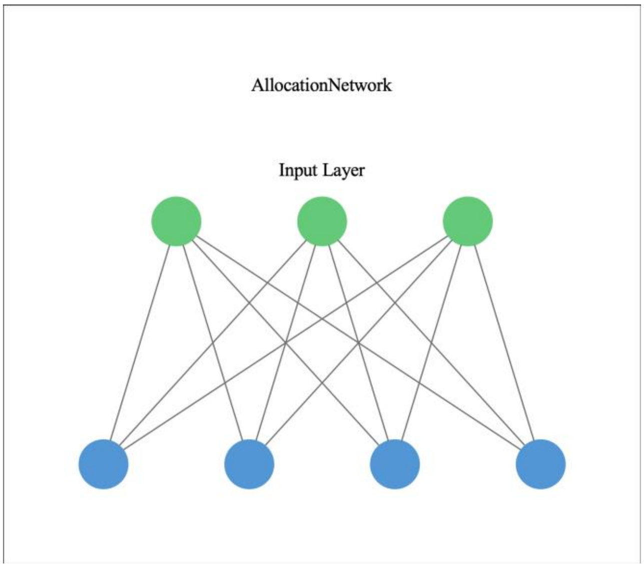

Fig. 5. Allocation Network

Plugging these networks under a Stackelberg game with each iteration, the entire pseudo code looks as mentioned in algorithm 3-

Our neural network approach extends the work of [5] on optimal economic design through deep learning and [6] who demonstrated differentiable auction mechanisms using neural networks. This combination allows for revenue maximization in budget-constrained environments, similar to the approach discussed by [7] who studied revenue maximization when bidders have hard budget constraints.

Algorithm 3 Run Scenario

```
procedure RUNSCENARIO(item_id, identifier)
    Parse n_users, n_auctioneers, n_bidders from identifier
Initialize empty lists: users, bidders, auctioneers
for i in range(n_users) do
Create user with id i
Set user preferences
Append user to users list
end for
for i in range(n_bidders) do
Create bidder with id i
Set bidder preferences using users list
Append bidder to bidders list
end for
for i in range(n_auctioneers) do
Create auctioneer with id i
Set auctioneer preferences
Append auctioneer to auctioneers list
end for
Initialize the model with users, bidders, auctioneers
Train the model
Predict using the model: allocation, payment, item_probability
return allocation, payment, item_probability, users, bidders, auctioneers
end procedure
```

IV. NUMERICAL RESULTS AND ANALYSIS

In this section we discuss about the simulation trials that were conducted based on the algorithms prepared as part of the solution. It was imperative to write code in a manner that scaled to cover multiple scenarios across simulations, for e.g. <u>1 user, 1 auctioneer, 2 item, 2 bidder</u> or else, <u>1 user, 1 auctioneer, 1 item, 2 bidder</u>. Hence, we created an identifier string called "SCENARIO" structured as "num-users U × num-bidders B × num-auctioneers A" (12) for e.g., "1Ux2Bx1A" which means running the code for <u>1 user, 2 bidders and 1 auctioneer</u>. To encompass a wide vari- ety of simulation scenarios, we perform multiple iterations for various scenarios and then accumulate our results. The trials are structured in this format -

- Trial 1 −→ 50 iterations | 1 item - 1Ux2Bx1A | 2 items
- 1Ux2Bx1A | 2 items - 1Ux3Bx1A
- Trial 2 −→ 100 iterations | 1 item - 1Ux2Bx1A | 2 items
- 1Ux2Bx1A | 2 items - 1Ux3Bx1A
- Trial 3 −→ 500 iterations | 1 item - 1Ux2Bx1A | 2 items
- 1Ux2Bx1A | 2 items - 1Ux3Bx1A

For each of these trials, the model training accuracy % has been averaged and represented in tables (I II

III) shown below (*please note that all the values have been rounded to the nearest whole number*) -

From the results in these table I, we can see that -

- With increasing iterations, the model does a good job at reducing gradient error, learning across iterations

TABLE I: MODEL TRAINING ACCURACY - TRIAL 1

Network Output	50 Iteration	100 Iterations	500 Iterations
1. Payment	37%	41%	44%
2. Allocation	48%	54%	57%

