Turning the tables in online ad auctions

Alex Steer

Final integrated assessment

Dissertation submitted in part-fulfilment of the regulations for the Postgraduate Diploma in Artificial Intelligence for Business

Saïd Business School, University of Oxford

2 October 2022

NB this is a public copy of the dissertation; some information has been redacted.

Turning the tables in online ad auctions

Abstract

The proposed AI application ('Turntable') will reverse the information asymmetry in online advertising auctions, transferring information advantage back to large advertisers and away from media platforms selling targeted advertising.

Specifically, it will enable a greater share of large advertisers' investment in digital media platforms to be optimised towards buying high-value ad impressions, rather than digitally-trackable outcomes. It will do this by predicting the differential expected value of different impressions on outcomes (e.g. sales), allowing advertisers to find underpriced impressions inside the 'walled garden' media platforms, and limit the passback of outcome data to these platforms.

This will benefit large advertisers by returning market power through information advantage, driving higher return on advertising spend. It will benefit [redacted] by creating a new source of value through IP and information scale, offsetting commoditisation of media buying services by technology.

The evolution of platform auctions

In 2022, 72% of all global paid advertising investment - $616 billion – will be spent in environments where the ad placement is determined dynamically by an algorithm at the point of exposure.[1] This is known as 'programmatic' ad placement because the allocation of ads – i.e which users see which ads, and in which contexts – is handled by software. Programmatic placement became the dominant form of ad placement globally in 2017.

Within this, 50% of all global ad investment will go into three large media and commerce platforms: Google, Meta and Amazon. These platforms offer ad targeting capability based on high-volume, high-granularity, high-frequency data that these platforms collect about their users' identities, behaviours, inferred interests and propensity, created by predicting behaviour based on user features to identify 'interest' and 'in-market' audiences (for example, people likely to buy a new car in the near future, or people actively looking to book flights from London to New York).[2] These platforms – sometimes called 'walled gardens' because they do not share user data with advertisers, just the ability to target ads to certain cohorts of users – offer global availability and high flexibility of when and where to spend advertising budgets, making it easy to move investment around between brands, products, markets, time periods, etc., to capitalise on shifting consumer demand.

The walled gardens use a real-time auction mechanism to determine which ads are showed to a given user in a given context (geography, time, and specific surroundings where an ad appears). This technique, known as real-time bidding, first appeared in online display advertising on publisher websites and is often called 'open-web' bidding.[3] In its original form, at the point at which a user appears in a

[1] For this and the following statistics see GroupM (2021).
[2] For a detailed overview of the business logic of how features are assigned to users and used to generate predictions, see Zuboff (2018).
[3] See 'Understanding Demand-Side Platforms' in Busch (2014).

given context, information about user and context is sent from the publisher (using software called a supply-side platform, or SSP) to all eligible advertisers (using a demand-side platform, or DSP). This information is broadcast in real time as a set of data – called a *bid request* – so each advertiser can evaluate whether the user is of interest, and decide whether to bid for the right to serve an ad. Open-web online ad auctions are held in software environments called *ad exchanges*, with the highest-bidding advertiser winning on a first- or second-price auction basis.[4]

Crucially, in an open-web programmatic auction, each advertiser sees the relevant information and decides whether to bid or not. Every bid is an automated decision based on a prediction – the expected return on ad-spend (ROAS) from the user if the ad is served. For example, suppose a winning bid price of $0.02 to serve an ad; an expected conversion rate from ad exposure to purchase of 5%; and an average purchase value of $5. Each impression served has an expected value of $0.25, hence an expected ROAS of $0.23 (an 11.5x multiple).

Walled-garden programmatic auctions ('platform auctions') represent an evolution of this approach, and have become the dominant form of auction over the last five years. In an open-web auction, information about the user is broadcast to each bidder, informing the decision to bid. In a platform auction, advertisers submit their targeting and bidding preferences to the platform: which audiences and contexts they are interested in, which outcomes they want to generate (e.g. impressions, clicks, website visits, online sales), and how much they are willing to spend. The

[4] For a technical analysis of open-web bid requests, see Olejnik (2014).

platform then determines which auctions each advertiser enters, and the bid price, ostensibly to maximise the delivery of outcomes for a given budget.[5]

I will show how this necessarily puts advertisers at an information disadvantage, by nature of the dual requirement of the platform to generate returns for advertisers and maximise yield for itself. Specifically, the platform auction model constrains the range of outcomes and makes it effectively impossible for advertisers to find and exploit pockets of excess value – things they know that the platform doesn't, that allow them to pay 'below the odds' for ad impressions. I then propose an AI application which reverses this information asymmetry by allowing advertisers to keep information out of the auction (and therefore out of the market) without penalising them for doing so.

The information asymmetry problem in platform auctions

Advertisers want to maximise ROAS, so they want to exploit temporary efficiencies in the market for attention. Certain kinds of knowledge about users in given contexts are valuable to advertisers, and to the platforms who auction the right to deliver targeted advertising. Someone is valuable if they:

- are strongly pre-disposed to buy in this category within a given time period, though have not yet been triggered to buy imminently
- are strongly pre-disposed to buy a particular brand when buying in category
- have a high typical spend per purchase

[5] See Amazon, Google, Meta (all 2022) for how this is explained to advertisers.

