# Delivery Optimized Discovery in Behavioral User Segmentation under Budget Constraint


Harshita Chopra*
Adobe Research, India
harshitac@adobe.com

Atanu R Sinha*
Adobe Research, India
atr@adobe.com

Sunav Choudhary
Adobe Research, India
schoudha@adobe.com

Ryan A Rossi
Adobe Research, USA
ryrossi@adobe.com

Paavan Kumar Indela†
Adobe Research, India
indela.paavankumar@gmail.com

Veda Pranav Parwatala†
Adobe Research, India
pvedapranav@gmail.com

Srinjayee Paul†
Adobe Research, India
srinjayeepaulsep@gmail.com

Aurghya Maiti†
Adobe Research, India
aurghya.kgp@gmail.com



## ABSTRACT

Users' behavioral footprints online enable firms to discover behavior-based user segments (or, segments) and deliver segment specific messages to users. Following the discovery of segments, delivery of messages to users through preferred media channels like Facebook and Google can be challenging, as only a portion of users in a behavior segment find match in a medium, and only a fraction of those matched actually see the message (exposure). Even high quality discovery becomes futile when delivery fails. Many sophisticated algorithms exist for discovering behavioral segments; however, these ignore the delivery component. The problem is compounded because (i) the discovery is performed on the behavior data space in firms' data (e.g., user clicks), while the delivery is predicated on the static data space (e.g., geo, age) as defined by media; and (ii) firms work under budget constraint. We introduce a stochastic optimization based algorithm for delivery optimized discovery of behavioral user segments and offer new metrics to address the joint optimization. We leverage optimization under a budget constraint for delivery combined with a learning-based component for discovery. Extensive experiments on a public dataset from Google and a proprietary dataset show the effectiveness of our approach by simultaneously improving delivery metrics, reducing budget spend and achieving strong predictive performance in discovery.



## CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; **Machine learning**; • **Information systems** → **Data mining**.

## KEYWORDS

Behavioral Segmentation; User Segment Discovery; Media Delivery; Media Selection; Budget Constraint; Media Mix Modeling


*Equal Contribution.
†The work was done while authors were in Adobe Research, India.



*CIKM '23, October 21–25, 2023, Birmingham, United Kingdom.*



ACM ISBN 979-8-4007-0124-5/23/10…$15.00
https://doi.org/10.1145/3583780.3614839

**ACM Reference Format:**
Harshita Chopra, Atanu R Sinha, Sunav Choudhary, Ryan A Rossi, Paavan Kumar Indela, Veda Pranav Parwatala, Srinjayee Paul, and Aurghya Maiti. 2023. Delivery Optimized Discovery in Behavioral User Segmentation under Budget Constraint. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3583780.3614839


## 1 INTRODUCTION

The ubiquitous online user behavior data afford opportunities for behavioral user segmentation to online firms. With ever more sophisticated algorithms, firms leverage the data of its own users to discover segments' propensities, behavioral tendencies, etc. to send messages, make predictions, offer recommendations, and improve users' experiences. Behavioral user segments (hereafter, segments) are fundamental blocks for firms to target different segments with different messages and offerings (hereafter, messages) [20]. **Discovery** or formation of behavior segments, is only the first act and becomes ineffective unless messages can be actually delivered to the segments. **Delivery** of messages to the segment is the second act. Increasingly, delivery occurs through media channels (hereafter, media) such as Facebook, Google and others, as exemplified by media spend's strong growth approaching 300 billion USD [2]. Two uncertainties facing the firms thwart delivery: (1) only a portion of a behavior segment find **match** in a medium; and (2) only a fraction of those matched actually see the message, or has **exposure** to the message. In media parlance [3], when a message is sent to a segment, **Reach** occurs provided *both* match and exposure are realized. Yet, even advanced algorithms for discovery, unsupervised or supervised, ignore these uncertainties of reach inherent in delivery [10, 11, 15, 17, 19, 21, 29]. Instead, those focus only on performance for discovery while tacitly assuming away these delivery roadblocks. This is a problem and a research gap.

Firms perennially face the problem of segmenting its users (customers) for effective targeting, and allocating segments thus formed to different media within a media budget [6]. Figure 1 is a pictorial sketch of this problem. When the discovery is done independently of delivery consideration, suppose three behavior segments are formed. Then, as delivery with static data is decided, a typical outcome is that each segment gets mapped to multiple media. If the firm wants to deliver a message specifically aimed at the red-dots behavior segment, which is spread across two media, two problems

arise: the budget can be exceeded, and the message is delivered to some users in blue-dot and black-dot segments, a potential waste. To avoid these problems it would be prudent to account for delivery, during discovery of segments, by considering alignment with the three media.

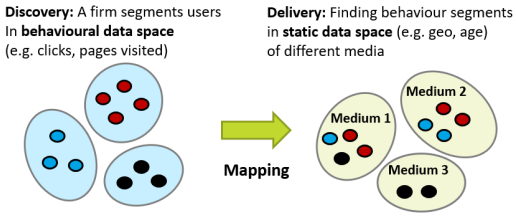


**Figure 1: Need for Delivery Optimized Segment Discovery**

The formal research problem is:

1. How to improve match across media with the goal of improving reach for delivering communication to users?

2. How to address (1) for endogenously formed audiences (as opposed to exogenously defined audiences as in the prior art) and multiple media?

3. How to address (1) and (2) within a budget?

4. How to also obtain high accuracy on predicting conversion?

Achieving the *dual goals of high conversion accuracy and reach maximization*, embedded in the research problem, is compounded by two other factors. (a) the discovery of behavior segments is performed on the firm's behavior data space (*e.g.* users' click, page visit, product viewed), all of which exist in users' data the firm possesses, but do not exist in the data of users the media possess. The media have users' static data (*e.g.* geo, age, interests) on which the delivery is predicated. (b) Firms work under tight budget constraints since media are costly. Elaborating on (a), users' visits to a firm's site or app generate a user behavior log of *pageurls*, analyzing which users are clustered into behavior segments, in a supervised or unsupervised manner. Behaviors are dynamic data. For the same users, static data such as geo, age, and interests are available to the firm. When a firm ports over a behavior segment to a medium for sending messages to the segment, the firm defines a set of static characteristics (*e.g. country-us, age-35-40, interests-jazz*) that "best" represents the behavior segment. Then this set, along with the message, are passed along to a medium, which does the delivery. The medium relies on static data to find users with (*country-us, age-35-40, interests-jazz*) from among the medium's vast user base.

A question may arise that the firm can bypass the behavior segmentation step and instead form segments in the static data space. However, behavior data are much more predictive of a user's decision (*e.g.* whether to convert, whether to renew a subscription) than static data [20], reinforcing the premium put on behavior segments. For effectiveness, it is necessary to maintain the primacy of behavior segmentation in the discovery phase and map to the static data for delivery.

Coming to (b), the firm can spend the media budget on various combinations of different behavior segments and different media. Different behavior segments, defined by their sets of static characteristics, result in different proportions of match and exposure, and these proportions differ across different media. Further, the cost of sending a message to a user varies by media. Thus, a firm's spend depends on the composition of users in each behavior segment, those users' representation in static data (which determines match and the exposure proportions), and the cost per medium. It follows that the discovery of behavior segments is intrinsically linked to the delivery and the spend; however, the prior art misses this linkage.

To address these limitations, we propose joint optimization of discovery and delivery for behavioral segmentation and optimize subject to the media spend budget constraint. We address Reach, a common currency for media spend [3], which maps to *pay-per-click*. We focus on *Direct* [4] media spend and not on programmatic ad-bidding spend. For reach maximization subject to a budget, two different forms are modeled - one, based on mean squared error (MSE); and two, based on optimization under constraint [14]. In each form we present multiple models; totaling five proposed models. Moreover, since the space of behavior segments is different from that of static data, a separate network learns a mapping function *Beh2Stat* such that a behavior segment can be expressed in terms of reach, through the match and exposure rates, to make stochastic assignments to the media. Five performance metrics are used to span conversion prediction, spend and reach efficiency and effectiveness. We find that our proposed Delivery Aware Discovery (**DAD**) models produce predictive conversion accuracy, AUROC, comparable to models focused only on discovery (**DISC**), and yet reduces Spend and increases Reach. In addition, within our five proposed models, the Augmented Lagrangian stochastic optimization model has the best performance in closeness of spend to the budget, and better than that of the MSE based models.

