AI and Experimentation

Learning more from limited human data

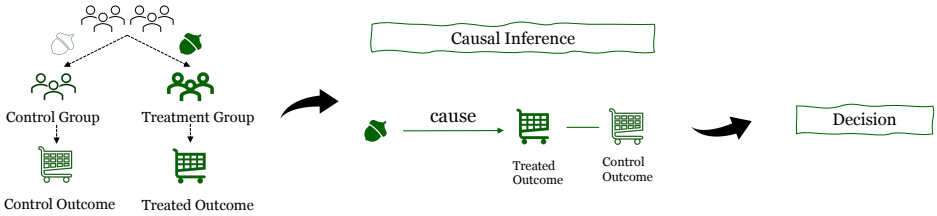
Dennis Zhang

Olin Business School
Washington University in St. Louis

Singapore
November 2026

A/B testing is everywhere

Platform	Reported experimentation scale	Year
LinkedIn	400+ concurrent tests each day, up from 50 at XLNT's launch	2015
Meta	About 10,000 versions of Facebook running at any time	2017
Airbnb	About 500 concurrent experiments, monitoring 50,000 metrics	2017
Netflix	About 250 recommendation tests per year, each with 100,000 subscribers	2018
ByteDance	About 1,500 experiments daily; more than 700,000 over the preceding years	2024
Booking.com	About 25,000 online experiments per year	2025



A/B testing in research

	Goal: Higher accuracy	Goal: Faster decisions
Pre-experiment design	**Balanced assignment** Make groups comparable before launch for cleaner, less noisy estimates. Stratification for covariate balance; switchback design	
Post-experiment analysis		

A/B testing in research

	Goal: Higher accuracy	Goal: Faster decisions
Pre-experiment design	**Balanced assignment** Make groups comparable before launch for cleaner, less noisy estimates. Stratification for covariate balance; switchback design	**Adaptive traffic** As evidence builds, shift more users to the winner during the run. Bandit algorithms
Post-experiment analysis		

A/B testing in research

	Goal: Higher accuracy	Goal: Faster decisions
Pre-experiment design	**Balanced assignment** Make groups comparable before launch for cleaner, less noisy estimates. Stratification for covariate balance; switchback design	**Adaptive traffic** As evidence builds, shift more users to the winner during the run. Bandit algorithms
Post-experiment analysis	**More information from data** Reduce variance or uncover heterogeneity after the experiment. Causal forests for heterogeneous treatment effects	

A/B testing in research

	Goal: Higher accuracy	Goal: Faster decisions
Pre-experiment design	**Balanced assignment** Make groups comparable before launch for cleaner, less noisy estimates. Stratification for covariate balance; switchback design	**Adaptive traffic** As evidence builds, shift more users to the winner during the run. Bandit algorithms
Post-experiment analysis	**More information from data** Reduce variance or uncover heterogeneity after the experiment. Causal forests for heterogeneous treatment effects	**Faster decisions after a sparse experiment** Avoid exhaustive enumeration. **Our focus**

AI and Experimentation

Learn beyond tested decisions

Experiments cannot cover every combination or policy.

1 **DeDL**

Deep Learning-Based Causal Inference for Large-Scale Combinatorial Experiments: Theory and Empirical Evidence

Ye et al.

Management Science (2026)

2 **DLPT**

Personalized Policy Learning through Discrete Experimentation: Theory and Empirical Evidence

Zhang, Zeng, Zhan, and Zhang

ICML (2026)

This presentation

AI and Experimentation

Learn beyond tested decisions

Experiments cannot cover every combination or policy.

1 DeDL

Deep Learning-Based Causal Inference for Large-Scale Combinatorial Experiments: Theory and Empirical Evidence

Ye et al.

Management Science (2026)

2 DLPT

Personalized Policy Learning through Discrete Experimentation: Theory and Empirical Evidence

Zhang, Zeng, Zhan, and Zhang

ICML (2026)

This presentation

Augment limited human data

Use AI to make costly human evidence go further.

3 AAE

Large Language Models for Market Research: A Data-augmentation Approach

Wang, Zhang, and Zhang

Marketing Science (2026, accepted)

4 GAI

Generative Augmented Inference of LLM-generated Data for Market Research: Theory and Empirical Evidence

Lu, Wang, Zhang, and Zhang

ICML (2026)

Today's focus

Personalized Policy Learning through Discrete Experimentation

Theory and Empirical Evidence

Zhiqi Zhang (Notre Dame) Zhiyu Zeng (SJTU)
Ruohan Zhan (LBS) Dennis Zhang (WashU)

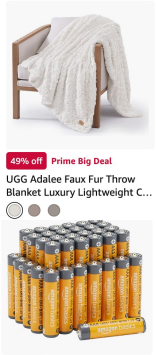




Singapore
November 2026

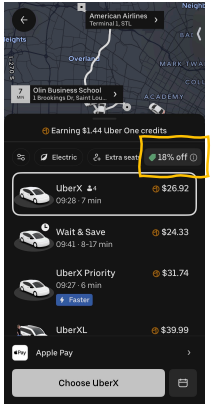
Strategic decisions with continuous variables

Amazon discounts



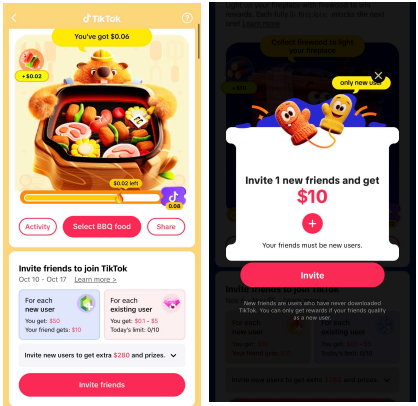
Amazon Discount

Uber promotions



Uber Promotion

TikTok cash rewards



TikTok Cash Reward

Choosing a reward for video production

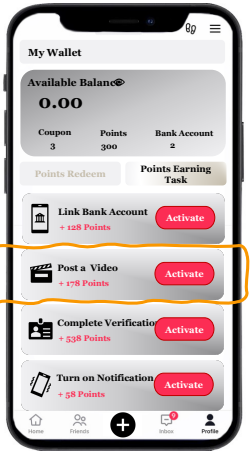
Imagine you manage the video boosting task team.

How many points should each user receive?



200? 400? 600? 800?

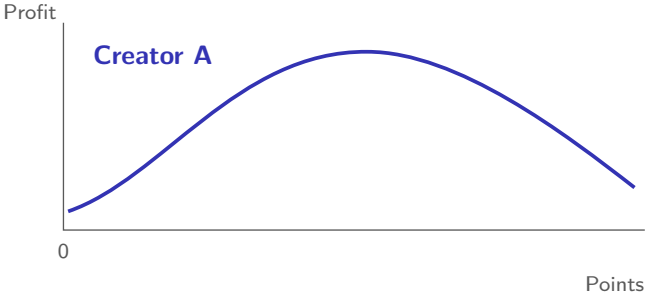
Points serve as cash rewards for video production.
100 points = $0.01.



DLPT: which reward should each creator receive?

Offer points for posting a video. More reward can raise participation *and* cost.

One reward for everyone?



A discrete experiment

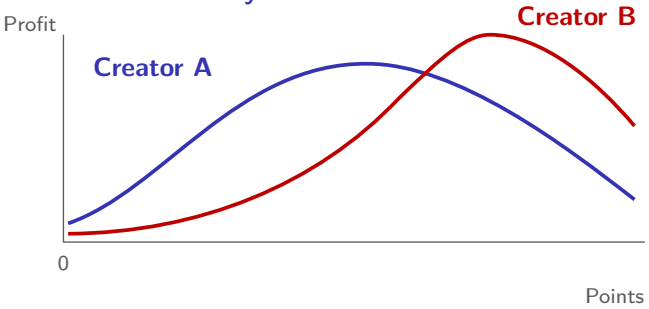
Only a few rewards are tested:

$$T \in \{0, 178, 358, 538, 718\}.$$

DLPT: which reward should each creator receive?

Offer points for posting a video. More reward can raise participation *and* cost.

One reward for everyone?



