DPO Unchained:

Your Training Algorithm is Secretly Disentangled in Human Choice Theory (and its Loss' Convexity is Dispensable)


Wenxuan Zhou


Shujian Zhang


Brice Magdalou


John Lambert


Ehsan Amid


Richard Nock


Andrew Hard

Google DeepMind
Google Research

UNIVERSITÉ DE MONTPELLIER

cnrs

Summary of the task

- Vocabulary:
  - Set of states / prompts $\mathcal{X}$, set of actions / answers $\mathcal{Y}$, probability simplex over $\mathcal{Y}$: $\Delta_{\mathcal{Y}}$
  - Want to learn a policy π, an application $\mathcal{X} \to \Delta_{\mathcal{Y}}$, from *data*
  - *Data* ~ distribution $\mathcal{D}$ over triples (x, y, y') meaning: given prompt $x \in \mathcal{X}$, answer $y \in \mathcal{Y}$ is *chosen* (by a *human*) over answer $y' \in \mathcal{Y}$

⚠ Setup learns $p(y \succ y'|x)$ ($\succ$: "*chosen over*") ≠ output we want, $\pi_\theta(y|x), \pi_\theta(y'|x)$ (policy)

- To get $p \longrightarrow \pi_\theta$ without mishaps requires carving appropriate normative properties in loss function(s) to keep training from reaching inadequate solutions
- Our starting point is a loss measured on p, $L(p)$

Google Research

Normative Bedrock (on *loss functions*, on **human choice**)

Choice

Policy

(pre-DPO, RLHF)

Google Research

DPO = Direct Preference Optimization, RLHF = Reinforcement Learning from Human Feedback

Normative Bedrock (on *loss functions*, on **human choice**)

Choice

$$L(p)$$

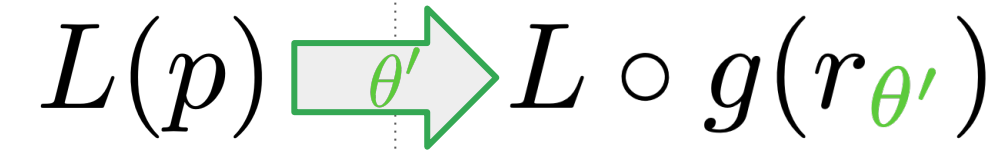
(pre-DPO)

Policy

Google Research

Normative Bedrock (on *loss functions*, on **human choice**)

Choice | Reward | Policy



(pre-DPO)

In green: ML / training element

Google Research

Normative Bedrock (on *loss functions*, on **human choice**)

Choice | Reward | Policy

$$L(p) \xrightarrow{\theta'} L \circ g(r_{\theta'})$$

(pre-DPO)

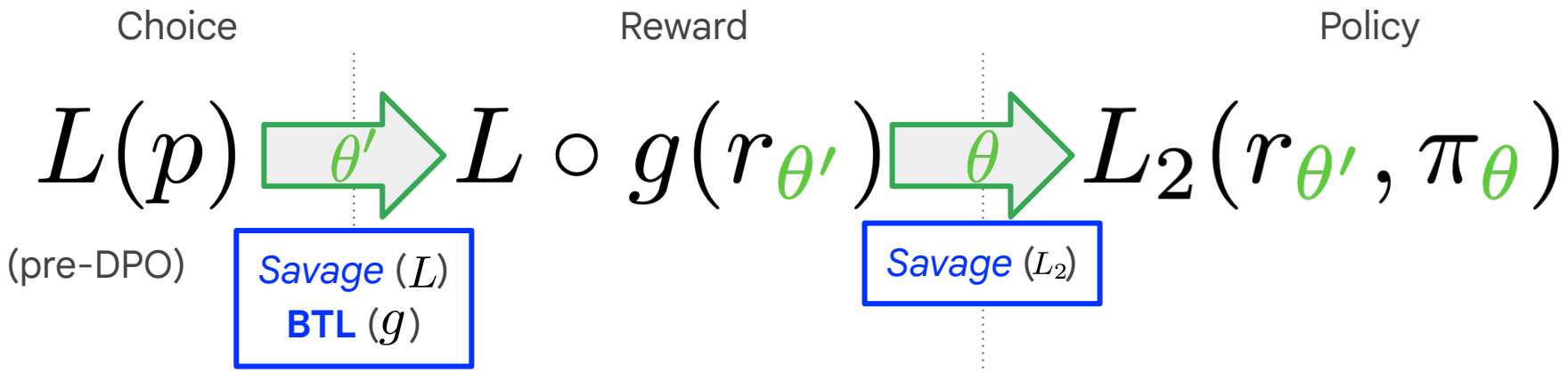
Savage (L)
BTL (g)