**Summary of Main Contributions.** Focusing on firm's *direct* media spend [4], not programmatic spend, our key contributions are:

- A new research problem of delivery aware discovery in behavioral user segmentation.
- A new model for joint optimization of discovery and delivery, under budget constraint.
- Recognizing the mapping from firm's behavioral data space to the media's static data space since behavioral segments are not realizable on media platforms.
- Performing stochastic optimization for the dual goals of (a) predictive segmentation and (b) optimizing reach.
- Introducing two new metrics for spend and reach efficacy.

## 2 RELATED WORK

Segmentation of users is a well-studied problem. Clustering approaches are commonly used for segmentation, emanating from statistical cluster analysis in an influential 1963 paper [32]. As online data have evolved, so have clustering algorithms [10, 11, 15, 17]. Both unsupervised and supervised clustering methods dot the literature [19, 21, 29]. In particular, a recent paper [24] performs temporal predictive clustering on dynamic user data. The outcome, conversion, is our prediction objective, for which segments are formed from event-level-behavior-sequence data (logs). That is, our goal is not to cluster based on the behavior sequence per se, as papers [31] on unsupervised sequence clustering do. A large literature of deep clustering [26, 33, 35], time series clustering [9, 27], and progression of diseases [25, 28] do not address the research questions we

**Table 1: Statistics for the datasets used**

| | Dataset I | Dataset II |
|---|---|---|
| #Train Users | 1664 | 26292 |
| #Test Users | 416 | 6572 |
| #Sessions per user: | | |
| Min | 1 | 3 |
| Max | 8 | 8 |
| #Pages per session: | | |
| Min | 2 | 5 |
| Max | 40 | 40 |

tackle. All these works address only the discovery of user segments, but none incorporates the delivery. On the discovery side of the equation, the state of the art (SOTA) is the paper [24]. Without any prior art for delivery aware discovery and our context of predictive clustering with users' dynamic, behavior data, we use [24] as the baseline SOTA model.

The problem of advertising (message) delivery under budget constraint occupies an early prominent role in search [18, 22]. Optimization of message spend rate with budget is well attended [8, 23]. An area of research emphasis is budget pacing, whereby a given budget from a firm is dynamically allocated over a time horizon, based on anticipated traffic with specified characteristics, as exemplified by the empirical and theoretical contributions [12, 13, 16, 30, 34], to name a few. This important body of research makes strong contributions toward managing delivery of messages to users within budget and addresses the context of ad bidding. None of these works considers the discovery or formation of segments; they start with a given set of users. We model discovery and delivery jointly. Our work does not address the ad bidding based media spend; instead we address the direct media spend, whereby the cost of a message to a user is known and is not an outcome of bidding. We also do not perform pacing of messages.

# 3 DATA

## 3.1 Firm data

The two datasets contain time stamped behavioral activity of users, for each user. Dataset I is proprietary, from a provider of SaaS services to individual users. Dataset II is public, from Google Analytics [1]. For each dataset, we work with the site's pages (hereafter, page-urls), where the behaviors are captured in the format of page-urls. A user visits the site on multiple sessions, and in each session (visit) browses multiple pages. The logs show the time-stamped sequence of page-urls for each user as they click on pages, and are mapped to the user with an anonymized code. Additionally, each dataset has a binary target label for conversion, corresponding to each session. Statistics for the datasets are shown in Table 1. These statistics are quite comparable to the industry standard *average* number of pages per user, per session, ranging from 10 to 31 [5].

For the data at hand, behaviors are page-names in the form of page-urls, common for online browsing data. In other examples, such as campaign, behaviors can be open email, click email, unsubscribe, etc. Other target variables of interest to any firm can be used, including target variables with more than two classes.

Processed public data are available [here].

## 3.2 Media data

Data of match rate and exposure rate are media dependent (*e.g.* different across Facebook, Google-YouTube, TikTok), and specific to the combination of static characteristics. In our proprietary data, there are 5 static characteristics, (*country, source, member-type, browser, os(operating system)*). As an illustration, the 5-tuple, static characteristic set (*country-us, source-bookmarked, member-frequent, browser-chrome, os-windows*) has a match rate 0.56 and exposure rate 0.47 for a medium $j$; while for another 5-tuple (*country-uk, source-bookmarked, member-infrequent, browser-edge, os-iOS*) those values are 0.39 and 0.61 for the same medium $j$. However, in another medium $j'$, the former 5-tuple (*country-us, source-bookmarked, member-frequent, browser-chrome, os-windows*) has values 0.45 and 0.69. The match and exposure rates vary across the combinatorial set of static characteristics, for each medium. These data are available from a medium to the firm which advertises on the medium through reporting APIs but are *not publicly* disclosed. To overcome this, we move to generate a match rate and exposure rate from a joint distribution over the support of the combinatorial set of static characteristics, which implies that for every combination (tuple) of the 5 static features, a match rate and an exposure rate, lying between 0.25 and 0.75 are drawn, for a medium. This is repeated for each medium. The match and exposure rates are held fixed for all the models. The cardinality of the set of 5-tuple static characteristics in our data is $1,296$. We thus have a table of pairs of (match rate, exposure rate) for each $1,296$ five tuples and three media.

From the Google data, we use 6 static features, each categorized as follows: (*medium [organic, referral/cpc/cpm, others]; deviceCategory [desktop, mobile/tablet]; operatingSystem [Macintosh/iOS, Windows, others]; categorized-geo [North America, Asia, Europe, others]; browser [Chrome, Safari and others]; source [direct, Google, others]*). Match and exposure rates for the Google data are generated in the manner described in the above paragraph. These static characteristics and their categories yield a table of 432 six tuples and three media, giving us pairs of (match rate, exposure rate) for each of 432 six tuples, for each of three media.

# 4 JOINT DISCOVERY-DELIVERY MODEL ARCHITECTURE

To achieve the dual goals of (i) conversion prediction and (ii) reach maximization subject to a budget, we introduce a two-part network. Figure 2 summarizes the notation and depicts the model architecture. In the Figure, green colored network is for goal (i), and black for goal (ii). Additionally, as a necessity to map behavioral data of users to static characteristics required by media, we introduce a mapping function *Beh2Stat*. The match rate and exposure rate vary by static characteristics and medium. The cost varies by medium. The *Beh2Stat* function's mapping to the static space, allows computation of cost of reach, which is then taken to the budget constraint. Training the whole network in three steps as shown in Figure 2 achieves the joint optimization. The SOTA in discovery [24] is preserved by green network.

| | |
|---|---|
| $x$ | input, user behavior |
| $y$ | target label, conversion |
| $i$ | index for segment |
| $j$ | index for medium |
| $t$ | time step for user behaviors |
| $\tau$ | total number of time steps of user behaviors |
| $z$ | intermediate user representation (embedding) |
| $\pi$ | users' segment assignment probabilities |
| $c$ | sample from categorical distribution of $\pi$ |
| $B$ | total spend budget |
| $\chi_j$ | cost per user in medium $j$ |
| $n_i$ | number of users in $i$-th segment |
| $\mathcal{S}$ | static characteristics of a segment |
| $\rho_{ij}$ | match rate of $i$-th segment assigned to $j$-th medium |
| $\eta_{ij}$ | exposure rate of $i$-th segment assigned to $j$-th medium |
| $A_{ij}$ | 1, if segment $i$ assigned to medium $j$; 0, otherwise |

1. Pre-Train Encoder-Predictor, Selector
2. Train Actor(Encoder, Selector)-Critic (predictor), Embeddings
3. Train Selector, Predictor, Reach