A discrete experiment

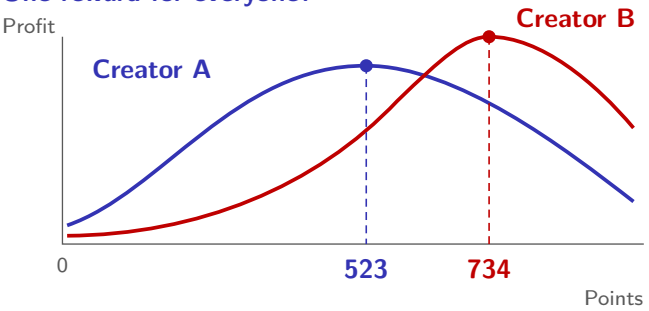
Only a few rewards are tested:

$$T \in \{0, 178, 358, 538, 718\}.$$

DLPT: which reward should each creator receive?

Offer points for posting a video. More reward can raise participation *and* cost.

One reward for everyone?



A discrete experiment

Only a few rewards are tested:

$$T \in \{0, 178, 358, 538, 718\}.$$

A continuous decision

Choose a reward for each user:

$$t^*(x) = \arg\max_{t \in \mathcal{T}} \pi(x, t).$$

$\pi(x, t)$ is expected content value minus reward cost.

Learn continuous, personalized rewards from discrete experimental data.

Industry practice: discrete experiments

Binary treatments

Compare no reward with a single reward level.

$$\{0,\ 900\} \text{ points}$$

Industry practice: discrete experiments

Binary treatments

Compare no reward with a single reward level.

$$\{0,\ 900\} \text{ points}$$

Several selected treatments

Compare no reward with a few selected "lucky numbers."

$$\{0,\ 178,\ 358,\ 518,\ 738\} \text{ points}$$

Industry practice: discrete experiments

Binary treatments

Compare no reward with a single reward level.

$$\{0,\ 900\} \text{ points}$$

Several selected treatments

Compare no reward with a few selected "lucky numbers."

$$\{0,\ 178,\ 358,\ 518,\ 738\} \text{ points}$$

Granular discretization

Compare many closely spaced treatments.

$$\{0,\ 100,\ 200, \ldots,\ 900\} \text{ points}$$

Limitation 1: statistical power decreases

For fixed sample size n, power falls as the number of tested levels and user segments increases.

Limitation 1: statistical power decreases

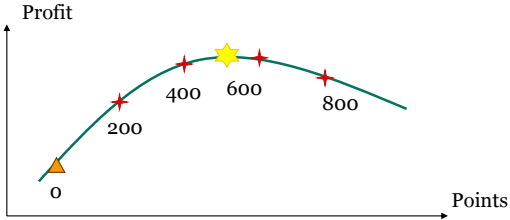
For fixed sample size n, power falls as the number of tested levels and user segments increases.

	Statistical power		
Treatment levels	2 segments	10 segments	20 segments
2	1.000	0.999	0.997
5	0.999	0.985	0.840
10	0.999	0.865	0.435

Sample size: 10,000. Significance level: 0.05. Effect size: 1. Response variance: 1.

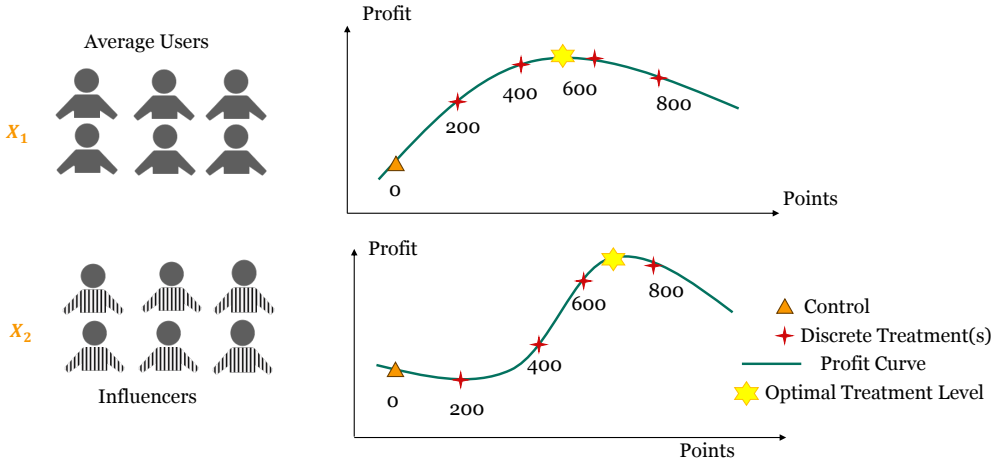
Limitation 2: the optimum may lie between tested levels





Control
Discrete Treatment(s)
Profit Curve
Optimal Treatment Level

Limitation 3: optimal treatment may be heterogeneous

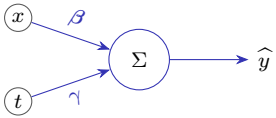


Conventional extrapolation

Y_i: video upload indicator X_i: 77 user features T_i: reward points

Linear model

$$Y_i = \alpha + X_i^\top \boldsymbol{\beta} + \gamma T_i + \varepsilon_i.$$



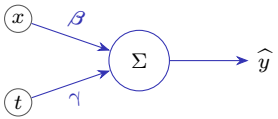
- Common linear treatment response.
- Nonlinearities and interactions must be specified.

Conventional extrapolation

Y_i: video upload indicator X_i: 77 user features T_i: reward points

Linear model

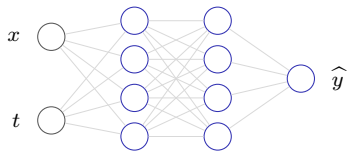
$$Y_i = \alpha + X_i^\top \boldsymbol{\beta} + \gamma T_i + \varepsilon_i.$$



- Common linear treatment response.
- Nonlinearities and interactions must be specified.

Pure deep learning

$$Y_i = f_{\boldsymbol{\eta}}(X_i, T_i) + \varepsilon_i.$$



- Flexible nonlinearities and interactions.
- No automatic guarantee for policy value or regret.

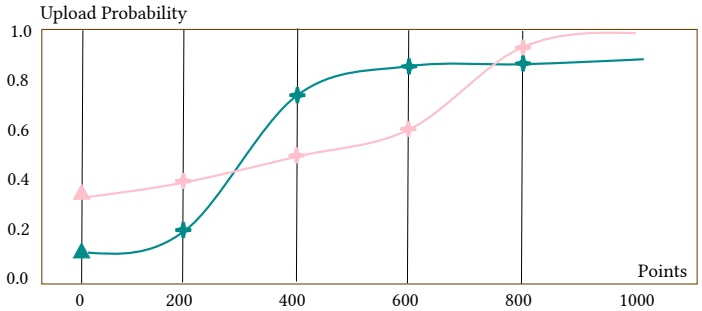
Research question

How can we learn optimal personalized policies for continuous variables from experiments with limited discrete treatments?

- Provable theoretical guarantees
- Strong empirical performance
- Preservation of existing experimentation infrastructure
- High-dimensional user features and nonlinear responses

DLPT: policy value estimation

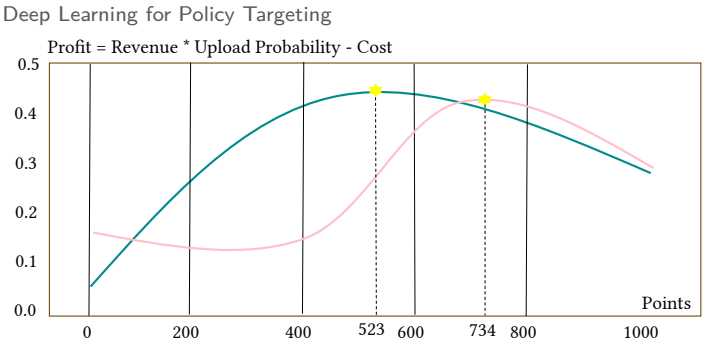
Deep Learning for Policy Targeting



1. Capture user heterogeneity with deep learning.
2. Model responses semiparametrically.
3. Use DML for $\sqrt{n}$-consistent policy evaluation.