- have a high probability of repeat purchase if acquired
- are in-market and have a high probability of imminent purchase

Platforms apply their user data to make these kinds of predictions, because advertisers are willing to pay more to reach users who exhibit these traits. The expected value of an impression to an advertiser is the product of the average value of an outcome (e.g. purchase) and the likelihood of an incremental outcome given ad exposure. Brand propensity and category intent are functions that improve the expected outcome conversion rate within a given time period: you are more likely to respond to an ad for a brand you are pre-disposed to, and to do so quickly if you're already in the market – and both of these improve expected near-term marketing performance.[6] Therefore advertisers and platforms are seeking information arbitrage and are willing to incur costs to gather data that provides it.[7]

Performance marketers want to be in the right auctions and pay the right price to generate outcomes.[8] In open-web auctions, they collect bid request information and build predictive models based on all the user features broadcast, and historic bids won and lost, to predict a fair winning bid price. Publishers use similar data to set floor prices for different users based on expected bid prices. In an open-web auction there is information parity so long as advertisers and publishers are making predictions and decisions based on the same data. Each side is motivated to create information advantage by acquiring more features (e.g., from data brokers) that allow

[6] For classic summaries of the statistics of preference and recency-frequency effects on purchase, see McDonald (1992, 1996) and Broadbent (1999).
[7] For the concept of information arbitrage cost, see Grossman & Stiglitz (1980).
[8] 'Performance marketers' here refers to buyers of advertising 'remunerated on the basis of performance..for the leads or customers they acquire..when the results of marketing campaigns can be quickly recorded' (Kreutzer, 2022, p.25).

better predictions about users. All advertisers are free to decide which auctions to participate in.

In a platform auction the opposite is true. The advertiser asks the platform to be invited to certain auctions where a given user action (e.g. click) is for sale. The advertiser sets a budget and may set a maximum or average bid. The platform then decides which auctions the advertiser will bid in. A max or average bid constrains the number of auctions in which an advertiser participates.[9] A max bid implies a minimum number of auctions (e.g. at least ten wins to spend £100 with a £10 max bid). An average bid gives more flexibility to the platform to bid high or low across more auctions but implies an expected value of number of wins in a time period. Free budget allocation gives the platform maximum freedom.

This auction model represents a little-discussed principal-agent problem. The agent acting on behalf of the advertiser, deciding whether to bid, is controlled by the platform hosting the auction, which is also the seller of the ad inventory. Chen (2021) looks at the question of optimal auction design in keyword-based search auctions in the context of 'two distinct sources of information', where 'each consumer privately observes her search cost' (i.e. effort of searching) and 'each advertiser privately observes her probability' of driving a conversion, so 'these attributes are unknown to other consumers, advertisers, and the search engine'. However, this is not true of platform auctions in practice for two reasons, detailed below. First, platforms know the success probability of each advertiser (and ad), and incentivise advertisers to

[9] E.g. see Meta (2022, 1) on budgets and caps.

share conversion information back to improve auction performance. Second, the platform controls which auctions each advertiser enters, and how much they bid.

If the ‘auction agent’ were entirely in service of the advertiser, then in a market for fixed-value outcomes (e.g. where all clicks are equal in value), it would assess a user’s click probability and use this to predict the bid needed to win. To do this it would need access to auction data:

1. The features of previous ads (winning and losing) in previous auctions – e.g. brand, format, copy, image elements, destination, etc.
2. The features of users (as in a bid request)
3. The conditional probability of a click given the set of user features and ad features, if an ad is shown
4. The winning bid prices for ads of type X and for users of type Y

In short it would need access to the whole historic user stream, ad stream and bid stream, to build a model to predict the winning bid price for any combination of user and ad features. Specificially, it would want to predict the second price so it could bid just above it. Therefore to act optimally on behalf of the advertiser the auction agent needs to be able to see all auction data.[10]

An equivalent agent acting on behalf of the *seller* would want to predict bid prices in order to set a floor price in an open auction, wherever the item sold has durable

[10] For some attempts to predict optimal bid price using open-web auction data, see Wu (2015), Spentzouris (2018) and Ren (2019).

value. In an auction for clicks, it is better to refuse a sale for a low price if another auction can be held later, rather than saturate the user's attention unprofitably which may reduce the chance of a click later. So the agent acting for the seller also wants to calculate the conditional probability of a click given the number of different ads shown over a given time period, with diminishing-response and time-decay functions applied for frequency and recency.[11]

Now what happens if the auction agent is acting on behalf of *both* buyer and seller? The buyer wants a given number of clicks at the lowest price, so needs to predict the second price, and also predict whether the second price in future auctions will be higher or lower (to pace spend, and reach a forecast number of successful bids over a given time period). The seller wants to set a floor price or hold back inventory, to maximise yield.