Separately, Train Beh2Stat after Step 2, before Step 3

$e_i$ Behavior segment $i$ centroid embedding

$S_i = Beh2Stat(e_i)$, Behavior segment $i$ static representation

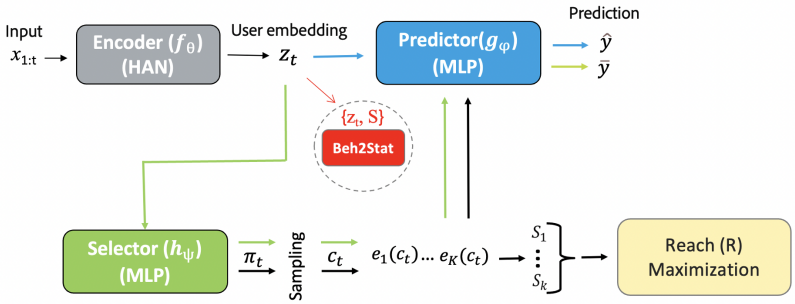


**Figure 2: Overview of approach.**

## 4.1 HAN Encoder

**Encoder** $f_\theta$: $\prod_{i=1}^{t} \mathcal{X} \rightarrow \mathcal{H}$. The encoder takes user behavior till time $t$ as input represented by $\mathbf{x}_{1:t} = \{x_1, x_2, \dots, x_t\}$ for $\mathbf{x}_t \in \mathcal{X}$, and learns to produce an intermediate user-level representation $\mathbf{z}_t \in \mathcal{H}$, by predicting each user's target label, as in, $\hat{y}_t$. Note $\mathbf{z}_t$, a hidden vector in the latent space $\mathcal{H}$, is a representation that embodies the latent tendency of a user, and is used for the clustering task. The encoder used here is a Hierarchical Attention Network (HAN) [36]. We choose HAN since it better encapsulates the two-level sequence of user behaviors - one, multiple pages browsed in each session and two, multiple sessions of each user. The first level encodes activities within each session to a session-level vector and the second level encodes the session-vectors to a user-level vector.

## 4.2 Predictor, Selector, Embedding Dictionary

**Predictor** $g_\phi$: $\mathcal{H} \rightarrow [0, 1]$ is a fully connected network that takes an embedding $z$ as input and predicts the target, i.e., conversion probability $y$.

**Selector** $h_\psi$: $\mathcal{H} \rightarrow \Delta^{K-1}$ is a fully connected network that takes a user embedding $z$ as input and computes a distribution $\pi$ where $\pi(k)$ is the probability of the embedding $z$ being assigned to the $k$-th cluster.

**Embedding Dictionary** $\mathcal{E}$: This is a dictionary of the centroids of $K$ clusters. Given a sampled cluster assignment $c_t$, it outputs the centroid embedding $\mathbf{e}(c_t) \in \mathcal{H}$.

## 4.3 Beh2Stat

*Beh2Stat*. $b_\omega$: $\mathcal{H} \rightarrow \mathcal{S}$ is a fully connected network that learns the function mapping user behavioral embedding $z_t$ to the user static characteristics vector $S \in \mathcal{S}$. Note that, by definition, *static* characteristics of a user do not depend upon $t$. Once trained, this function projects the $i$-th behavioral segment-specific centroid embedding $\mathbf{e}(c_t) \in \mathcal{H}$ to the $i$-th segment-specific static characteristics $S_i \in \mathcal{S}$. The projection is necessary to assign match rate $\rho_{ij}$ and exposure $\eta_{ij}$ to the $i$-th segment, since for any medium $j$, $\rho_{ij}$ and $\eta_{ij}$ are defined in terms of static characteristics $S_i$, but not in terms of behavioral embedding. The $\rho_{ij}$ and $\eta_{ij}$ are used in the reach computation, defined later. Specifically, to preserve an important property, that the behavioral segment contains users with a variety of static characteristics, we project a probability distribution $p$ over the support of $\mathcal{S}$ and use that probability distribution to compute expected reach. This affords stochastic optimization of the objective, reach. The loss associated with Beh2Stat is given by $\mathcal{L}_\mathcal{B}(\omega) = -\sum S \log p(S)$.

# 5 OBJECTIVE FUNCTION AND CONSTRAINTS

## 5.1 Behavioral Segmentation Objective

The input $\mathbf{x}_{1:t}$ are user behaviors present in the sequence of page urls clicked. The output $y_t$ of a user denotes whether a conversion was observed or not. The objective of behavioral segmentation is to cluster users into segments based on users' embeddings $\mathbf{z}_t$ such that segment-wise average prediction of conversion performs well. The following describes the training.

## 5.2 Reach Maximization Objective

We assume that each segment is assigned to only one medium (*i.e.* one media channel), but multiple segments can be communicated through the same medium. During training, an MLP $v_\delta(\mathcal{S})$ learns to map static features to mediums. For $i$-th segment's static characteristics $S$, given a medium $j$, match rate and exposure rate coming from a table of match and exposure rates, as stated in Section 3.2. The *reach* objective is denoted by $\mathcal{L}_R$ and is calculated as follows.

The loss, $\mathcal{L}_4$, for the joint optimization of discovery and delivery is,

$$\mathcal{L}_4(\theta, \phi, \psi, \omega, \delta) = \mathcal{L}_A(\theta, \psi, \phi) + \mathcal{L}_C(\phi) + \mathcal{L}_R(\delta) \quad (1)$$

where, $\mathcal{L}_R$ is the loss for the reach maximization. Note that $\mathcal{L}_R$ changes with the specific formulation of the optimization's objective function. $\mathcal{L}_A$ denotes Actor loss and $\mathcal{L}_C$ denotes Critic loss, both of which are defined in Section 6.

Since each segment is activated on only one channel, the channel should be selected to maximize the reach of that segment. Hence, $i$-th segment's reach $R_i = \max_j(A_{ij}\rho_{ij}\eta_{ij})$. Expected total reach $R$

is the sum of the reaches of individual segments and is given by

$$R = \sum_i R_i = \sum_i \max_j (A_{ij}\rho_{ij}\eta_{ij})n_i \tag{2}$$

*5.2.1 Mean Squared Error (MSE).* The first form of reach maximization is based on MSE. The total budget is split among the different media proportionate to the number of people it contains, which is the sum of the number of people in all segments assigned to that media. Depending upon match rate $\rho_{ij}$, and exposure rate $\eta_{ij}$ of the static characteristic tuple of the segment to which the user belongs, the spend for reach differs across users in different segments and across different media. Note that $(\rho_{ij}, \eta_{ij})$ vary by static characteristic tuples and by media. Segments formed through the Selector yield the size $n_i$, for the $i$-th segment, where $i = 1, ..., K$, and $K$ is the number of segments. The Selector yields the group of users in segment $i$, whose latent, segment-centroid embedding $\mathcal{E}_i$ is passed to the *Beh2Stat*, which outputs $i$-th segment's static characteristics tuple. Given this tuple, corresponding $(\rho_{ij}, \eta_{ij})$ for each medium $j$ are read from a table for segment $i$. For MSE, per individual user in segment $i$, medium $j$, the expected reach = $\rho_{ij}\eta_{ij}$. Per individual user assigned to medium $j$, the reach goal = $B/N\chi_j$. The intuition is that with $N$ users, and cost per user reached $\chi_j$, the budget $B$ is divided into a reach goal per user. Two variations of MSE are modeled as loss functions and described below.

**Cluster Specific Reach MSE**: A loss is defined per cluster and the back propagation is per cluster. The loss for $i$-th cluster is,

$$\mathcal{L}_R = \sum_j A_{ij}\left(\rho_{ij}\eta_{ij} - B/N\chi_j\right)^2 \tag{3}$$

**Cluster Agnostic Reach MSE**: Here the loss is averaged across all clusters, and the back propagation is across all clusters. The MSE is,

$$\mathcal{L}_R = \frac{\sum_i \sum_j A_{ij} n_i * \left(\rho_{s_{ij}}\eta_{s_{ij}} - B/N\chi_j\right)^2}{\sum_i n_i} \tag{4}$$

## 5.3 Budget Constraint

Let $j' \triangleq \arg\max_j (A_{ij}\rho_{ij}\eta_{ij})$ be the channel which maximizes $R_i$ for any given segment $i$. The budget constraint for total reach is

$$T_R = B - \sum_i (A_{ij'}\rho_{ij'}\eta_{ij'})\chi_{j'}n_i \geq 0 \tag{5}$$

where the second term in equation (5) is the **Spend**.