Discrete experiment $\longrightarrow$ Policy evaluation $\longrightarrow$ Policy learning

DLPT: personalized policy learning



Personalized optima

523 points for one user type.

734 points for another.

Policy learning regret at rate

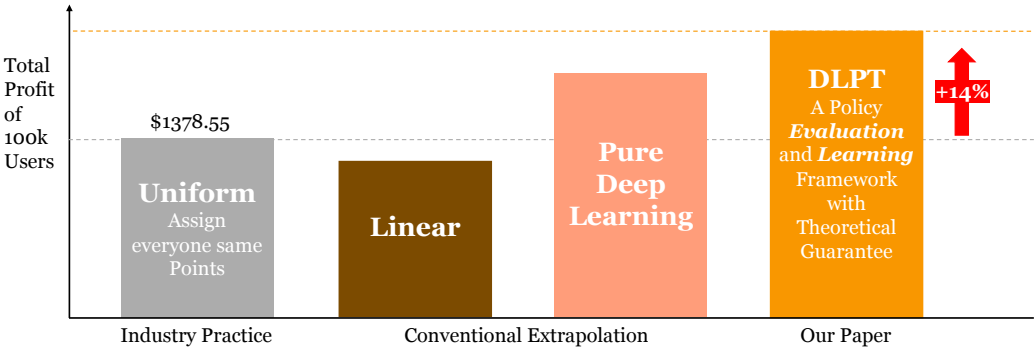
$$O\left(n^{-1/2}\right).$$

Discrete experiment $\longrightarrow$ Policy evaluation $\longrightarrow$ Optimal personalized policy

Continuous, personalized policies

Deep Learning for Policy Targeting (DLPT)

Uniform discrete A/B data $\longrightarrow$ continuous and personalized policies.



Policy evaluation and learning with theoretical guarantees. Total profit for 100,000 users.

A semiparametric model for discrete experiments

Observe $Z_i = (X_i, T_i, Y_i)$ with $X_i \in \mathcal{X} \subseteq \mathbb{R}^{d_X}$, $Y_i \in [0,1]$, and randomized $T_i \in \{t_0, \ldots, t_m\}$.

$$\mathbb{E}[Y_i \mid X_i = x, T_i = t] = G\big(\boldsymbol{\theta}_K^*(x), t\big),$$

$$G(\boldsymbol{\theta}, t) = \sigma\left(\sum_{k=0}^{K} \theta_k t^k\right), \qquad \sigma(u) = \frac{1}{1 + e^{-u}}.$$

A semiparametric model for discrete experiments

Observe $Z_i = (X_i, T_i, Y_i)$ with $X_i \in \mathcal{X} \subseteq \mathbb{R}^{d_X}$, $Y_i \in [0,1]$, and randomized $T_i \in \{t_0, \ldots, t_m\}$.

$$\mathbb{E}[Y_i \mid X_i = x, T_i = t] = G\big(\boldsymbol{\theta}_K^*(x), t\big),$$

$$G(\boldsymbol{\theta}, t) = \sigma\left(\sum_{k=0}^{K} \theta_k t^k\right), \qquad \sigma(u) = \frac{1}{1+e^{-u}}.$$

User heterogeneity

A deep neural network estimates the coefficient functions $\theta_0^*(x), \ldots, \theta_K^*(x)$.

A semiparametric model for discrete experiments

Observe $Z_i = (X_i, T_i, Y_i)$ with $X_i \in \mathcal{X} \subseteq \mathbb{R}^{d_X}$, $Y_i \in [0,1]$, and randomized $T_i \in \{t_0, \ldots, t_m\}$.

$$\mathbb{E}[Y_i \mid X_i = x, T_i = t] = G\big(\boldsymbol{\theta}_K^*(x), t\big),$$

$$G(\boldsymbol{\theta}, t) = \sigma\left(\sum_{k=0}^{K} \theta_k t^k\right), \qquad \sigma(u) = \frac{1}{1+e^{-u}}.$$

User heterogeneity

A deep neural network estimates the coefficient functions $\theta_0^*(x), \ldots, \theta_K^*(x)$.

Treatment response

A polynomial provides structure between experimental treatment levels. Identification requires $K \leq m$.

Higher polynomial degree expands approximation capacity under smoothness conditions.
Weierstrass (1885); Jackson (1912).

Policy evaluation and learning objectives

A policy $\pi \in \Pi$ assigns a continuous treatment: $\pi : \mathcal{X} \to \mathcal{T}$.

$$V_K(\pi) = \mathbb{E}_X \left[H\big(\boldsymbol{\theta}_K^*(X), \pi(X)\big) \right],$$

$$H(\boldsymbol{\theta}, t) = w\, G(\boldsymbol{\theta}, t) - ct.$$

w: revenue per response $\quad c$: cost per treatment unit

Policy evaluation

Estimate the value of treatments beyond the experimental grid.

Policy evaluation and learning objectives

A policy $\pi \in \Pi$ assigns a continuous treatment: $\pi : \mathcal{X} \to \mathcal{T}$.

$$V_K(\pi) = \mathbb{E}_X\left[H\left(\boldsymbol{\theta}_K^*(X), \pi(X)\right)\right],$$
$$H(\boldsymbol{\theta}, t) = w\, G(\boldsymbol{\theta}, t) - ct.$$

w: revenue per response c: cost per treatment unit

Policy evaluation

Estimate the value of treatments beyond the experimental grid.

Policy learning

Choose a policy in Π that minimizes regret relative to the best policy in that class:

$$R_K(\pi) = V_K(\pi_K^*) - V_K(\pi), \qquad \pi_K^* \in \arg\max_{\pi \in \Pi} V_K(\pi).$$

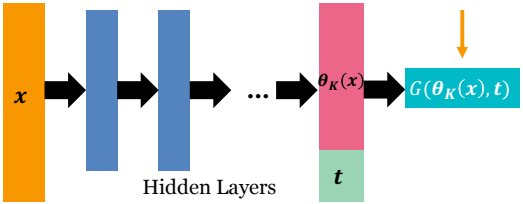
DLPT: three-stage method

Stage	Object and guarantee
1. Nuisance estimation	Structured deep neural network for heterogeneous coefficients $\boldsymbol{\theta}_K^*(x)$.
2. Policy evaluation	Neyman-orthogonal score: plug-in estimate minus a debiasing term.
3. Policy optimization	Maximize estimated policy value; regret scales at $n^{-1/2}$.

Three sources of error

$$\text{Real world} \xrightarrow{\text{approximation}} \text{Model} \xrightarrow{\text{estimation}} \text{Estimated value} \xrightarrow{\text{optimization}} \text{Policy}.$$

Stage 1: structured deep learning



The network maps user features into heterogeneous coefficients $\widehat{\boldsymbol{\theta}}(x)$.

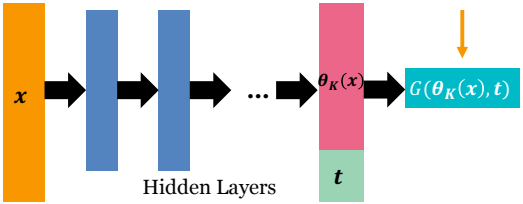
The known link G maps those coefficients and the reward into a response probability.

$$\widehat{\boldsymbol{\theta}} \in \underset{\boldsymbol{\theta} \in \mathcal{F}_{\mathrm{DNN}}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \ell\big(Y_i, G\big(\boldsymbol{\theta}(X_i), T_i\big)\big)\,.$$

Discrete outcomes: cross-entropy loss. For binary Y,

$$\ell(y, p) = -y \log p - (1 - y) \log(1 - p).$$

Stage 1: structured deep learning



The network maps user features into heterogeneous coefficients $\widehat{\boldsymbol{\theta}}(x)$.

The known link G maps those coefficients and the reward into a response probability.

$$\widehat{\boldsymbol{\theta}} \in \underset{\boldsymbol{\theta} \in \mathcal{F}_{\text{DNN}}}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \ell\big(Y_i, G\big(\boldsymbol{\theta}(X_i), T_i\big)\big)\,.$$

Discrete outcomes: cross-entropy loss. For binary Y,

$$\ell(y, p) = -y \log p - (1 - y) \log(1 - p).$$

Nuisance estimation

$\left\| \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_K^* \right\|_{L_2} = o_p(n^{-1/4})$ under smoothness and network-capacity conditions.