So the 'buyer-seller agent' is incentivised to control which advertisers are invited to which auctions. This is possible by taking the ad, user and bid streams, and predicting the likely bids of all potential participants. Then, only those advertisers likely to bid in a way that clears the floor price are likely to be invited to bid on the auction. The agent also wants to predict the likelihood of future auction participation and winning, to avoid bidders leaving the platform because they don't win often enough. This is similar to the problem that casinos have to solve to keep players in the game, but in the platform's case it must solve it twice: both for advertisers and for users.[12] To avoid advertiser defection, it's in the seller's interest to invite advertisers

[11] See chapter 4 of Broadbent (1999), and Jin (2017).
[12] Chen et al. (2012)

only to auctions which they are fairly likely to win, and then only those they are likely to bid relatively high for.[13] To avoid user defection, platfoms 'weight' bids based on the predicted impact of the proposed ad on the likelihood of users returning to the platform.[14]

In effect this creates strong sorting: many auctions with few invitees, based on the expected bid-value of each click. Effectively, the buyer-seller agent says, 'This click is likely to sell for $1 so we will invite buyers likely to bid $1 to bid on it.' Within the context of an individual auction, this creates efficient clearing, with low risk to the seller.

Advertisers cannot choose which auctions they are in; they cannot see information about auctions they don't participate in'; and they cannot see their own bid prices except a max or an average. They rely on the sorting function of the buyer-seller agent to clear the market for them.

The buyer-seller agent is making predictions about how much they will pay and whether they will win. This is based on data from all advertisers. So priors based on other advertisers are being used to assess what they can bid on, and how much they can bid (especially where no max or average is applied). Given strong sorting, this

[13] That auction participation rates can be engineered in practice is suggested by documents included in discovery in the legal case State of Texas (2021), which alleges that Facebook and Google entered a 'Network Bidding Agreement' guaranteeing Facebook a 'bid rate' and 'win rate' in auctions on the Google platform. 'the agreement outlines that Facebook will use "commercially reasonable efforts" to bid on at least 90 percent of auctions in which Facebook recognizes the end user… The parties agreed up front on what Facebook's Win Rate in auctions would be. The..agreement specifies that Facebook would have a Win Rate of at least equal to 10 percent. The agreement terms require Facebook to bid high enough to win the minimum percent quota of 10 percent, irrespective of how high others in the auctions bid.'
[14] Meta (2022, 2).

means their ads are shown to users based on predicted similarity to other ads seen by these users and users like them. In other words, only certain types of people get to see certain types of ads.

This sorting effect keeps advertisers ‘in the game’ with regular wins and stable ROAS, and the sorting effect maximises yield to the platform. However, it reduces serendipity and has the effect of narrowing the ROAS spread, because the advertiser cannot adopt random or experimental strategies that may have a highly asymmetric payoff. The buyer-seller agent *may* even punish this if experiments (e.g. creative choices) make the ad’s performance less predictable using priors from other ads.

On balance this auction mechanic seems to give security to platforms at the expense of serendipity to advertisers. This affects large advertisers most, as they are more dependent on asymmetric performance to drive excess return than smaller advertisers, for whom merely increasing spend is a stronger lever.[15]

Feeding the beast

The principal-agent problem is compounded because platforms incentivise advertisers to share as much performance information as possible. Ordinarily, an advertiser would not wish to share information with the platform/seller about which

[15] Schmitt et al. (2010) describes the asymmetric response function from increased advertising spend. Large advertisers by definition have high share of voice so can drive less marginal improvement per dollar of excess adspend than smaller competitors, increasing reliance on finding asymmetric payoffs for their existing spend.

users are most valuable. In an open-web auction the advertiser can perform information arbitrage to identify under-priced users/contexts.

However, a platform auction removes this opportunity because the platform acts as agent and decides which auctions the advertiser participates in. This gatekeeping function, plus the strong sorting described above, actively penalises advertisers who do not signal to the platform which users are most valuable.

There are three main types of data passback from advertisers to platforms:

1. Placing platform data collection tags on brand websites and apps, so the platform can see which of its users are visiting a brand's site, buying from the brand online, etc.
2. Onboarding of lists of personally identifiable customer information (e.g. email addresses), so the platform can see which of its users are customers of the brand.
3. Selecting target audience criteria within the platform, so the platform can see (for example) 'Brand X is interested in reaching women aged 25-34 in New York City'.

Performance marketers who are trying to maximise short-term ROAS performance in individual auctions pass as much of this information back to the platforms as possible to improve the likelihood of ending up in the right auctions. Each of these data passback mechanisms transfers information from the advertiser to the platform before the auction. It allows the platform to pre-emptively score certain users as being of interest and value to certain brands, by identifying features and feature categories (e.g. geography, time, content types, occasions) that are predictive of a positive outcome and therefore valuable.

The platform is highly incentivised to use this information to create interest/propensity signals specific to individual categories, i.e. to identify which other brands a user might respond to given the ones responded to in the past, by that users or others like them. This means in practice that any response data from an advertiser is likely to end up as a propensity signal to other advertisers, even if the algorithms are just using a nearest-neighbour analysis rather than a defined category taxonomy.