# 6 TRAINING

Pseudo-code of the model is given in Algorithm 2, and the joint optimization in Algorithm 3. Top left of Figure 2 mentions the ordered sequence of steps needed for initialization and training.

Pre-training and Initialization consists of doing the following (Algorithm 1).

(1) Pre-train the Encoder and the Predictor using the loss

$$\mathcal{L}_1(\theta, \phi) = \mathbb{E}_{\mathbf{x}, y\sim p_{XY}}\left[-\sum_{t\in\mathcal{T}} l_1(y_t, \hat{y}_t)\right] \tag{6}$$

where $\hat{y}_t = g_\phi(f_\theta(\mathbf{x}_t))$ is the predicted conversion probability of a user and $l_1(y_t, \hat{y}_t) = -\sum_{c\in\{0,1\}} y_t^c \log(\hat{y}_t^c)$.

(2) Initialize the cluster embeddings using K-means on the representations $z_t^{(n)}$ for all $n$ users and for all $t$ that are obtained after pre-training the encoder.

(3) Pre-train the Selector on all $z_t^{(n)}$ and corresponding cluster assignments obtained from K-means.

Subsequently, we train as follows. Lines 1-8 in Algorithm 2 use an alternating minimization approach to alternate between training an Actor-Critic network and updating the embedding dictionary. Here the Actor is the (Encoder, Selector) pair of networks and the Critic is the Predictor network. Lines 9-13 in Algorithm 2 train the Beh2Stat network. Finally, lines 14-19 use alternating minimization to alternate between maximizing the reach and updating the embedding dictionary.

The actor's loss is $\mathcal{L}_A(\theta, \phi, \psi) = \mathcal{L}_1(\theta, \phi, \psi) + \alpha\mathcal{L}_2(\theta, \phi)$ which combines two losses with $\alpha$ as the hyperparameter. The loss term $\mathcal{L}_2(\theta, \psi)$ promotes sparse cluster assignment such that each user belonging to only one cluster with high probability. It is given by

$$\mathcal{L}_2(\theta, \psi) = \mathbb{E}_{\mathbf{x}\sim p_X}\left[-\sum_{t\in T}\sum_{k\in K} \pi_t(k)\log\pi_t(k)\right] \tag{7}$$

The loss term $\mathcal{L}_1(\theta, \phi, \psi)$ promotes the prediction of cluster level outcomes $\bar{y}_t$ from the cluster centroid. It is the partial function obtained by fixing the embedding $\mathcal{E}$ is the loss expression $\mathcal{L}_1(\theta, \phi, \psi, \mathcal{E})$ given by

$$\mathcal{L}_1(\theta, \phi, \psi, \mathcal{E}) = \mathbb{E}_{\mathbf{x}, y\sim p_{XY}}\left[\sum_{t\in T} \mathbb{E}_{c_t\sim Cat(\pi_t)}\left[l_1(y_t, \bar{y}_t)\right]\right] \tag{8}$$

The critic's loss is set to $\mathcal{L}_C(\phi) = \mathcal{L}_1(\theta, \phi, \psi)$. Further, to promote well-separated cluster centroids in the embedding dictionary representation, the loss $\mathcal{L}_E(\mathcal{E}) = \mathcal{L}_1(\mathcal{E}) + \beta\mathcal{L}_3(\mathcal{E})$ is used to update the embedding dictionary, where

$$\mathcal{L}_3(\mathcal{E}) = -\sum_{k\neq k'} l_1(g_\phi(\mathbf{e}(k), g_\phi(\mathbf{e}(k')) \tag{9}$$

For the reach maximization subject to budget constraints, we use the known techniques of Barrier method and Augmented Lagrangian from constrained deterministic optimization [14] to convert it to an equivalent unconstrained objective for Algorithm 3. These methods are fairly well understood for convex optimization. However, neural network training is a non-convex problem and in this setting, the methods employing Barriers or Augmented Lagrangians are not as well understood, despite being intuitive. The three formulations that we implement are

(1) *Slack Minimization* with loss

$$\mathcal{L}_R = \frac{1}{\ln(R)} + \frac{max(T_R, 0)}{w} \tag{10}$$

The dual variable update rule is $w \leftarrow \mu * w$ with initialization $\mu \leftarrow 0.3$.

(2) *Barrier Method* with loss

$$\mathcal{L}_R = \frac{1}{\ln(R)} - \frac{\log(-T_R)}{w} \tag{11}$$

and the same dual update rule and initialization as above.

(3) *Augmented Lagrangian Method* with loss

$$\mathcal{L}_R = \frac{1}{\ln(R)} - \frac{\lambda T_R}{B} + \frac{\mu}{2}\left(\frac{T_R}{B}\right)^2 \tag{12}$$

The update rule is $\lambda_k \leftarrow max(\lambda_{k-1} + \mu * \frac{T_{R,k-1}}{B}, 0)$, where $k$ is iteration. Note that $\lambda \geq 0$ and we use initialization $\mu \leftarrow 0.1$, $\lambda \leftarrow 0.1$.

## 7 IMPLEMENTATION DETAILS

Our experimental setup including network parameters and hyperparameters used are as follows. The encoder is a HAN, with the dimension of the hidden layer being 50. The predictor is a Multi-Layered Perceptron (MLP), with input $z_t$ (of dimension 50), two hidden layer with 50 perceptrons and an output layer of size 1. A dropout rate of 0.3 is used after the hidden layer. Hidden layers have ReLU activation and the output layer has sigmoid activation. Selector is also an MLP with input $z_t$, followed by hidden layer with 50 perceptrons, which uses ReLU activation and a dropout 0.3. The output layer is of size $K$, with softmax activation. The Encoder-Predictor, Selector, Actor and Critic are trained using Adam Optimizer with a learning rate of 0.001 on the aforementioned loss functions. The network weights and biases are initialized using 'glorot uniform'. Batch size used is 128. Initialization iterations for pretraining the Encoder-Predictor is 1000 and that for pretraining the selector is 5000. Number of iterations for training actor, critic and embeddings is 1000, with an early stopping after 100 epochs, based on minimum value of $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_3$ obtained on validation data within previous 15 iterations. Beh2Stat is an MLP with input $z_t$ (of dimension 50), four hidden layer with 500 perceptrons and the output layer uses Softmax activation to yield probabilities over the static features. It is trained for the same number of iterations as actor-critic, using Adam Optimizer with a learning rate of 0.005. The joint optimization model (Algorithm 3) is trained to update the parameters of Selector, Predictor and MLP $v_\delta(\mathcal{S})$, with the input as static features and output as mediums. It is trained for 2000 iterations, with an early stopping after 100 epochs, based on minimum value of $\mathcal{L}_4$ obtained on validation data within previous 15 iterations. Depicting training and validation convergence plots for Augmented Lagrangian Method on Dataset II, Figure 3 shows good convergence for the four losses.

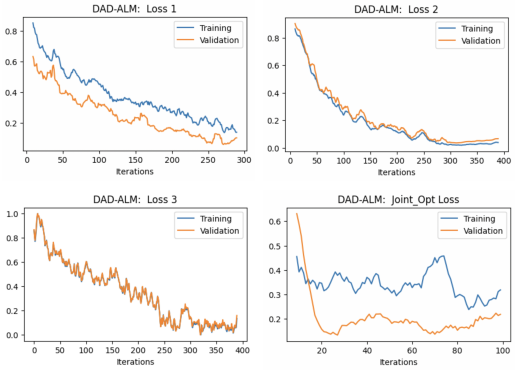


**Figure 3: Convergence plots for Losses $\mathcal{L}_1$, $\mathcal{L}_2$, $\mathcal{L}_3$ and $\mathcal{L}_4$. DAD-ALM, Dataset II. Training - Blue, Validation - Orange.**

## 8 EXPERIMENTS

We design experiments around joint optimization of two goals: (i) strong predictive conversion performance for behavior segments; and (ii) achieve high reach relative to spend, subject to budget constraint. In direct media spend, one can think of reach as corresponding to pay-per-click, used by businesses to pay for messages they send to the targeted segments. Our experiments test these goals and are described next.