Continuous outcomes: squared-error loss. Yarotsky (2017); Farrell et al. (2021).

Stages 2 and 3: debias evaluation, then optimize

Plug-in value

For $z = (x, t, y)$ and loss $\ell(z; \boldsymbol{\theta})$,

$$\underbrace{H\big(\boldsymbol{\theta}(x), \pi(x)\big)}_{\text{plug-in value}}$$

Under the regularity conditions of the framework. Newey (1994); Chernozhukov et al. (2018); Farrell et al. (2020).

Stages 2 and 3: debias evaluation, then optimize

Neyman-orthogonal score

For $z = (x, t, y)$ and loss $\ell(z; \boldsymbol{\theta})$,

$$\psi(z, \pi; \boldsymbol{\theta}, \boldsymbol{\Lambda}) = \underbrace{H\big(\boldsymbol{\theta}(x), \pi(x)\big)}_{\text{plug-in value}} - \underbrace{\nabla_{\boldsymbol{\theta}} H\big(\boldsymbol{\theta}(x), \pi(x)\big)^{\top} \boldsymbol{\Lambda}(x)^{-1} \nabla_{\boldsymbol{\theta}} \ell(z; \boldsymbol{\theta}(x))}_{\text{correction for nuisance estimation error}}.$$

$\boldsymbol{\Lambda}(x)$ is the conditional expected Hessian of the loss.

Orthogonality reduces sensitivity to first-stage estimation error.

Under the regularity conditions of the framework. Newey (1994); Chernozhukov et al. (2018); Farrell et al. (2020).

Stages 2 and 3: debias evaluation, then optimize

Neyman-orthogonal score

For $z = (x, t, y)$ and loss $\ell(z; \boldsymbol{\theta})$,

$$\psi(z, \pi; \boldsymbol{\theta}, \boldsymbol{\Lambda}) = \underbrace{H\big(\boldsymbol{\theta}(x), \pi(x)\big)}_{\text{plug-in value}} - \underbrace{\nabla_{\boldsymbol{\theta}} H\big(\boldsymbol{\theta}(x), \pi(x)\big)^{\top} \boldsymbol{\Lambda}(x)^{-1} \nabla_{\boldsymbol{\theta}} \ell(z; \boldsymbol{\theta}(x))}_{\text{correction for nuisance estimation error}}.$$

$\boldsymbol{\Lambda}(x)$ is the conditional expected Hessian of the loss.

Evaluate and optimize

$$\widehat{V}_K(\pi) = \frac{1}{n} \sum_{i=1}^{n} \psi\big(Z_i, \pi; \widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\Lambda}}\big), \qquad \widehat{\pi} \in \arg\max_{\pi \in \Pi} \widehat{V}_K(\pi).$$

Orthogonality reduces sensitivity to first-stage estimation error.

Under the regularity conditions of the framework. Newey (1994); Chernozhukov et al. (2018); Farrell et al. (2020).

DLPT: approximation, estimation, and regret

$$\mathbb{E}[Y \mid X = x, T = t] = \sigma\left(\sum_{k=0}^{K} \theta_k^*(x) t^k\right).$$

Object	**Rate**
Model approximation	$\sup_{x,t} \lvert f(x,t) - G(\boldsymbol{\theta}_K(x), t)\rvert = O(K^{-s_t})$

Under the stated smoothness and regularity conditions. s_t: treatment smoothness; d: policy-class complexity. Approximation error remains distinct from estimation and policy regret.

DLPT: approximation, estimation, and regret

$$\mathbb{E}[Y \mid X = x, T = t] = \sigma\left(\sum_{k=0}^{K} \theta_k^*(x) t^k\right).$$

Object	**Rate**
Model approximation	$\sup_{x,t} \lvert f(x,t) - G(\boldsymbol{\theta}_K(x), t)\rvert = O(K^{-s_t})$
Stage 1: nuisance estimation	$\left\lVert \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_K^* \right\rVert_{L_2} = o_p(n^{-1/4})$

Under the stated smoothness and regularity conditions. s_t: treatment smoothness; d: policy-class complexity. Approximation error remains distinct from estimation and policy regret.

DLPT: approximation, estimation, and regret

$$\mathbb{E}[Y \mid X = x, T = t] = \sigma\left(\sum_{k=0}^{K} \theta_k^*(x) t^k\right).$$

Object	**Rate**
Model approximation	$\sup_{x,t} \lvert f(x,t) - G(\boldsymbol{\theta}_K(x), t)\rvert = O(K^{-s_t})$
Stage 1: nuisance estimation	$\left\lVert \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_K^* \right\rVert_{L_2} = o_p(n^{-1/4})$
Stage 2: policy evaluation	$\left\lvert \widehat{V}_K(\pi) - V_K(\pi) \right\rvert = O_p(n^{-1/2})$

Under the stated smoothness and regularity conditions. s_t: treatment smoothness; d: policy-class complexity. Approximation error remains distinct from estimation and policy regret.

DLPT: approximation, estimation, and regret

$$\mathbb{E}[Y \mid X = x, T = t] = \sigma\left(\sum_{k=0}^{K} \theta_k^*(x) t^k\right).$$

Object	**Rate**
Model approximation	$\sup_{x,t} \lvert f(x,t) - G(\boldsymbol{\theta}_K(x), t)\rvert = O(K^{-s_t})$
Stage 1: nuisance estimation	$\left\lVert \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}_K^* \right\rVert_{L_2} = o_p(n^{-1/4})$
Stage 2: policy evaluation	$\left\lvert \widehat{V}_K(\pi) - V_K(\pi) \right\rvert = O_p(n^{-1/2})$
Stage 3: policy optimization	$R_K(\widehat{\pi}) = O_p\left(\sqrt{d/n}\right)$

Under the stated smoothness and regularity conditions. s_t: treatment smoothness; d: policy-class complexity. Approximation error remains distinct from estimation and policy regret.

Why content creation matters

User-generated content sustains the platform

Creation supports consumption

- Fresh, diverse content for the feed
- Increased video watching time

Consumption encourages creation

- Desire for recognition
- Low barrier to entry
- Inspiration to create

Without a steady flow of creation

Content stagnates and engagement declines. Growth slows and revenue falls.

Field setting: a leading short-video platform

We collaborate with one of the world's top three short-video sharing platforms.

400M+
Daily active users

735M+
Monthly active users
Platform scale reported in the original study.

The platform's decision
Cash rewards can encourage uploads. Which reward should each user receive?

Personalize the incentive using user features.

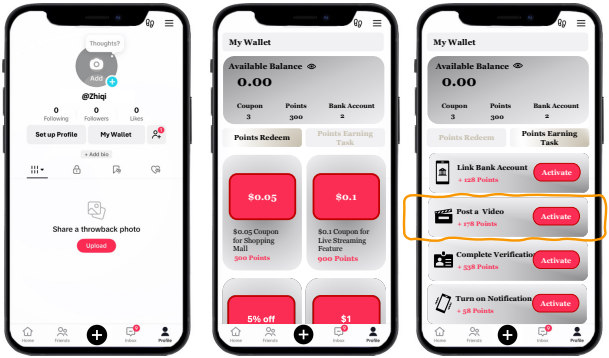
Cash rewards within the creator incentive strategy

Ways to encourage creation

- Cash incentives
- Easier editing tools
- Feedback and analytics
- Greater content exposure

Experimental intervention

A "Post a Video" task offers a point reward that users can activate.



(a) (b) (c)

Experimental design and randomization

August 23–29, 2021 **7,349,648 users** **12% of users**

Arm	Control	1	2	3	4
Reward (points)	0	178	358	538	718

Balance across randomized arms

Pre-experiment covariate	Reported p-value range
Female users	0.08–0.95
Highly active users	0.23–0.76
Users in moderately developed cities	0.22–0.85
App usage duration	0.34–0.95
Video-uploading days	0.26–0.80

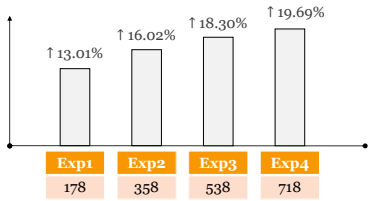
All reported balance-test p-values exceed 0.05. Treatment assignment is randomized.