Data passback becomes, in effect, a cost of entry to the right auctions. The same information is likely also used to identify which *other* advertisers should be invited to subsequent auctions. This transfers information advantage from an individual advertiser back to the platform, and in doing so erases the future ability of that data to generate market power. Once the knowledge that 'Person X is of value to advertisers like Advertiser Y' is in the market, its only remaining source of value is as an exclusion criterion – i.e. it lets Advertiser Y *decline* to bid on Person X because they are someone who can be reached using non-paid media (e.g. email marketing).

The information transfer (and therefore disadvantage) is greatest for large advertisers who represent a large share of impressions in the platforms. Their adspend is disproportionately informing the platforms, and smaller competitors.

Hence, assume that data passback mechanisms are benefiting the platform in the long run by allocating advertisers to auctions more efficiently, and shifting information advantage from the advertiser to the platform. This certainly leads to higher expected bid prices and narrower ROAS distributions (through the strong sorting effect of putting only highly-motivated, similar, willing-to-pay advertisers in the same auction), and could also lead to declining effectiveness because of category-level high frequency effects – e.g. showing an ad for shoes to someone who’s seen lots of ads for shoes, because shoe-sellers attend the auction each time.

<u>Beating the system</u>

What happens if advertisers withhold information? The types that can be withheld are audience (e.g. customer lists, retargeting pools) and conversion data. Removing these impairs auction-sorting: the buyer-seller agent cannot so easily determine which bid from which advertiser is most likely to deliver the biggest expected yield. By not signalling which users have high interest, intent and value, advertisers are more likely to be invited to auctions for users with a lower projected value to the advertiser, simply because they miss out on being classified as a most-relevant advertiser to users when they in fact are. So withholding audience pools and conversion data passback in an outcome-based auction may cost more in reduced ROAS than it saves by avoiding the yield inflation effect of a ‘winners invited’ strong sorting approach.

The advertiser has little to no control over the platform's knowledge of the user, or of how much information other advertisers give the platform. This makes it hard for the advertiser to impair the platform's knowledge at auction time. The platform will know that the user is interested in furniture, for example, and can predict this even for users who have not directly exhibited such information. So merely by announcing 'we sell furniture' to the platform, the advertiser begins to get sorted into some auctions and not others. (Nor is there any practical way for an advertiser *not* to signal what products and services it is advertising, as this would defeat the whole object of advertising!) The platform sees information about buyers, users and sellers (since the platform is the seller), which the advertiser does not. This allows it to sort advertisers into user auctions in a way that sets an effective floor price.

An advertiser can only beat the system if it can somehow identify people whose actual expected value is greater than the expected value predicted by the platform at auction time. The advertiser cannot pretend to be a different brand, and cannot fool the platform into thinking people are not interested in furniture when they are. So what can advertisers do?

If there is an information advantage to be had for advertisers, it is likely to be in those auctions that clear less efficiently, i.e. those where the data passback is weak for all advertisers. In a click auction, the platform immediately knows whether the user clicked, and this outcome can be fed back into the model space at user level just after auction time, potentially informing the user's next impression or the advertiser's next bid. The same is true, with a short time lag, for any outcome event that can be digitally tracked by the platform through data passback.

The extreme form of a non-clearing auction is an auction for impressions. Advertisers bid to serve ads to consumers, but no information is shared back with the platform about what happens next. The platform therefore knows the acceptable cost of each user to each advertiser at auction time (from the bid price), but *not* the expected value. In an impression auction, it is also easier for advertisers to disguise *why* they are bidding at a given price – the price may be a composite function of user profile features (e.g. 'aged 25'), contexts ('on a travel website'), locations ('Edinburgh'), time ('12.00pm'), recency/frequency ('third impression in the last week'), etc.

This returns information advantage to the advertiser. The more advertisers can bid on impressions, without exposing the *results* of those impressions, the more they can disguise their intentions in the auction. This increases the possibility that under-priced impressions can be identified and exploited for longer, rather than being re-priced by the platform based on data passback. We therefore want to shift spend away from buying trackable outcomes, towards buying towards ad delivery goals (e.g. reach and frequency) to given audiences in a given time period.[16]

However, to do this, advertisers need a way to predict *which* impressions are worth most. This means predicting the conditional probability of a valuable goal event (e.g. sale) given ad exposure, and score all audience features based on their relative importance. The advertiser can then bid on impressions, weighting spend towards

[16] For example reach and frequency buying with Meta (Meta, 2022, 3).

the features with highest affinity. In some platforms this can be done as a custom algorithm, in others more crudely as an audience targeting rule set.[17]

This approach is not wholly new: marketing mix models and simulations describing the incremental impact of ad exposure on business outcomes have been in common use since the 1960s.[18] What is new is the need, and opportunity, to use predictive models of this kind to execute high-frequency trading across many different buyable audience features, and multiple platforms, in real time, and to update models dynamically based on predictive accuracy (uplift in outcomes).

Given the information complexity involved and the need for rapid and iterative decision-making in where, when and how much to bid, there is a clear case for an AI application.

<u>The solution: "Turntable"</u>

The proposed application, **Turntable**, uses AI to predict the expected value of impressions to advertisers given available biddable features, and dynamically set bid prices for different features to automate bidding instructions into the walled gardens. Turntable will be used to increase return on advertising spend for large [redacted] advertisers. The data and modelling workflow of Turntable is illustrated in Figure 1 (with three example platforms: Amazon, Google, Meta) and described below.