### 8.1 Experimental Setup

Two SOTA baselines (**DISC**) plus five proposed **D**elivery-**A**ware-**D**iscovery (**DAD**) models are now presented.

*8.1.1 Baselines.*

- **DISC-UC**: The first baseline, based on behavioral segmentation objective in Section 5.1, is the **disc**overy focused SOTA baseline, not delivery-optimized, since delivery-optimized-discovery is not available in the prior art. Here, the spend is unconstrained (**UC**) by the budget.
- **DISC-BC**: The second baseline, is the **disc**overy focused SOTA baseline, not delivery-optimized, but the spend is budget constrained (**BC**). We train Step 1 and Step 2 of the network to obtain $K$ segments, and skip Step 3 reach maximization. We compute reach for each segment, for each medium, and assign each segment to a medium giving the highest reach value for it. This process is repeated until no budget is left over.

*8.1.2 Proposed Models.*

- **DAD-CSSE**: Cluster Specific MSE, equation (3)
- **DAD-CASE**: Cluster Agnostic MSE, equation (4)
- **DAD-SMIN**: Slack Minimization, equation (10)
- **DAD-BARR**: Barrier, equation (11)
- **DAD-ALM**: Augmented Lagrangian, equation (12)

*8.1.3 Performance Metrics.* We evaluate against the ground truth of outcome, conversion, and spend efficiency to increase reach. We introduce two new metrics which combine conversion accuracy, spend, and budget to give new insights about efficiency and effectiveness of discovery and delivery.

**AUROC.** Consistent with the first goal of strong predictive conversion for behavior segments, the performance metric is $AUROC_{\overline{y}}$, for which *higher values are better.*

**Spend per Unit Reach.** The spend metric comparable across models including baselines is the spend per unit reach. Since delivery-optimized-delivery is not available in the two DISC baselines an overall spend metric is unfair to baselines. However, spend per unit reach can be used since both reach and spend are affected in a comparable way for each model. Here, *lower values are better.*

**Within % Budget.** This metric expresses the optimal spend relative to the budget and applies to all five proposed DAD models. Since the DISC-UC baseline's delivery has no budget constraint, the metric is undefined for DISC-UC. The DISC-BC baseline is budget constrained and this metric applies. Here, a *tighter interval around zero is better* since it indicates closeness to the budget.

**Effective Spend.** In a new contribution, this metric, capturing 'effectiveness of spend,' combines the two metrics $AUROC_{\overline{y}}$ and

**Algorithm 1** Pre-training

**Input:** Dataset $\mathcal{D} = \{(\mathbf{x}_t^n, y_t^n)_{t=1}^{\tau^n}\}_{n=1}^N$, number of clusters $K$, learning rate $\eta$, mini-batch size $n_{mb}$
**Output:** Model parameters $\{\theta, \psi, \phi\}$, initialized by Glorot-Uniform, embedding dictionary $\mathcal{E}$.

*Pre-train Encoder-Predictor*
**repeat**
  Sample mini-batch of $n_{mb}$ samples
  **for** n = 1 to $n_{mb}$ **do**
    Calculate $\hat{y}_t^n \leftarrow g_\phi(f_\theta(\mathbf{x}_{1:t}^n))$
  $\theta \leftarrow \theta - \eta \frac{1}{n_{mb}} \sum_{n=1}^{n_{mb}} \sum_{t=1}^{T^n} \nabla_\theta l_1(y_t^n, \hat{y}_t^n),\ \phi \leftarrow \phi - \eta \frac{1}{n_{mb}} \sum_{n=1}^{n_{mb}} \nabla_\phi l_1(y_t^n, \hat{y}_t^n)$
**until** convergence
Calculate embeddings dictionary $\mathcal{E}$ and initial cluster assignments $c_t^n$
$\mathcal{E}, \{\{c_t^n\}_{t=1}^{\tau^n}\}_{n=1}^N \leftarrow$ `K-Means`$(\{\{z_t^n\}_{t=1}^{n_{mb}}\}_{n=1}^N)$, where $z_t^n = f_\theta(\mathbf{x}_{1:t})$
*Pre-train the Selector*
**repeat**
  Sample a minimbatch $n_{mb}$ of data samples
  **for** $n = 1 \ldots n_{mb}$ **do**
    Calculate cluster assignment probability $\pi_t^n \leftarrow h_\psi(f_\theta(\mathbf{x}_{1:t})$
  Update selector parameters, $\psi \leftarrow \psi + \eta \frac{1}{n_{mb}} \sum_{n=1}^{n_{mb}} \sum_{t=1}^{\tau^n} \sum_{k=1}^{K} c_t^n(k) \log \pi_t^n(k)$
**until** convergence

**Algorithm 2** Pseudo-code

*Pre-train Encoder-Predictor*
*Initialize Cluster Centroid Embeddings*
*Pre-train Selector* ▷ Run Algorithm 1
**repeat**
  Sample mini-batch of $n_{mb}$ samples
  **for** n = 1 to $n_{mb}$ **do**
    Train Actor $\leftarrow \min \mathcal{L}_A(\theta, \phi, \psi)$
    Train Critic $\leftarrow \min \mathcal{L}_C(\phi)$
  **for** n = 1 to $n_{mb}$ **do**
    Update embedding dictionary $\leftarrow \min \mathcal{L}_E(\mathcal{E})$
**until** convergence
**repeat**
  Sample mini-batch of $n_{mb}$ samples
  **for** n = 1 to $n_{mb}$ **do**
    Train Beh2Stat $\leftarrow \min \mathcal{L}_B(\omega)$
**until** convergence
**repeat**
  Sample mini-batch of $n_{mb}$ samples
  **for** n = 1 to $n_{mb}$ **do**
    Train Joint_Opt $\leftarrow \min \mathcal{L}_4(\theta, \phi, \psi, \omega, \delta)$
    Update embedding dictionary $\leftarrow \min \mathcal{L}_E(\mathcal{E})$
**until** convergence

**Algorithm 3** Joint_Opt training

**for** each mini-batch in $\{(x_t^n, y_t^n)_{t=1}^{\tau^n}\}_{n=1}^{n_{mb}} \sim \mathcal{D}$ **do**
  $z_t^n \leftarrow f_\theta(\mathbf{x}_{1:t}^n)$
  $\pi_t^n \leftarrow h_\psi(z_t^n)$
  sample cluster assignment $c_t^n \sim Cat(\pi_t^n))$
  centroid embedding $\bar{z}_t^n := \mathcal{E}(c_t^n)$
  $\bar{y}_t^n \leftarrow g_\phi(\bar{z}_t^n)$
  $\mathcal{S} \leftarrow b_\omega(\bar{z}_t^n)$
  $A_{ij} \leftarrow v_\delta(\mathcal{S})$
  Compute reach, $R$ using (2)
  Compute constraint, $T$ using (5)
  Update $h_\psi$, $g_\phi$ and $v_\delta$, minimize $\mathcal{L}_4(\theta, \phi, \psi, \omega, \delta)$

*Spend per Unit Reach*, used respectively for the two goals of high predictive accuracy and low spend per unit reach (Section 8), into a composite scalar metric. By combining two metrics, a scalar metric represents the combined goal. We define *Effective Spend = Spend per Unit Reach / $AUROC_{\overline{y}}$*. For the numerator, lower is better, while for the denominator, higher is better. Put together, for Effective Spend, *lower values are better.*