Cash rewards increase uploads, with heterogeneous effects

Outcome: whether a user uploads a video during the experimental week.

(a) Average effect

Each reward versus control



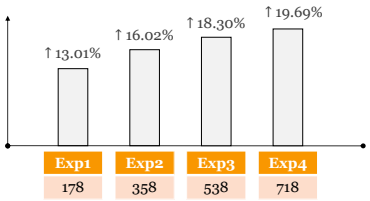
All adjacent-reward comparisons have $p < 0.05$.

Cash rewards increase uploads, with heterogeneous effects

Outcome: whether a user uploads a video during the experimental week.

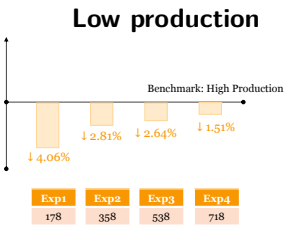
(a) Average effect

Each reward versus control



All adjacent-reward comparisons have $p < 0.05$.

(b) Heterogeneous effects



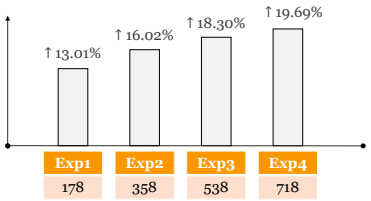
Low means below-median activity in the prior 30 days.
Each subgroup is compared with its high-activity counterpart.

Cash rewards increase uploads, with heterogeneous effects

Outcome: whether a user uploads a video during the experimental week.

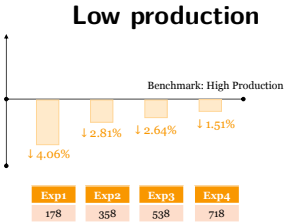
(a) Average effect

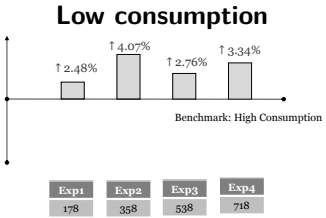
Each reward versus control



All adjacent-reward comparisons have $p < 0.05$.

(b) Heterogeneous effects





Low means below-median activity in the prior 30 days.
Each subgroup is compared with its high-activity counterpart.

DLPT implementation and benchmark models

Structured neural network

77 covariates; hidden layers of 128, 32, and 8 nodes; coefficient outputs feeding a polynomial in treatment and a sigmoid link.

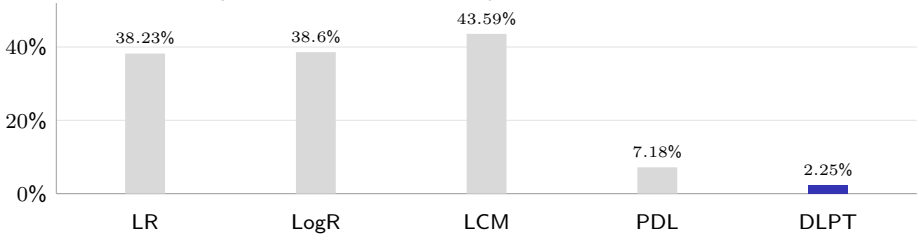
$$\widehat{\mu}_{\mathrm{DLPT}}(x,t) = \sigma\left(\sum_{k=0}^{K} \widehat{\theta}_k(x) t^k\right).$$

Benchmark	**Specification**
Linear regression (LR)	$\mu(x,t) = \alpha + x^\top \boldsymbol{\beta} + \gamma t$
Logistic regression (LogR)	$\mu(x,t) = \sigma\big(\alpha + x^\top \boldsymbol{\beta} + \gamma t\big)$
Logit choice model (LCM)	$\mu(x,t) = \sigma\big(\alpha + x^\top \boldsymbol{\beta} + t\, x^\top \boldsymbol{\gamma}\big)$
Pure deep learning (PDL)	$\mu(x,t) = f_{\boldsymbol{\eta}}(x,t)$, an unrestricted neural network

$\mu(x,t) = \mathbb{E}[Y \mid X = x, T = t]$. LCM allows the treatment response to vary with user features.

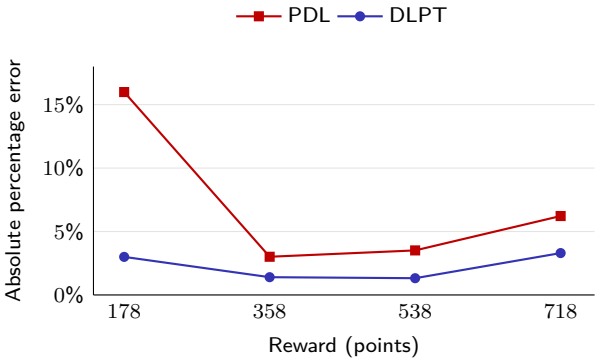
Counterfactual evaluation: DLPT has the lowest error

Leave one treatment out: train on the other arms, predict the held-out ATE, and compare it with the experimental estimate. Repeat for all four reward levels.



Mean absolute percentage error across the four held-out rewards. Lower is better.

Experimental coverage matters at the boundaries



Errors at each held-out reward. Series show each model's error separately.

Larger boundary errors

Both models perform better at intermediate reward levels.

Experiment design

Span the operational range, including its endpoints.

$$\{0,\ 0.33,\ 0.66,\ 1\}.$$

Policy learning: profit objective and feasible classes

Evaluate policies against a fitted DNN response surface with over 10 million parameters.

$$R(x,t) = \underbrace{y(x,t)}_{\text{upload probability}} \left(\underbrace{0.5}_{\text{unit revenue}} - \underbrace{0.1t}_{\text{reward cost}} \right).$$

Profit uses normalized treatment units. Policy levels below are in points.

Policy class	**Feasible assignment**
Finite observed	$\pi(x) \in \{0, 178, 358, 538, 718\}$ points

Policy learning: profit objective and feasible classes

Evaluate policies against a fitted DNN response surface with over 10 million parameters.

$$R(x,t) = \underbrace{y(x,t)}_{\text{upload probability}} \left(\underbrace{0.5}_{\text{unit revenue}} - \underbrace{0.1t}_{\text{reward cost}} \right).$$

Profit uses normalized treatment units. Policy levels below are in points.

Policy class	**Feasible assignment**
Finite observed	$\pi(x) \in \{0, 178, 358, 538, 718\}$ points
Discretized continuous	$\pi(x) \in \{0, 10, 20, \ldots, 1000\}$ points

Policy learning: profit objective and feasible classes

Evaluate policies against a fitted DNN response surface with over 10 million parameters.

$$R(x,t) = \underbrace{y(x,t)}_{\text{upload probability}} \left(\underbrace{0.5}_{\text{unit revenue}} - \underbrace{0.1t}_{\text{reward cost}} \right).$$

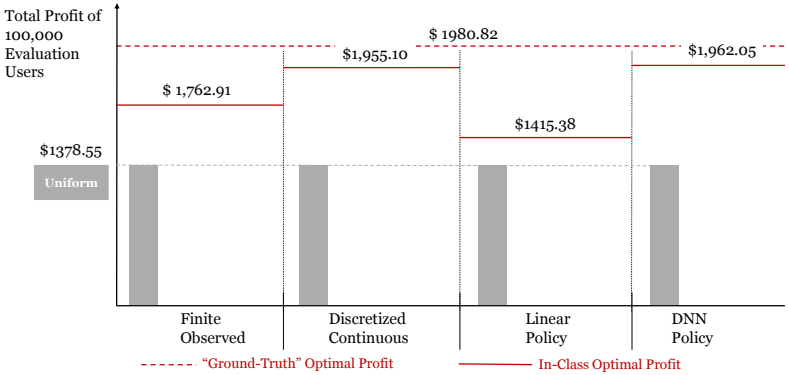
Profit uses normalized treatment units. Policy levels below are in points.