- The payment network seems to do a poor job because the training happens only on 1 item, 1 user and 2 bidders, not allowing much learning due to limited internal train iterations
- The allocation network also gives inaccurate results as the aggregation of allocation probabilities for a scenario with less items (ad spots) than bidders yields a lot of 100% error cases

TABLE II: MODEL TRAINING ACCURACY - TRIAL 2

Network Output	50 Iteration	100 Iterations	500 Iterations
1. Payment	40%	45%	52%
2. Allocation	88%	91%	92%

From the results in these table II, we can see that -

Increasing iterations results in better fit for both networks with reduced gradient error

The payment network seems to do a better job than the previous trial because the training happens with more internal iterations and nodes on 2 items, 1 user and 2 bidders, allowing more learning

The allocation network gives great results as there are sufficient items for each bidder so the only error coming is from the true gradient error

TABLE III: MODEL TRAINING ACCURACY - TRIAL 3

Network Output	50 Iteration	100 Iterations	500 Iterations
1. Payment	39%	41%	47%
2. Allocation	50%	54%	66%

From the results in these table III, we can see that -

- With increasing iterations, we see a predictive lift in both allocation and payment networks, following previous observations
- With increasing iterations, we see a predictive lift in both allocation and payment networks, following

previous observations

- This scenario performs closely similar to the trial 1 but slightly better than it because the scenario has more internal iterations and neural points to train on with 2 items, 1 user and 3 bidders in case of the payment

With these results and its inferences, we can confirm that in order to get a well trained model fit, we have two directions -

1. Tweak any well-balanced trial (equal bidders and items), by updating the scenario to accommodate several items and bidders (>100) keeping 1 user & 1 auctioneer as this would lead to increased nodes in the network and more internal iterations to train upon till the model converges to a stable equilibrium.
2. Scale the trial iterations, by running for a high value, providing more time to accumulate user preference history and bidder belief with training across iterations.

Optimizing in both these possible directions turned out to be computationally expensive, providing very gradual predictive lift. With limitations to the hardware, we decided to perform more statistical tests to verify certain hypotheses by capturing auction level information like allocation and payment values, item distribution probabilities against the generate user and bidder preferences, resulting in these visualizations (*please note that the outputs are plotted on different trials iterations, but same scenarios [1 item - 1Ux2Bx1A]*) -

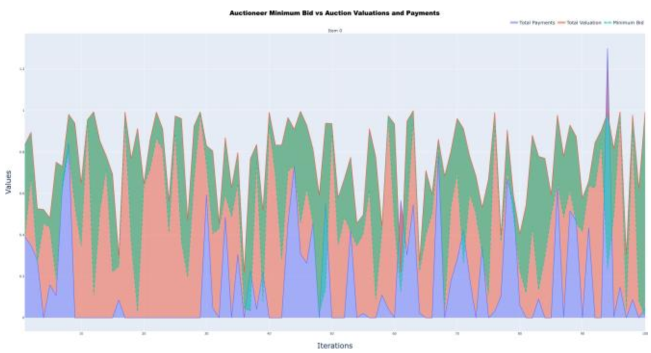

Fig. 6. Auctioneer Min Bid vs Payment vs Valuations

Hypothesis: Random sampling of entity preferences from some set of distributions breaks the auction integrity

Inference: Fig. 6 is the representation of the auction integrity which tracks the constraint violation for minimum bid set by the auctioneer during each iteration (100 runs). On the X-axis, we see the simulation iterations and on the Y-axis, we see minimum bid, bidder valuation and scaled payment values. The stacked area graph portrays that when the auction plays out, the modeled constraints, like the auctioneer preference of minimum bid (green dotted line) is retained as the bidder valuation (red line) of the item is almost always higher than it. Also the scaled payments with a factor of 0.75 ensure that the most of the payments fall in between the minimum bid and the item valuation by the bidder, meaning that the Vickrey auctions are intact with the introduction of randomness through Stackelberg trials. Hence, we can claim that the hypothesis fails and that the auction integrity stands.