**Reach Efficiency-Effectiveness, or, Reach Effc-Effe.** In contributing another new metric, this is the second composite, scalar metric which captures both the 'efficiency' and 'effectiveness' of reach. Note that we seek to maximize reach, subject to a budget. We define, *Reach Effc-Effe* = (*Reach* / *Spend as Proportion of Budget*) * ($AUROC_{\overline{y}}$). The term in the first parenthesis represents reach-efficiency, making reach-efficiency increase (decrease) when the denominator is less (more) than 1. The term in the second parenthesis stands for accuracy of prediction, and thus combined with the first term, provide a succinct representation of both the efficiency of budget spend to maximize reach, and the effectiveness in achieving the other goal of high performance in predictive accuracy. The DISC-UC baseline's delivery has no budget constraint; this metric is undefined for DISC-UC. Here, *higher values are better.*

## 8.2 Results, Dataset I

All results are shown with **Mean +/- Stderr** based on 8 seeds. Caption of Table 2 highlights the significantly improved performance of proposed `DAD` models over SOTA baselines `DISC`s. Note that

two metrics are undefined for DISC-UC since it is unconstrained. Overall, DAD-SMIN and DAD-ALM, each with good performance in 3 metrics, including two important composite metrics *Effective Spend* and *Reach-Effc-Effe*, stand out. Thus, decreased spend for maximizing reach is achievable with no decrease in conversion performance.

**Table 2: Results. Dataset I. K=5. In $AUROC_{\overline{y}}$ (higher is better), five proposed DAD models perform identical to the two DISC baselines. In *Within % Budget* (closer to 0 is better), baseline DISC-BC outperforms DAD models. In *Spend per unit Reach* (lower is better), DAD-SMIN, DAD-ALM outperforms DISC baselines. In *Effective Spend* (lower is better), DAD-SMIN, DAD-CSSE, DAD-CASE, DAD-ALM outperform DISC baselines. In *Reach Effc-Effe* (higher is better), DAD-SMIN, DAD-CSSE, DAD-CASE, DAD-ALM outperform DISC-BC baseline. Overall, the DAD models show strong and significantly improved performance (shown in bold) over DISC baselines. See Section 8.2 for discussion.**

| Model | AUROC $\overline{y}$ | Within % Budget | Spend per unit Reach | Effective Spend | Reach Effc-Effe |
|---|---|---|---|---|---|
| DISC-UC | 0.873 | – | 66.993 | 76.751 | – |
| | ±0.002 | – | ±1.42 | ±1.722 | – |
| DISC-BC | 0.873 | **-0.304** | 68.105 | 78.017 | 87.383 |
| | ±0.002 | ±0.149 | ±1.127 | ±1.361 | ±1.596 |
| DAD-CSSE | 0.88 | -18.301 | 60.02 | **68.181** | 100.173 |
| | ±0.009 | ±5.988 | ±1.553 | ±1.631 | ±2.294 |
| DAD-CASE | 0.873 | -13.745 | 57.618 | **65.984** | **103.226** |
| | ±0.004 | ±8.163 | ±0.833 | ±0.96 | ±1.492 |
| DAD-SMIN | 0.873 | -10.474 | **59.827** | **68.62** | **99.951** |
| | ±0.009 | ±5.252 | ±1.709 | ±2.26 | ±3.25 |
| DAD-BARR | 0.841 | -9.122 | 61.751 | 73.539 | 93.535 |
| | ±0.014 | ±16.786 | ±2.275 | ±2.873 | ±3.426 |
| DAD-ALM | 0.875 | -12.002 | **57.403** | **65.693** | **103.74** |
| | ±0.009 | ±4.775 | ±0.737 | ±1.083 | ±1.737 |

## 8.3 Results, Dataset II, Public

Throughout all tables, results are shown with **Mean +/- Stderr** based on 8 seeds. Caption of Table 3 highlights the significantly improved performance of DAD models over SOTA baselines, DISCs. Note that two metrics are undefined for DISC-UC. Overall, DAD-ALM with strong performance in four metrics stand out. Similar to Dataset I, results with Public, Dataset II strongly reinforce that decreased spend for maximizing reach is achievable with no depreciation in performance in conversion prediction. That is, delivery aware discovery outweighs traditional discovery which ignore delivery and budget constraint. We now move on to sensitivity experiments by varying $K$.

## 8.4 Sensitivity to $K$

Sensitivity to the choice of $K$, number of segments, with Dataset II, public data, is shown in Table 4, for K=7 and 9. DAD-BARR and DAD-ALM strongly outperform baselines in crucial composite metrics *Effective Spend* and *Reach-Effc-Effe*, and perform the same in predicting conversion, $AUROC_{\overline{y}}$. Comparing with Table 3 finds remarkable consistency across K=5, 7, 9, strongly favoring our proposed models over SOTA baselines.

**Table 3: Results. Dataset II. K=5. In $AUROC_{\overline{y}}$ (higher is better), five proposed DAD models perform identical to two baselines DISCs. In *Within % Budget* (closer to 0 is better), DAD-ALM performs same as baseline DISC-BC. In *Spend per unit Reach* (lower is better), DAD-ALM outperforms DISC baselines. In *Effective Spend* (lower is better), DAD-SMIN, DAD-ALM outperform DISC baselines. In *Reach Effc-Effe* (higher is better), DAD-SMIN, DAD-ALM outperform DISC-BC baseline. Overall, the DAD models show strong and significantly improved performance (shown in bold) over baselines DISC. See Section 8.3 for discussion.**

| Model | AUROC $\overline{y}$ | Within % Budget | Spend per unit Reach | Effective Spend | Reach Effc-Effe |
|---|---|---|---|---|---|
| DISC-UC | 0.956 | – | 61.543 | 64.333 | – |
| | ±0.002 | – | ±1.803 | ±1.84 | – |
| DISC-BC | 0.956 | **-6.43** | 62.736 | 65.594 | 1640.449 |
| | ±0.002 | ±2.246 | ±0.824 | ±0.898 | ±22.355 |
| DAD-CSSE | 0.961 | -24.498 | 63.471 | 66.038 | 1643.048 |
| | ±0.001 | ±6.616 | ±2.189 | ±2.304 | ±58.492 |
| DAD-CASE | 0.963 | -25.344 | 62.285 | 64.7 | 1671.964 |
| | ±0.001 | ±9.345 | ±1.826 | ±1.919 | ±48.866 |
| DAD-SMIN | 0.957 | 5.237 | 59.229 | **61.884** | **1745.79** |
| | ±0.002 | ±4.846 | ±1.629 | ±1.707 | ±44.765 |
| DAD-BARR | 0.945 | -27.171 | 62.667 | 66.369 | 1623.459 |
| | ±0.006 | ±4.207 | ±1.009 | ±1.309 | ±30.659 |
| DAD-ALM | 0.957 | **-3.485** | **58.548** | **61.196** | **1757.545** |
| | ±0.001 | ±3.955 | ±0.749 | ±0.738 | ±21.582 |

## 8.5 Ablation Study

The ablation study checks whether the proposed full model, or, the full network architecture shown in Figure 2 is necessary to achieve our dual goals of strong predictive performance in conversion and reach maximization, subject to budget constraint. Or, can a reduced architecture give comparable performance? To test this, we *turn off* Step 2 of the architecture in training, that is, did not train Actor(Encoder, Selector)-Critic(Predictor) and centroid Embeddings (see green text in Figure 2). We trained Steps 1 and 3 only. The ablated results, with Dataset II, are shown in Table 5 for each $K$ = 5, 7, 9. We compare the full architecture's performance with that of the reduced architecture, for each $K$, affording generalization of the ablation. As the caption in Table 5 states, the ablated results show large decrease in performance compared to the full architecture, for each $K$ = 5, 7, 9. This strongly justifies the use of the proposed full architecture for the delivery aware discovery problem.

## 8.6 Evaluation of Beh2Stat

For the proprietary data, Table 6 shows that Beh2Stat has accuracies appreciably higher than that of the random model. Random prediction accuracy values are shown in the last column, which vary by the number of classes.