[17] E.g. Google (2022, 2).
[18] E.g. Bass (1968), Kotler (1968).

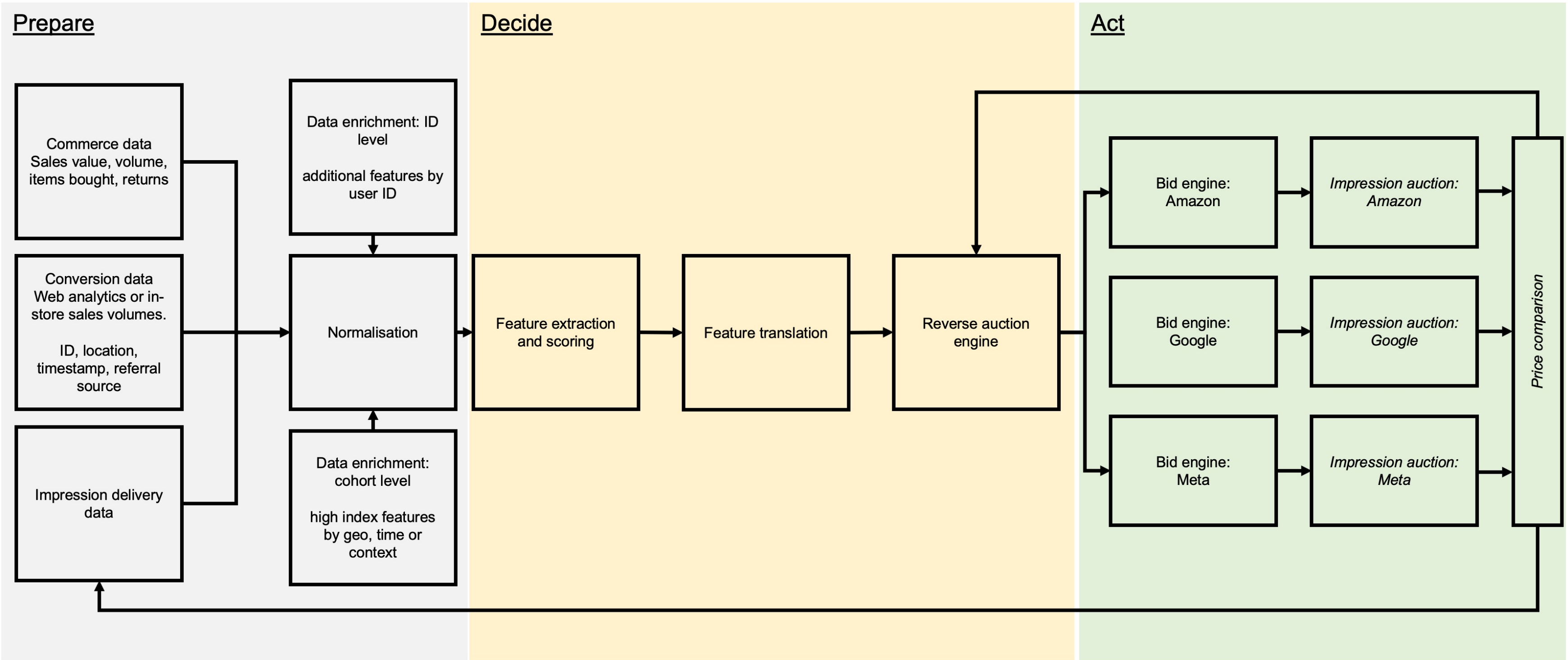

Fig. 1. Turntable workflow showing integration with three platforms (Amazon, Google, Meta)

The **Prepare module** assembles, organises and enriches three main types of data:

- **Commerce data** is volumetric and financial information: what has been sold, how many units, which products/services, revenue and gross profit. For businesses who sell directly this normally comes from a data warehouse and includes information at a customer level. For businesses who sell indirectly (e.g. through retailers) the data may come from third parties (e.g. Nielsen supermarket data) and be organised by store or sales channel. Commerce data gives us our measures of value.
- **Conversion data** tells us about the circumstances of sale – e.g. the channel, time, location, referral source. Increasingly this is from digital analytics systems and may be highly granular (e.g. showing which paid media campaigns referred traffic, which individual pages assisted conversion, which device a customer transacted on, etc.). It is typically matched to commerce data via a consistent identifier (e.g. user/customer ID in web analytics, store ID for aggregated sales channel data).
- **Impression data** is a measure of where and when paid media impressions were delivered, and to which groups. This is described in more detail below.

This data needs to be sourced and stored in a set of data environments (tables for commerce, conversion and impressions). It then needs to be normalised to create a common data set for modelling. Typically data of this kind can be normalised at two levels:

- ID level: sets of records sharing a common identifier denoting an individual user or customer
- Cohort level: well-defined sets of records with shared criteria – for example, all impression, conversion and commerce data associated with a particular geographic area, time period, store, etc.