Hypothesis: Lower the user privacy threshold, lower the auction payout for participating bidders

Inference: Fig. 7 plots the user privacy threshold, one of the user preferences, on X-axis and the bidder payouts, post winning the auction, on Y-axis (500 runs). The visualization shows that the auction payouts throughout the iterations are bounded within the confidence band made from the scaled beta distribution of the original variable user preference - privacy threshold. This is really insightful as we can see that with decreasing privacy threshold, the winning amount from the second-price auction (Vickrey) also decreases. Assuming that the trust score and relevance score (other two user preferences) are constant and plateau after certain iterations, we can state that once the user releases more data on the Internet, the bidder acceptance rate increases, hence showing a better click-through-rate. We also performed a t-test to confirm this through division of the data and cross validation to calculate differences in group means resulting in no significant difference. Hence the hypothesis is valid and stands.

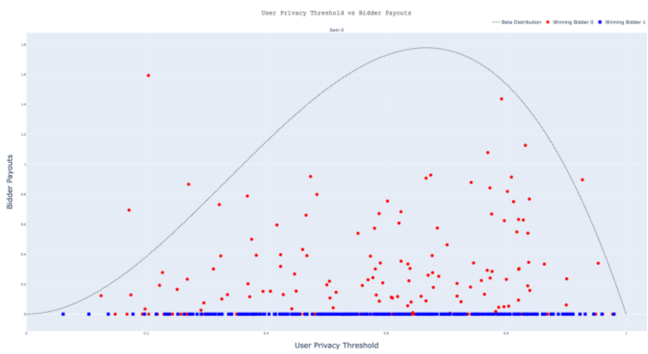

Fig. 7. User Privacy threshold vs Auction Payouts

bounded within the confidence band made from the scaled beta distribution of the original variable user preference - privacy threshold. This is really insightful as we can see that with decreasing privacy threshold, the winning amount from the second-price auction (Vickrey) also decreases. Assuming that the trust score and relevance score (other two user preferences) are constant and plateau after certain iterations, we can state that once the user releases more data on the Internet, the bidder acceptance rate increases, hence showing a better click-through-rate. We also performed a t-test to confirm this through division of the data and cross validation to calculate differences in group means resulting in no significant difference. Hence the hypothesis is valid and stands.

Our privacy-payout relationship confirms [7] who discussed revenue maximization with budgets.

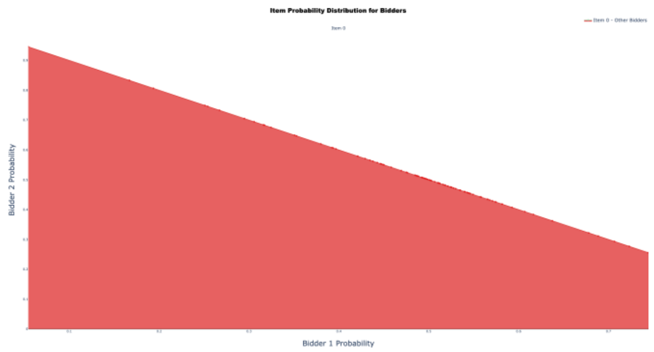

Fig. 8. Item Probability Allocation amongst bidders

Hypothesis: There is no equilibrium amongst bidders for item allocation across trials

Inference: Fig. 8 takes into account the item allocation probabilities throughout historical runs (100 in this case) and sorts them based on the winning bidder, which is showcased on the X-axis. The Y-axis shows the item allocation for the losing bidder. The line depicts the equilibrium state of allocation that exists for each iteration. To further prove this, we find the iteration where the gap in the bids of both bidders is the least to yield the maximum revenue to the auctioneer using *argmin*(). For all re-runs we always landed the point close to where both bidders had nearly 50% allocation probabilities indicating that model trained well to minimize the payout gap, yielding the maximum revenue generation whilst also keeping the auction fair by providing healthy allocation competition to the bidders. Hence, the initial hypothesis fails and for such a modeled game, there is an equilibrium.

The revenue maximization patterns observed align with [8] who established fundamental principles for optimal mechanisms with budget-constrained buyers. Our empirical results show...