# 9 CONCLUSION

Behavioral segmentation of users is highly desirable by a firm as emphasized in both industry and academia [7, 20] since users' online behaviors are very predictive of outcomes such as conversion. Firms send different offers, messages, communications to different

**Table 4: Sensitivity to K. Dataset II. See Section 8.4. Similar to K=5 results, DAD models, namely DAD-BARR and DAD-ALM stand out in strong performance in 3 metrics as compared to baselines DISCs.**

| Model | AUROC $\overline{y}$ | Within % Budget | Spend per unit Reach | Effective Spend | Reach Effc-Effe |
|---|---|---|---|---|---|
| K=7 | | | | | |
| DISC-UC | 0.955 | – | 60.605 | 63.437 | – |
| | ±0.002 | – | ±2.26 | ±2.358 | – |
| DISC-BC | 0.955 | **-3.787** | 63.113 | 66.068 | 1627.736 |
| | ±0.002 | ±1.55 | ±0.633 | ±0.701 | ±17.448 |
| DAD-CSSE | 0.961 | -19.716 | 62.695 | 65.234 | 1661.574 |
| | ±0 | ±7.19 | ±2.089 | ±2.163 | ±55.77 |
| DAD-CASE | 0.962 | -24.853 | 61.416 | 63.842 | 1698.022 |
| | ±0.001 | ±3.967 | ±2.066 | ±2.167 | ±56.525 |
| DAD-SMIN | 0.959 | 6.298 | 61.386 | 63.984 | 1691.268 |
| | ±0.001 | ±5.006 | ±1.949 | ±2.009 | ±49.046 |
| DAD-BARR | 0.947 | -43.356 | **60.354** | **63.794** | **1693.691** |
| | ±0.004 | ±3.921 | ±1.486 | ±1.781 | ±43.628 |
| DAD-ALM | 0.962 | -13.002 | **57.89** | **60.207** | **1787.501** |
| | ±0.001 | ±3.842 | ±0.846 | ±0.901 | ±26.752 |
| K=9 | | | | | |
| DISC-UC | 0.959 | – | 59.482 | 61.99 | – |
| | ±0.001 | – | ±1.596 | ±1.645 | – |
| DISC-BC | 0.959 | **-4.024** | 63.656 | 66.346 | 1621.532 |
| | ±0.001 | ±1.629 | ±0.772 | ±0.837 | ±20.866 |
| DAD-CSSE | 0.962 | -29.568 | 58.923 | **61.272** | **1757.175** |
| | ±0.001 | ±4.949 | ±0.958 | ±1.013 | ±29.456 |
| DAD-CASE | 0.963 | -28.462 | 59.628 | **61.897** | **1744.099** |
| | ±0.001 | ±6.72 | ±1.563 | ±1.592 | ±41.464 |
| DAD-SMIN | 0.957 | 1.845 | 62.286 | 65.079 | 1671.734 |
| | ±0.002 | ±6.86 | ±2.438 | ±2.582 | ±66.204 |
| DAD-BARR | 0.955 | -40.699 | **60.307** | **63.168** | **1703.421** |
| | ±0.001 | ±5.292 | ±0.846 | ±0.915 | ±23.898 |
| DAD-ALM | 0.962 | **-3.617** | **58.102** | **60.405** | **1781.633** |
| | ±0.001 | ±4.486 | ±0.887 | ±0.908 | ±26.626 |

segments. The obstacle in delivering messages on media channels lies in reaching these *behavioral* segments on media. On a medium, only a proportion of a behavior segment can be matched, and of those matched only a fraction sees / clicks on a message. Moreover, on a medium, users are defined by their static characteristics, but not by their online interactions with the firm, which only the firm knows, but the media do not. For discovery to be useful, delivery ought to be considered simultaneously, and not sequentially after discovery. Extant work on discovery of user segments ignore the need for this simultaneity between discovery and delivery. We offer an approach to fill this research gap. Extensive experiments on two datasets - proprietary and public (Google Analytics) - find strong support for our approach. Moreover, sensitivity experiments on Google data, by varying the hyperparameter, number of segments, affirm those findings that our approach achieves high, improved performance on Spend and Reach metrics, while achieving equally good predictive conversion performance AUROC, compared to two SOTA baselines. Ablation studies, including sensitivity of ablation to number of segments, further justify the value of our proposed network to address this joint delivery-aware-discovery optimization problem at hand.

**Table 5: Ablation Sensitivity to K. Dataset II. See Section 8.5. For each K, comparing results of the full model (see Tables 3 and 4), the ablated results below show that: Each DAD model has appreciably large (i) decrease in $AUROC_{\overline{y}}$, (ii) increase in *Effective Spend*, and (iii) decrease in *Reach Effc-Effe*.**

| Model | AUROC $\overline{y}$ | Within % Budget | Spend per unit Reach | Effective Spend | Reach Effc-Effe |
|---|---|---|---|---|---|
| K=5 | | | | | |
| DAD-CSSE | 0.898 | -31.965 | 61.615 | 68.606 | 1575.407 |
| | ±0.004 | ±3.561 | ±1.672 | ±1.945 | ±42.224 |
| DAD-CASE | 0.898 | -28.608 | 62.268 | 69.374 | 1561.016 |
| | ±0.004 | ±5.573 | ±1.984 | ±2.196 | ±49.403 |
| DAD-SMIN | 0.866 | **2.948** | 60.238 | 69.716 | 1546.403 |
| | ±0.015 | ±1.597 | ±1.126 | ±1.479 | ±32.301 |
| DAD-BARR | 0.857 | -41.1 | **58.436** | 68.221 | 1577.539 |
| | ±0.008 | ±4.475 | ±0.794 | ±1.01 | ±23.999 |
| DAD-ALM | 0.898 | -11.66 | **58.997** | **65.7** | **1641.753** |
| | ±0.001 | ±2.53 | ±1.352 | ±1.546 | ±35.428 |
| K=7 | | | | | |
| DAD-CSSE | 0.9 | -26.207 | 59.284 | 65.847 | 1637.582 |
| | ±0.003 | ±4.598 | ±1.357 | ±1.429 | ±35.284 |
| DAD-CASE | 0.9 | -15.201 | 61.319 | **68.114** | **1582.834** |
| | ±0.003 | ±4.831 | ±1.258 | ±1.425 | ±33.951 |
| DAD-SMIN | 0.877 | **-2.069** | **61.442** | **70.08** | 1539.038 |
| | ±0.005 | ±2.513 | ±1.479 | ±1.571 | ±34.241 |
| DAD-BARR | 0.876 | **-36.016** | **62.338** | **71.203** | 1520.433 |
| | ±0.005 | ±4.633 | ±1.776 | ±2.261 | ±46.257 |
| DAD-ALM | 0.891 | **-18.332** | **58.468** | **65.602** | **1639.33** |
| | ±0.001 | ±5.067 | ±0.69 | ±0.758 | ±19.304 |
| K=9 | | | | | |
| DAD-CSSE | 0.896 | -27.864 | 58.868 | 65.673 | 1637.147 |
| | ±0.003 | ±2.485 | ±0.571 | ±0.658 | ±16.876 |
| DAD-CASE | 0.894 | -26.952 | 59.551 | 66.605 | 1620.766 |
| | ±0.003 | ±6.121 | ±1.517 | ±1.725 | ±38.121 |
| DAD-SMIN | 0.819 | **11.684** | **62.404** | **76.973** | 1416.837 |
| | ±0.03 | ±6.11 | ±1.947 | ±3.41 | ±60.135 |
| DAD-BARR | 0.859 | **-36.784** | **61.994** | **72.136** | 1498.258 |
| | ±0.003 | ±3.793 | ±1.736 | ±2.005 | ±40.722 |
| DAD-ALM | 0.896 | **-12.456** | **58.753** | **65.592** | **1639.492** |
| | ±0.003 | ±3.386 | ±0.74 | ±0.74 | ±18.771 |

**Table 6: Evaluation of Beh2Stat function. See section 8.6.**

| | | Accuracy | |
|---|---|---|---|
| **Static Characteristics** | **# Classes** | **Beh2Stat** | **Random** |
| *Country* | 6 | 0.58 | 0.17 |
| *Source* | 2 | 0.86 | 0.5 |
| *Member* | 3 | 0.53 | 0.33 |
| *Browser* | 6 | 0.51 | 0.17 |
| *Operating System (OS)* | 6 | 0.51 | 0.17 |

As a way of limitation, our joint optimization works when the cost of a message sent to a user is known and not an outcome of bidding. Future research may address this problem, under ad bidding based media spend and budget pacing.