Policy class	**Feasible assignment**
Finite observed	$\pi(x) \in \{0, 178, 358, 538, 718\}$ points
Discretized continuous	$\pi(x) \in \{0, 10, 20, \ldots, 1000\}$ points
Linear	$\pi(x) = x^\top \boldsymbol{\alpha} + \beta$
DNN	$\pi(x) = f_{\boldsymbol{\omega}}(x)$, three layers of ten nodes

Optimize estimated profit within each feasible policy class.

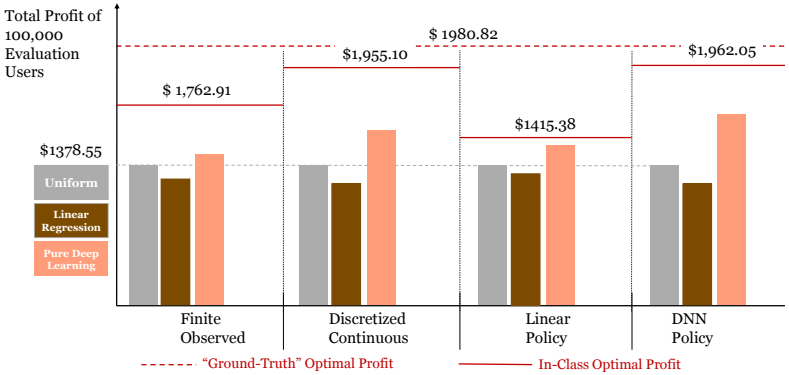
Personalization raises profit by 6–14%

Model-based profit for 100,000 users.



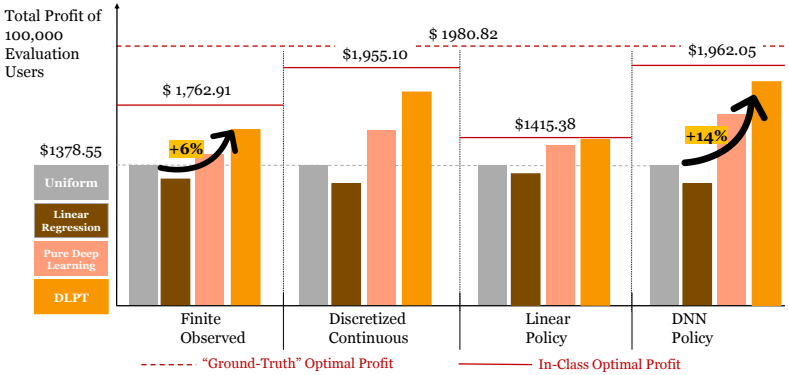
Personalization raises profit by 6–14%

Model-based profit for 100,000 users.



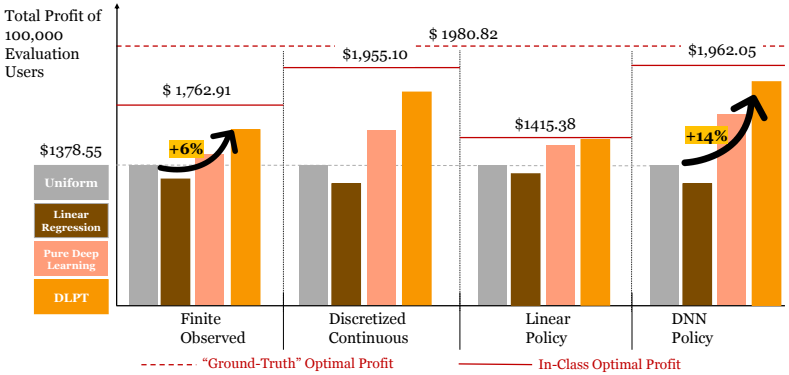
Personalization raises profit by 6–14%

Model-based profit for 100,000 users.



Personalization raises profit by 6–14%

Model-based profit for 100,000 users.



Policy choice

Prefer a simpler class when estimated values are similar.

Computation

Fine grid: 3.92 s

DNN policy: 0.42 s

Intel i9 benchmark.
Five DNN training epochs.

Takeaways

Continuous targeting from discrete experiments

DLPT combines flexible user heterogeneity with a structured treatment response.

Theory

Approximation, policy-value estimation, and policy regret have separate error bounds.

Evidence

A field experiment with 7.35 million users supports counterfactual evaluation. Model-based policy comparisons show profit gains of 6–14% over a uniform policy.

Implementation

The platform uses the framework. Code is publicly available:

Policy Learning through Discrete Experimentation (GitHub).

AI and Experimentation

Learn beyond tested decisions

Experiments cannot cover every combination or policy.

1 DeDL

Deep Learning-Based Causal Inference for Large-Scale Combinatorial Experiments: Theory and Empirical Evidence

Ye et al.

Management Science (2026)

2 DLPT

Personalized Policy Learning through Discrete Experimentation: Theory and Empirical Evidence

Zhang, Zeng, Zhan, and Zhang

ICML (2026)

This presentation

Augment limited human data

Use AI to make costly human evidence go further.

3 AAE

Large Language Models for Market Research: A Data-augmentation Approach

Wang, Zhang, and Zhang

Marketing Science (2026, accepted)

4 GAI

Generative Augmented Inference of LLM-generated Data for Market Research: Theory and Empirical Evidence

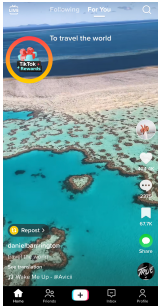
Lu, Wang, Zhang, and Zhang

ICML (2026)

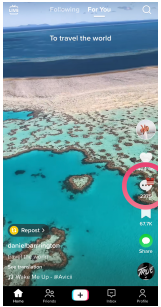
DeDL: what happens when we combine product changes?

A platform tests two interface changes to increase **watch time**.

Experiment A
Reward sticker



Experiment B
Gift sticker

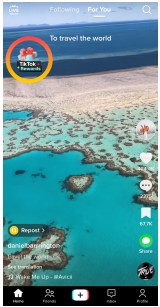


Ye et al. (2026), *Management Science*.

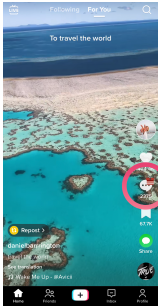
DeDL: what happens when we combine product changes?

A platform tests two interface changes to increase **watch time**.

Experiment A
Reward sticker



Experiment B
Gift sticker

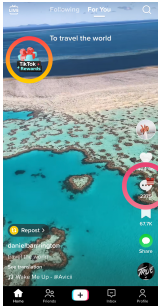


What if we turn on both?

The changes may reinforce each other or compete for attention.

Their combined effect need not equal the sum of their separate effects.

A + B
Both stickers

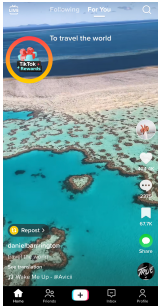


Ye et al. (2026), *Management Science*.

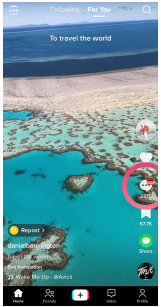
DeDL: what happens when we combine product changes?

A platform tests two interface changes to increase **watch time**.

Experiment A
Reward sticker



Experiment B
Gift sticker

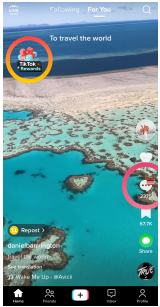


What if we turn on both?
The changes may reinforce each other or compete for attention.

Their combined effect need not equal the sum of their separate effects.

10 changes: 1,024 combinations
Testing every combination spreads the experiment's users across more arms.

A + B
Both stickers

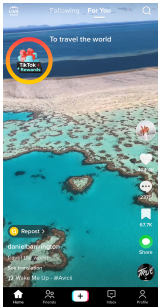


Ye et al. (2026), *Management Science*.

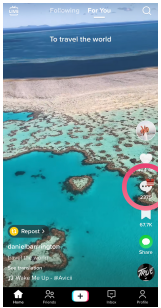
DeDL: what happens when we combine product changes?

A platform tests two interface changes to increase **watch time**.

Experiment A
Reward sticker



Experiment B
Gift sticker

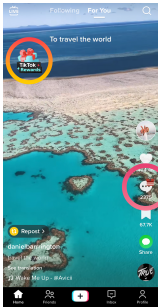


What if we turn on both?
The changes may reinforce each other or compete for attention.