V. Conclusions

In this research paper, we started with a problem statement highlighting the potential changes that advertisements might undergo with the transition to Web3.0 happening currently; users gaining more authority over personal data protection. The goal of this exercise was to certainly establish any impact of user preferences like - privacy threshold, relevance score and trust score on auctions. We studied existing online advertisement auctioning techniques to then propose a blend of game theory models: partial auction driven by Stackelberg leadership model paired with neural network based Vickrey auctions to incorporate 3 entities — users, bidders, and auctioneers and their preferences to play out auction simulations.

After extensive deep-dives into Online Auctioning Model (OAM) and analyzing its output in the form of modeling accuracy or visualizations, we can draw certain conclusive insights that may prove helpful to think upon -

- User preferences have a direct impact on the auction especially the payout, as the amount of data that the user wishes to withhold will directly affect the click through rate, and shooting up bidder valuations of the ad spots. Thus, for a bidder, be it an advertisement agency or a goods/service provider, they should focus on increasing user's trust on them with improving their own ad quality and spreading awareness about extra lengths of ensuring user data privacy and encryption

- This newly proposed model with stacked game theory approaches involving a Stackelberg game followed by a neural network-based Vickrey auction (like a simultaneous game) is plausible to implement and is statistically sound, even as a sophisticated version. A more complex model with a larger training data and additional attributes would probably result in a very optimized alternate auctioning model for auctioneer's to implement. (maximizing their revenue whilst maintaining competitive allocation nature)

Finally, this work can prove to be foundational for future work on modeling Auctions under Web3.0 architecture, when layered with more granular concepts like blockchain and data security.

Future Web3.0 implementations could leverage blockchain frameworks like [9] while marketing strategies adapt as shown in [10].

Data Availability

This paper is a hybrid of theoretical and empirical analysis. The data used in this paper is synthetic and simulated to closely resemble the real-world data. It has been generated using the code provided in the GitHub repository.

Conflict of Interest

The author declares no conflict of interest in the preparation and publication of this research.

Appendix A Code repository

The source code for this entire project can be found at the following public GitHub repository: https://github.com/Jwalin- Thaker/user-preference-online-ad-auctions.

References

[1] X. Jin, "The drive and support of web3. 0 to rich media advertisement." *Comput. Inf. Sci.*, vol. 1, no. 3, pp. 83–87, 2008.

[2] G. Aggarwal, S. Muthukrishnan, D. Pa´l, and M. Pa´l, "General auction mechanism for search advertising," in *Proceedings of the 18th interna- tional conference on World wide web*, 2009, pp. 241–250.

[3] C. Borgs, J. Chayes, N. Immorlica, M. Mahdian, and A. Saberi, "Multi- unit auctions with budget-constrained bidders," in *Proceedings of the 6th ACM Conference on Electronic Commerce*, 2005, pp. 44–51.

[4] A. K. Chorppath and T. Alpcan, "Learning user preferences in mechanism design," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*. IEEE, 2011, pp. 5349–5355.

[5] P. Dutting, F. Zheng, H. Narasimhan, and D. Parkes, "Optimal economic design through deep learning (short paper)," in *Conference on Neural Information Processing Systems (NIPS)*. Neural Information Processing Systems Foundation, Inc., 2017.

[6] M. Curry, T. Sandholm, and J. Dickerson, "Differentiable economics for randomized affine maximizer auctions," *arXiv preprint arXiv:2202.02872*, 2022.

[7] Z. Abrams, "Revenue maximization when bidders have budgets," in *Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm*, ser. SODA '06. USA: Society for Industrial

and Applied Mathematics, 2006, p. 1074–1082.

[8] Y.-K. Che and I. Gale, “The optimal mechanism for selling to a budget- constrained buyer,” *Journal of Economic theory*, vol. 92, no. 2, pp. 198–233, 2000.

[9] Y. Lin, Z. Gao, Y. Tu, H. Du, D. Niyato, J. Kang, and H. Yang, “A blockchain-based semantic exchange framework for web 3.0 toward participatory economy,” *arXiv preprint arXiv:2211.16662*, 2022.

[10] M. Cheng and X. Qiu, “Research on we-media marketing in web3. 0 environment,” *Management & Engineering*, no. 29, pp. 15–22, 2017.