# REFERENCES


[1] 2017. https://www.kaggle.com/bigquery/google-analytics-sample
[2] 2022. https://www.statista.com/statistics/276671/global-internet-advertising-expenditure-by-type/
[3] 2022. https://www.facebook.com/help/274400362581037
[4] 2022. https://advertising.amazon.com/library/guides/media-buying
[5] 2022. https://www.statista.com/statistics/1106617/number-of-pages-seen-buying-session
[6] 2023. https://www.investopedia.com/terms/m/marketsegmentation.asp
[7] 2023. https://blog.hubspot.com/marketing/behavioral-segmentation
[8] Deepak Agarwal, Souvik Ghosh, Kai Wei, and Siyu You. 2014. Budget pacing for targeted online advertisements at linkedin. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1613–1619.
[9] Saeed Reza Aghabozorgi, Ali Seyed Shirkhorshidi, and Ying Wah Teh. 2015. Time-series clustering - A decade review. *Inf. Syst.* 53 (2015), 16–38. https://doi.org/10.1016/j.is.2015.04.007
[10] Elie Aljalbout, Vladimir Golkov, Yawar Siddiqui, Maximilian Strobel, and Daniel Cremers. 2018. Clustering with deep learning: Taxonomy and new methods. *arXiv preprint arXiv:1801.07648* (2018).
[11] Maha Alkhayrat, Mohamad Aljnidi, and Kadan Aljoumaa. 2020. A comparative dimensionality reduction study in telecom customer segmentation using deep learning and PCA. *Journal of Big Data* 7, 1 (2020), 1–23.
[12] Santiago Balseiro, Anthony Kim, Mohammad Mahdian, and Vahab Mirrokni. 2017. Budget management strategies in repeated auctions. In *Proceedings of the 26th International Conference on World Wide Web*. 15–23.
[13] Santiago Balseiro, Haihao Lu, and Vahab Mirrokni. 2020. Dual mirror descent for online allocation problems. In *International Conference on Machine Learning*. PMLR, 613–628.
[14] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. 2004. *Convex optimization*. Cambridge university press.
[15] Xiaojun Chen, Yixiang Fang, Min Yang, Feiping Nie, Zhou Zhao, and Joshua Zhexue Huang. 2017. Purtreeclust: A clustering algorithm for customer segmentation from massive customer transaction data. *IEEE Transactions on Knowledge and Data Engineering* 30, 3 (2017), 559–572.
[16] Vincent Conitzer, Christian Kroer, Eric Sodomka, and Nicolas E Stier-Moses. 2022. Multiplicative pacing equilibria in auction markets. *Operations Research* 70, 2 (2022), 963–989.
[17] Chinedu Pascal Ezenkwu, Simeon Ozuomba, and Constance Kalu. 2015. Application of K-Means algorithm for efficient customer segmentation: a strategy for targeted customer services. (2015).
[18] Jon Feldman, Shanmugavelayutham Muthukrishnan, Martin Pal, and Cliff Stein. 2007. Budget optimization in search-based advertising auctions. In *Proceedings of the 8th ACM conference on Electronic commerce*. 40–49.
[19] Zahra Ghasemi, Hadi Akbarzadeh Khorshidi, and Uwe Aickelin. 2021. A survey on Optimisation-based Semi-supervised Clustering Methods. In *2021 IEEE International Conference on Big Knowledge (ICBK)*. IEEE, 477–482.
[20] Sunil Gupta. 2014. Marketing reading: Segmentation and targeting. *Core Curriculum Readings Series. Boston: Harvard Business Publishing* 8219 (2014).
[21] K Kameshwaran and K Malarvizhi. 2014. Survey on clustering techniques in data mining. *International Journal of Computer Science and Information Technologies* 5, 2 (2014), 2272–2276.
[22] Chinmay Karande, Aranyak Mehta, and Ramakrishnan Srikant. 2013. Optimizing budget constrained spend in search advertising. In *Proceedings of the sixth ACM international conference on Web search and data mining*. 697–706.
[23] Bhuvesh Kumar, Jamie Morgenstern, and Okke Schrijvers. 2022. Optimal Spend Rate Estimation and Pacing for Ad Campaigns with Budgets. *arXiv preprint arXiv:2202.05881* (2022).
[24] Changhee Lee and Mihaela Van Der Schaar. 2020. Temporal phenotyping using deep predictive clustering of disease progression. In *International Conference on Machine Learning*. PMLR, 5767–5777.
[25] Duc Thanh Anh Luong and Varun Chandola. 2017. A K-Means Approach to Clustering Disease Progressions. In *2017 IEEE International Conference on Healthcare Informatics, ICHI 2017, Park City, UT, USA, August 23-26, 2017*. IEEE Computer Society, 268–274. https://doi.org/10.1109/ICHI.2017.18
[26] Naveen Sai Madiraju, Seid M. Sadat, Dimitry Fisher, and Homa Karimabadi. 2018. Deep Temporal Clustering : Fully Unsupervised Learning of Time-Domain Features. *CoRR* abs/1802.01059 (2018). arXiv:1802.01059 http://arxiv.org/abs/1802.01059
[27] Chotirat Ratanamahatana, Eamonn Keogh, Anthony J Bagnall, and Stefano Lonardi. 2005. A novel bit level time series representation with implication of similarity search and clustering. In *Pacific-Asia conference on knowledge discovery and data mining*. Springer, 771–777.
[28] Alexander Rusanov, Patric V. Prado, and Chunhua Weng. 2016. Unsupervised Time-Series Clustering Over Lab Data for Automatic Identification of Uncontrolled Diabetes. In *2016 IEEE International Conference on Healthcare Informatics, ICHI 2016, Chicago, IL, USA, October 4-7, 2016*. IEEE Computer Society, 72–80. https://doi.org/10.1109/ICHI.2016.14
[29] Jaehyeok Shin, Alessandro Rinaldo, and Larry Wasserman. 2019. Predictive clustering. *arXiv preprint arXiv:1903.08125* (2019).
[30] Nicolas E Stier-Moses. 2018. Multiplicative Pacing Equilibria in Auction Markets. In *Web and Internet Economics: 14th International Conference, WINE 2018, Oxford, UK, December 15–17, 2018, Proceedings*, Vol. 11316. Springer, 443.
[31] Gang Wang, Xinyi Zhang, Shiliang Tang, Haitao Zheng, and Ben Y Zhao. 2016. Unsupervised clickstream clustering for user behavior analysis. In *Proceedings of the 2016 CHI conference on human factors in computing systems*. 225–236.
[32] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association* 58, 301 (1963), 236–244.
[33] Junyuan Xie, Ross B. Girshick, and Ali Farhadi. 2015. Unsupervised Deep Embedding for Clustering Analysis. *CoRR* abs/1511.06335 (2015). arXiv:1511.06335 http://arxiv.org/abs/1511.06335
[34] Jian Xu, Kuang-chih Lee, Wentong Li, Hang Qi, and Quan Lu. 2015. Smart pacing for effective online ad campaign optimization. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2217–2226.
[35] Bo Yang, Xiao Fu, Nicholas D. Sidiropoulos, and Mingyi Hong. 2016. Towards K-means-friendly Spaces: Simultaneous Deep Learning and Clustering. *CoRR* abs/1610.04794 (2016). arXiv:1610.04794 http://arxiv.org/abs/1610.04794
[36] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 1480–1489. https://doi.org/10.18653/v1/N16-1174