Their combined effect need not equal the sum of their separate effects.

10 changes: 1,024 combinations
Testing every combination spreads the experiment's users across more arms.

A + B
Both stickers



Can we estimate untested effects and identify the best combination?

Ye et al. (2026), *Management Science*.

DeDL: structured individual responses, flexible averages

X: user characteristics; $T \in \{0,1\}^m$: treatments; Y: outcome.

Individual response: a generalized sigmoid

$$\mathbb{E}[Y \mid X = x, T = t] = G(\theta(x), t) = \theta_{m+1}(x)\, \sigma\Big(\theta_0(x) + \textstyle\sum_{j=1}^{m} \theta_j(x) t_j\Big).$$

Population response: average across user profiles

$$g(t) = \mathbb{E}_X[G(\theta(X), t)] = \sum_{\ell=1}^{K} p_\ell G(\theta(x_\ell), t).$$

p_ℓ is the population share of profile x_ℓ. For continuous X, take an integral.

Approximation: DeDL Theorem 1

With K distinct profiles, the weighted average can approximate any real-valued f on $\{0,1\}^m$:

$$\mathrm{RMS}_t[f(t) - g(t)] \leq C_{f,m} K^{-1/2}.$$

Approximation capacity does not itself identify untested effects.

DeDL: fit, debias, and infer

1. Fit the structured model with cross-fitting

Fit $X \mapsto \widehat{\theta}(X) \mapsto G(\widehat{\theta}(X), T)$ by squared error. Score each user with a model trained outside that user's fold.

DeDL: fit, debias, and infer

1. Fit the structured model with cross-fitting

Fit $X \mapsto \widehat{\theta}(X) \mapsto G(\widehat{\theta}(X), T)$ by squared error. Score each user with a model trained outside that user's fold.

2. Correct the model-implied treatment effect

$$\psi_i(t) = \underbrace{\widehat{H}_i(t)}_{\text{model-implied effect}} \underbrace{- [\nabla_\theta \widehat{H}_i(t)]^\top \widehat{\Lambda}_i^{-1} \nabla_\theta \widehat{\ell}_i}_{\text{debiasing correction}}.$$

Neyman orthogonality removes first-order sensitivity to nuisance-estimation error.

DeDL: fit, debias, and infer

1. Fit the structured model with cross-fitting

Fit $X \mapsto \widehat{\theta}(X) \mapsto G(\widehat{\theta}(X), T)$ by squared error. Score each user with a model trained outside that user's fold.

2. Correct the model-implied treatment effect

$$\psi_i(t) = \underbrace{\widehat{H}_i(t)}_{\text{model-implied effect}} \underbrace{- [\nabla_\theta \widehat{H}_i(t)]^\top \widehat{\Lambda}_i^{-1} \nabla_\theta \widehat{\ell}_i}_{\text{debiasing correction}}.$$

Neyman orthogonality removes first-order sensitivity to nuisance-estimation error.

3. Average corrected scores and report uncertainty

$$\widehat{\mu}_{\mathrm{DeDL}}(t) = \frac{1}{n} \sum_{i=1}^{n} \psi_i(t), \qquad 95\%\ \mathrm{CI}: \ \widehat{\mu}_{\mathrm{DeDL}}(t) \pm 1.96\,\mathrm{SE}.$$

Pointwise $\sqrt{n}$-consistent inference under the model, identification, convergence, and regularity assumptions.

DeDL: field experiment and estimation error

Three recommendation upgrades: **Discover, Live, and For You**.

Jan. 10–Feb. 1, 2021; 2,066,606 users. Outcome: daily watch time.

Eight randomized combinations: fit five, hold out three

Role	Combinations
Control	000
Individual upgrades and all-on	100, 010, 001, 111
Held out for validation	110, 101, 011

Error on held-out causal effects

Method	Mean absolute percentage error
Linear addition	30.06%
Pure deep learning	6.86%
DeDL	**1.75%**

DeDL recovers the sign and significance of all three held-out effects.

What if we have too little human data to test reliably?

Learning more from experiments

Use existing human data to evaluate decisions beyond tested arms.

What if we have too little human data to test reliably?

Learning more from experiments
Use existing human data to evaluate decisions beyond tested arms.

Augmenting human data with AI
Some experiments start with too few human responses to support reliable testing.

AI can act as synthetic humans to generate responses.

What if we have too little human data to test reliably?

Learning more from experiments
Use existing human data to evaluate decisions beyond tested arms.

Augmenting human data with AI
Some experiments start with too few human responses to support reliable testing.

AI can act as synthetic humans to generate responses.

AI is different from humans, and its responses can be strongly biased.

AAE: learning preferences with fewer surveys

Conjoint analysis asks people to choose among products with different attributes.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	$900	$800	$700
Screen	6.1-inch	6.7-inch	6.2-inch

Choices reveal how people value product attributes.

AAE: learning preferences with fewer surveys

Conjoint analysis asks people to choose among products with different attributes.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	$900	$800	$700
Screen	6.1-inch	6.7-inch	6.2-inch

Choices reveal how people value product attributes.

Human responses

Observe actual preferences.
Recruitment and survey completion take time and money.

AAE: learning preferences with fewer surveys

Conjoint analysis asks people to choose among products with different attributes.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	$900	$800	$700
Screen	6.1-inch	6.7-inch	6.2-inch

Choices reveal how people value product attributes.

Human responses

Observe actual preferences. Recruitment and survey completion take time and money.

AI responses

Generate many answers quickly. AI choices can differ systematically from human choices; simple pooling can increase bias.

AAE: learning preferences with fewer surveys

Conjoint analysis asks people to choose among products with different attributes.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	$900	$800	$700
Screen	6.1-inch	6.7-inch	6.2-inch

Choices reveal how people value product attributes.

Human responses

Observe actual preferences. Recruitment and survey completion take time and money.

AI responses

Generate many answers quickly. AI choices can differ systematically from human choices; simple pooling can increase bias.

Can inexpensive AI responses improve estimates of human preferences?

How do we debias AI responses?

Approach 1: train the AI model

Use human data to change model weights.

- Training can be costly; frontier models may restrict weight access or fine-tuning.
- New experiments can require new adaptation.

How do we debias AI responses?

Approach 1: train the AI model

Use human data to change model weights.

- Training can be costly; frontier models may restrict weight access or fine-tuning.
- New experiments can require new adaptation.

Our approach: calibrate the AI data

Collect a small human sample for this experiment.

1. Ask humans and AI the same questions.
2. Learn how their responses differ.
3. Apply the calibration to a larger AI sample.

How do we debias AI responses?

Approach 1: train the AI model

Use human data to change model weights.

- Training can be costly; frontier models may restrict weight access or fine-tuning.
- New experiments can require new adaptation.

Our approach: calibrate the AI data

Collect a small human sample for this experiment.

1. Ask humans and AI the same questions.
2. Learn how their responses differ.
3. Apply the calibration to a larger AI sample.

Calibrated data target human response patterns and can add statistical power.

An example of calibration

Learn a correction from matched human and AI answers

Matched willingness to pay ($)

Case	Human
A	800
B	700
C	900

An example of calibration

Learn a correction from matched human and AI answers

Matched willingness to pay ($)

Case	Human	AI
A	800	1,000
B	700	900
C	900	1,100

An example of calibration

Learn a correction from matched human and AI answers

Matched willingness to pay ($)

Case	Human	AI	AI − human
A	800	1,000	+200
B	700	900	+200
C	900	1,100	+200

Estimated correction

$200

AI answers exceed matched human answers.

An example of calibration

Learn a correction from matched human and AI answers

Matched willingness to pay ($)

Case	Human	AI	AI − human
A	800	1,000	+200
B	700	900	+200
C	900	1,100	+200

Estimated correction

$200

AI answers exceed matched human answers.