Normalisation involves some standardisation of criteria, for example re-coding geographic or timestamp data into a common format. Modern graph database architecture is convenient for this because it allows records in the normalised data set to be traversed quickly along different criteria (e.g. select by ID, geography, timestamp, channel) without creating multiple aggregated tables.[19] This means predictive models can be built on the whole joined data set of key-value pairs.

The benefit of a graph model is that we can treat IDs (which denote one individual) and cohorts (which denote a group of individuals) as equivalent in data model terms as **feature keys**, i.e. entities to which values are attached. So for example, a purchase value and a timestamp (e.g. '$1000 spent on Wednesday') may be attached to each other, and to *either* a user ID ('spent by Alice Smith') *or* a cohort ('spent in Leicester', 'spent by iPhone users', etc). This means we can treat user-level and cohort-level features as analytically equivalent which will improve our models in the Decide module (below).

Where possible the data should then be enriched to add additional features. This enrichment is at feature-key level whether for IDs or cohorts, e.g. adding features

[19] Robinson et al. (2015)

from a data broker that give more facts or predictions about individual customers matched against their IDs; or at a cohort level using indexed likelihood scores from consumer panel data for certain features, e.g. 'this region overindexes on homeowners or buyers of Apple devices'.

The Prepare capability described above can be assembled from components that already exist within the major cloud computing providers and [redacted], speeding up development of Turntable. [redacted]'s data products business [redacted] has a series of APIs for normalising data at an ID level and at cohort level (using [redacted]), and matching against third-party data sets for feature enrichment and cohort-level scoring.[20] I propose to use these services to provide the Prepare module of Turntable.

The **'Decide' module** applies AI to this joined and enriched data set, to identify which features within the data predict incremental value. The target variable will come from the commerce data (typically a sales volume, revenue or profit figure). The 'Feature extraction and scoring' component within Decide does the following:

1. **Revenue lift.** For each feature-key type which has both revenue and impressions attached as values (e.g. ID, location, time period, device…), predict the marginal revenue given ad exposure. Then do the same for each key-value in the feature-key type (e.g. User A, Location B, Time Period C, Device D…). This can be done using a hierarchical modelling technique, for example a Bayesian hierarchical regression or a random forest, to solve for

20 [redacted].

matrix sparsity, so that each key-value pair is assigned a marginal value conditional on having been exposed to advertising. Ideally this will also account for the marginal effect of frequency – i.e. using a continuous function (e.g. a Hill function) rather than a logistic one to incorporate the diminishing response of approach.[21]

2. **Feature extraction.** For each key-value pair for which marginal revenue is predicted, extract the other features attached to that entity – for example, 'Location X has features A, B, C'; 'Sales Channel Y has features D, E, F…'. Build a feature importance model, scoring the expected marginal revenue given ad exposure to an entity (ID/cohort) with a particular feature. This can be done using gradient boosting (e.g. XGBoost) or Shapley/LMG regression.[22]

3. **Scoring.** Return a scored list of features existing within the joined data, each with a value for the expected incremental revenue per advertising impression served to that feature.

[21] See Jin et al. (2013) for an example of this approach using Hill functions to estimate diminishing response.
[22] Chen and Guestin (2016), Lindeman, Merenda and Gold (1980).