BTL = Bradley-Terry-Luce

In green: ML / training element

Google Research

Normative Bedrock (on *loss functions*, on **human choice**)



BTL = Bradley-Terry-Luce

In green: ML / training element

Google Research

Normative Bedrock (on *loss functions*, on **human choice**)

Choice | Reward | Policy

$$L(p) \xrightarrow{\theta'} L \circ g(r_{\theta'}) \xrightarrow{\theta} L_2(r_{\theta'}, \pi_\theta)$$

(pre-DPO)

Savage (L)
BTL (g)

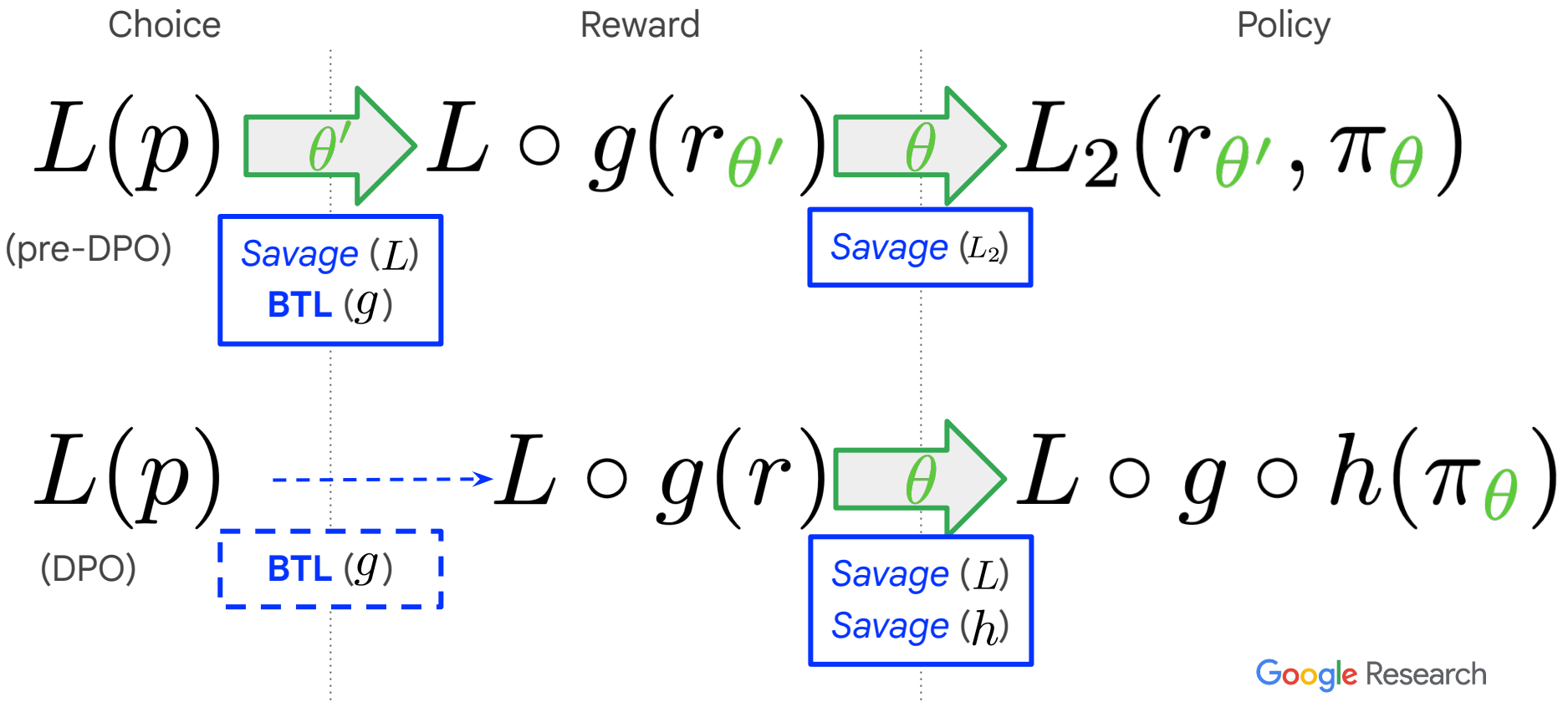
Savage (L_2)

$$L(p)$$

(DPO)