An example of calibration

Learn a correction from matched human and AI answers

Matched willingness to pay ($)

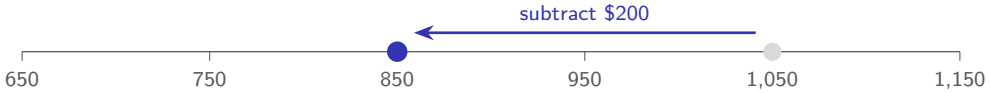
Case	Human	AI	AI − human
A	800	1,000	+200
B	700	900	+200
C	900	1,100	+200

Estimated correction

$200

AI answers exceed matched human answers.

Corrected mean: $850



$$\widehat{\mu} = \$1{,}050 - \$200 = \$850$$

AAE: observed human and calibrated AI choices

1. **Learn the calibration from matched human and AI answers**

$$\widehat{g}_j(X, Z; \widehat{\theta}) \approx \Pr(Y = j \mid X, Z).$$

2. **Fit one human preference model**

AAE: observed human and calibrated AI choices

1. Learn the calibration from matched human and AI answers

$$\widehat{g}_j(X, Z; \widehat{\theta}) \approx \Pr(Y = j \mid X, Z).$$

2. Fit one human preference model

Observed human choice

Human sample $\mathcal{H}$: m observations.

$$L_{\mathcal{H}}(\beta) = \sum_{i \in \mathcal{H}} \log p_{Y_i}(X_i; \beta).$$

Use the actual choice: all weight on the chosen option.

AAE: observed human and calibrated AI choices

1. Learn the calibration from matched human and AI answers

$$\widehat{g}_j(X, Z; \widehat{\theta}) \approx \Pr(Y = j \mid X, Z).$$

2. Fit one human preference model

Observed human choice

Human sample $\mathcal{H}$: m observations.

$$L_{\mathcal{H}}(\beta) = \sum_{i \in \mathcal{H}} \log p_{Y_i}(X_i; \beta).$$

Use the actual choice: all weight on the chosen option.

Calibrated AI choice

AI sample $\mathcal{A}$: n additional observations.

$$L_{\mathcal{A}}(\beta) = \sum_{i \in \mathcal{A}} \sum_{j} \widehat{g}_j(X_i, Z_i) \log p_j(X_i; \beta).$$

Use calibrated weights, e.g. 20% A, 60% B, 20% C.

AAE: observed human and calibrated AI choices

1. Learn the calibration from matched human and AI answers

$$\widehat{g}_j(X, Z; \widehat{\theta}) \approx \Pr(Y = j \mid X, Z).$$

2. Fit one human preference model

Observed human choice

Human sample $\mathcal{H}$: m observations.

$$L_{\mathcal{H}}(\beta) = \sum_{i \in \mathcal{H}} \log p_{Y_i}(X_i; \beta).$$

Use the actual choice: all weight on the chosen option.

Calibrated AI choice

AI sample $\mathcal{A}$: n additional observations.

$$L_{\mathcal{A}}(\beta) = \sum_{i \in \mathcal{A}} \sum_{j} \widehat{g}_j(X_i, Z_i) \log p_j(X_i; \beta).$$

Use calibrated weights, e.g. 20% A, 60% B, 20% C.

$$\widehat{\beta}_{\mathrm{AAE}} \in \arg\max_{\beta} \frac{L_{\mathcal{H}}(\beta) + L_{\mathcal{A}}(\beta)}{m + n}.$$

Both terms estimate the same human preference parameter β.

AAE: consistency and faster convergence

1. Consistency

$$\widehat{\beta}_{\mathrm{AAE}} \xrightarrow{p} \beta^*.$$

Recover human preferences without asymptotic AI-induced bias.

2. Faster convergence

Asymptotic: lower variance with informative AI data and efficient calibration.

Finite sample: a high-probability error bound,

$$\|\widehat{\beta}_{\mathrm{AAE}} - \beta^*\|_2 \leq C\sqrt{\varepsilon_{\mathrm{cal}}(m)}.$$

Estimation error depends on calibration accuracy.

Under the paper's calibration, sample-size, identification, and regularity conditions.

AAE: results across human sample sizes

Vaccine conjoint: change in MAPE relative to human-only estimation (pp).

Human responses	50	100	150	200
Human-only	0.00	0.00	0.00	0.00
Naive pooling	16.50	20.49	23.65	21.28
PPI	—	19.30	13.28	10.20
PPI++	—	4.42	3.29	1.99
AAE	**-15.74**	**-8.06**	**-3.31**	**-2.25**
AAE human-sample savings (%)	78.81	52.64	41.02	15.14

- Negative values mean lower estimation error than human-only estimation.
- AAE reduces error and saves human responses at every tested sample size.

GPT-4o with reasoning prompts; 1,000 AI responses; 50 repetitions. —: numerical instability.

GAI: from an AI answer to AI reasoning

The same phone-choice task: ask AI to choose *and explain*.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	$900	$800	$700
Screen	6.1-inch	6.7-inch	6.2-inch

AI answer: Phone B

Illustrative phone task and generated explanation; human choices remain the target.

GAI: from an AI answer to AI reasoning

The same phone-choice task: ask AI to choose *and explain*.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	$900	$800	$700
Screen	6.1-inch	6.7-inch	6.2-inch

AI answer: Phone B

Reasoning

"B has the **largest screen** and costs **$100 less than A**. I would pay **$100 more than C** for the extra screen size."

Illustrative phone task and generated explanation; human choices remain the target.

GAI: from an AI answer to AI reasoning

The same phone-choice task: ask AI to choose *and explain*.

Which phone would you choose?

	Phone A	Phone B	Phone C
Brand	Apple	Google	Samsung
Price	\$900	\$800	\$700
Screen	6.1-inch	6.7-inch	6.2-inch

AI answer: Phone B

Reasoning

"B has the **largest screen** and costs **\$100 less than A**. I would pay **\$100 more than C** for the extra screen size."

AI information

Answer + reasoning

Representation

Encode text as features

Learn from human data

Predict actual human choices

How can richer AI information improve estimates of human preferences?

Illustrative phone task and generated explanation; human choices remain the target.

GAI: an explicit debiasing correction

Y_i: human outcome; $W_i = 1$: human label observed; e_i: labeling probability.

$$\widetilde{Y}_i = \underbrace{\widehat{g}(X_i, Z_i)}_{\text{prediction}} + \underbrace{\frac{W_i}{e_i}\left[Y_i - \widehat{g}(X_i, Z_i)\right]}_{\text{debiasing correction}}.$$

Prediction

Predict human outcomes using task attributes and AI information with a flexible, nonparametric model.

Debiasing correction

Weight held-out human prediction errors by the inverse labeling probability.

Average the corrected outcomes

$$\widehat{\mu} = \frac{1}{N}\sum_{i=1}^{N} \widetilde{Y}_i.$$

Cross-fitted predictions, positive labeling probabilities, and the paper's sampling assumptions.

GAI: consistency and lower variance

1. Consistency

$$\widehat{\beta}_{\mathrm{GAI}} \xrightarrow{p} \beta^*.$$

Converges to the human target, even when AI answers are biased. Orthogonal debiasing and cross-fitting remove first-order nuisance errors.

2. Lower variance

$$\Sigma_{\mathrm{GAI}} \preceq \Sigma_{\mathrm{Human}}.$$

Lower asymptotic variance using the same human observations. Strict improvement under the paper's informativeness condition.

Under regularity and nuisance-estimation conditions. Efficiency uses known random labeling, $0 < e < 1$, and consistent prediction as both samples grow.

GAI: results across human sample sizes

Retail pricing: parameter-estimation error (MAPE, %; lower is better).

Human labels	100	200	300	400	500
Human-only	22.36	16.03	12.47	9.59	9.53
Naive pooling	22.22	21.97	19.99	19.67	19.43
PPI	26.91	15.31	14.10	13.98	12.88
PPI++	22.15	14.16	11.56	9.86	9.66
GAI	**12.43**	**9.14**	**8.74**	**6.57**	**7.60**

- **Sample savings:** GAI with 100 human labels has comparable error to 300 human-only labels (12.43% vs. 12.47%).
- **95% CI empirical coverage:** 96–100% for GAI; 50% for naive pooling.

1,000 auxiliary AI observations; baking-soda product; 50 repetitions.

Thank you

Dennis Zhang

denniszhang@wustl.edu

Singapore November 2026