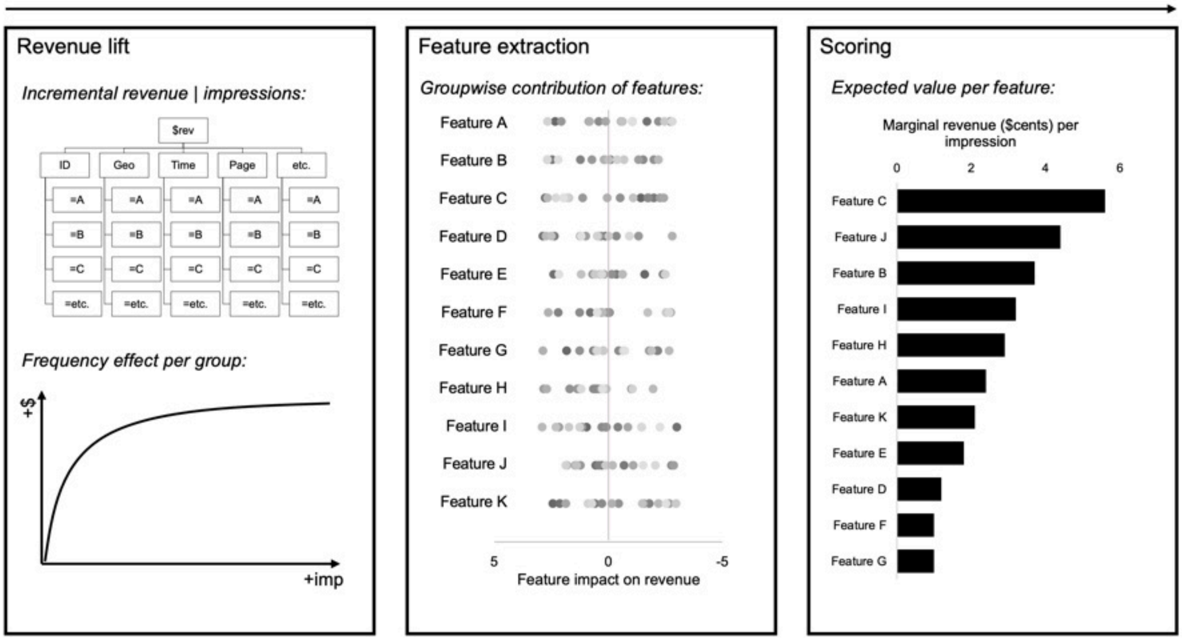

Fig. 2: Feature extraction and scoring model workflow

The list of features, each with its predicted revenue lift per impression, is then passed to the ‘Feature translation’ component. This is a set of data tables containing lists of features that can be bought in each walled garden platform: e.g. targetable audience definitions, locations, etc. These lists are typically available for download from the platforms’ buying tools. The Feature Translator uses natural-language AI (e.g. GPT-3) to find the closest-matching buyable features in each platform lists, and create a match table. Figure 3 shows the result of a simple proof of concept demonstrating this:

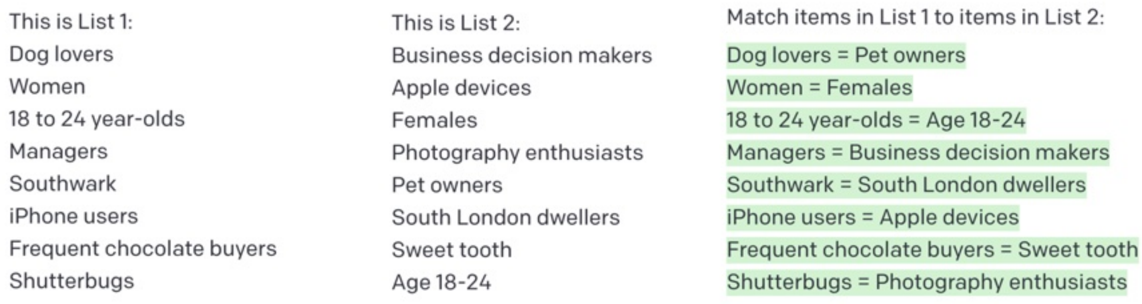

Fig. 3: Correspondence mapping between feature labels using GPT-3

Turntable now has a list of scored features with a predicted value attached to each, matched to buyable features in each platform. This information is passed to the 'Reverse Auction Engine'. This is a set of API connections that regularly pulls in availability and cost information from the tables, for each feature in the table. Because Turntable knows (a) the expected value per feature to the advertiser, and (b) the volume and cost of each feature across platforms using the match table, it can allocate auction budgets optimally per feature, per platform. This creates a reverse of the platform auction dynamic: one advertiser, effectively inviting many platforms to disclose availability and price for equivalent features, so the advertiser can prioritise those that are under-priced.

The **Act module** executes automated campaign setup and media buying based on information from the Reverse Auction Engine. For each platform, a Bidding Engine component connects to the platform's buying tool APIs and sets up campaigns defining the features, geographies and time periods, setting a total budget and max bid for each based on the predicted value. In simpler platforms, each desired feature set is built as a separate target audience or line-item, each with its own budget and max bid. In more sophisticated platforms that allow programmable bidding, each feature can be included in a custom bidding algorithm with a bid upweight set per matching feature.[23]

Finally, the 'Price comparison' component retrieves daily spend and delivery data per feature group from the platforms' reporting APIs and feeds these back into the reverse auction engine and the impression data store in the Prepare module. This ensures Turntable is always deciding based on an up-to-date view of delivery and cost.

Impact and change management

Turntable's main impact is on the problem stated above. By predicting marginal revenue per impression for different buyable features, and automating in-platform buying based on that information, it lets large advertisers bid on outcomes without passing outcome data back to the platforms, regaining information advantage in the auction, so advertisers can find under-priced features and improve ROAS.

Second, Turntable benefits [redacted] in two ways. It creates valuable IP that can be licensed to advertisers as software, hedging against commoditisation of media buying services and in-housing of buying. It also creates a new economy of scale. As well as implementing Turntable per client, I propose to create a 'feature co-op' where [redacted] clients can choose to share information about the value of features for mutual benefit: for example, an airline and a hotel chain might use federated learning

[23] For a summary of custom bidding algorithm design see Gilbert (2020) ch.10.

techniques to train Turntable to find features that drive revenue for both business-class flights and hotel rooms.[24] This creates a new incentive for large advertisers to use [redacted] to have access to this pooled feature information.

Turntable can be built in [redacted] existing cloud infrastructure and using some existing components for the Prepare module. The Decide module will be a novel build. Workflows for automating aspects of campaign setup and buying in the Act module already exist, as do contracts with systems for aggregating and automating collection of reporting data for the price comparison component.

The biggest organisational change will be to the relationship between platform buying teams and data science teams. [redacted]

When using Turntable, buyers' work does not change but their data source does: they will use data about the incremental results (e.g. sales) generated by each part of the campaign, from Turntable's models. This will require significant re-education of buying teams in the short term to trust and use results built on client data not platform metrics. Data science teams configuring Turntable for clients will need to generate new data outputs (e.g. performance dashboards) as often as daily, for use by performance teams. In the longer term, much of the work of both teams – making predictions and taking buying decisions – will be automated, and data science and buying teams will focus more on integrating new data sources into Turntable's

[24] Federated learning is collaborative model training without data sharing. See Martineau (2022).

models, designing specific tests, and measuring the impact of specific innovations (e.g. new media formats), rather than measuring or executing business-as-usual buying.

Ultimately, Turntable represents the convergence of econometric modelling and performance media buying to create a new approach, where advertisers can buy ad impressions in online auctions based on expected sales effect, without giving information advantage to the platforms. It shifts the work of data scientists and buying teams from delivering manual measurement and activation, to supporting progressive automation. Finally, it creates new sources of advantage for [redacted]: license revenue through IP as software, a defence against commoditisation in media buying, and efficiencies through automation of key media planning and buying tasks. I believe it is a valid and important case for an AI application within our business.

References

Amazon (2022) ‘Ad display auction’. https://sellercentral.amazon.com/gp/help/external/G201528470?language=en_US. Accessed 15 August 2022.

F.M. Bass et al. (1968) *Applications of the Sciences in Marketing Management.* Wiley.

S. Broadbent (1999). *When to Advertise.* Admap.

O. Busch (2014) *Programmatic Advertising: The successful transformation to automated, data-driven marketing in real-time*. Springer.

M. Chen et al. (2012), ‘A revenue management model for casino table games’, *Cornell Hospitality Quarterly* xx. 10, 1-10.

T. Chen & C. Guestin (2016) ‘Xgboost: A scalable tree boosting system’, KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Y.-J. Chen (2021) ‘Optimal design of revenue-maximizing position auctions with consumer search’, *Production and Operations Management* xxx. 9, 3297.

P. Gilbert (2020) *Join, or Die: Digital advertising in the age of automation,* Mill City Press.

Google (2022) 'How the Google Ads auction works'. https://support.google.com/google-ads/answer/6366577?hl=en-GB. Accessed 15 August 2022.

Google (2022) 'Implement custom bidding'. https://developers.google.com/display-video/api/guides/managing-line-items/custom-bidding. Accessed 20 August 2022.

S.J. Grossman & J.E. Stiglitz (1980) 'On the impossibility of informationally efficient markets', *American Economic Review* lxx. 3, 393-408.

GroupM (2021) *This Year, Next Year: Global end-of-year forecast*.

Y. Jin et al. (2017), 'Bayesian methods for marketing mix modeling with carryover and shape effects'. Google research papers. https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/46001.pdf. Accessed 20 August 2022.

Y. Jin et al. (2013) 'The optimal mix of TV and online ads to maximise reach.' Google research papers. https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/41669.pdf. Accessed 20 August 2022.

P. Kotler (1968) ‘Mathematical models of individual buyer behavior.’ *Behavioural Science* xiii. 4, 274-87.

R.H. Lindeman, P.F. Merenda and R.Z. Gold (1980) *Introduction to Bivariate and Multivariate Analysis*, Scott, Foresman and Co.

R.T. Kreutzer (2022) *Online Marketing*. Springer.

K. Martineau (2022) ‘What is federated learning?’, *IBM Research Blog* 24 August 2022. https://research.ibm.com/blog/what-is-federated-learning. Accessed 2 October 2022.

C. McDonald (1992) *How Advertising Works: A review of current thinking*. Advertising Association/NTC.

C. McDonald (1996) *Advertising Reach and Frequency*, 2nd ed. ANA/NTC.

Meta (2022) ‘About ad auctions’. https://www.facebook.com/business/help/430291176997542?id=561906377587030. Accessed 15 August 2022.

Meta (2022) ‘About ad relevance diagnostics’. https://www.facebook.com/business/help/403110480493160?id=561906377587030. Accessed 20 August 2022.

Meta (2022) ‘About reach and frequency buying’. https://www.facebook.com/business/help/251123081984768?id=842420845959022 Accessed 20 August 2022.

L. Olejnik et al. (2014) ‘Selling Off Privacy at Auction’. Conference paper, *Network and Distributed Systems Security Symposium*.

K. Ren (2019) ‘Deep Landscape Forecasting for Real-time Bidding Advertising.’ Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

I. Robinson et al. (2015) *Graph Databases*, 2nd ed, O’Reilly.

J. Schmitt et al. (2010) ‘Asymmetric advertising impact’. Proceedings of the 39th European Marketing Academy Conference (EMAC).

P. Spentzouris et al. (2018) ‘Advertiser Bidding Prediction and Optimization in Online Advertising.’ 14th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), 413-24.

State of Texas et al. (2021), *In re: Google digital advertising antitrust litigation*, case 1:21-md-03010-PKC, doc 152, p.81-2.

Uber (2018) ‘H3: Uber’s Hexagonal Hierarchical Spatial Index’, *Uber Engineering blog*. https://www.uber.com/en-GB/blog/h3/. Accessed 30 September 2022.

W. Wu et al. (2015) ‘Predicting Winning Price in Real Time Bidding with Censored Data.’ Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1305-14.

S. Zuboff (2019) *The Age of Surveillance Capitalism*. Profile Books.