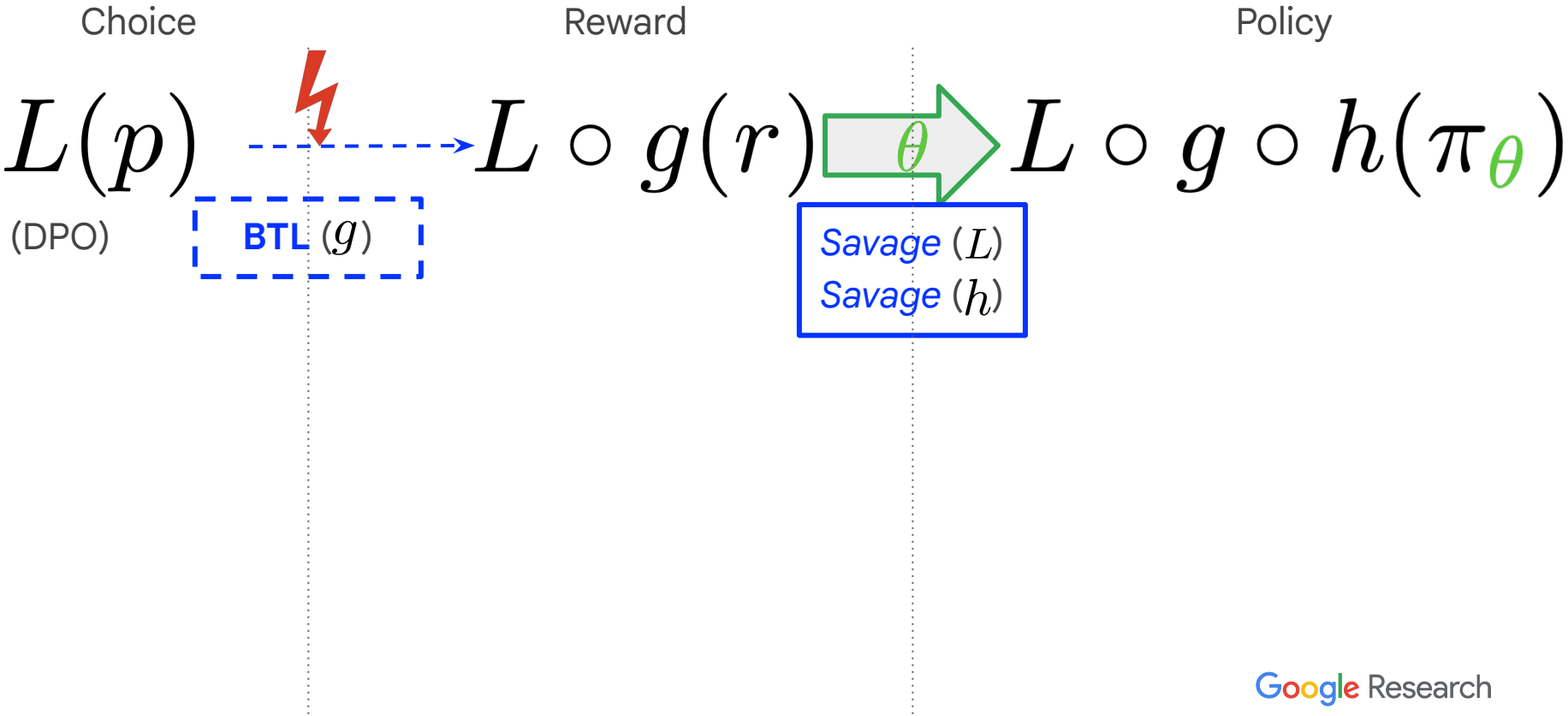
Google Research

Normative Bedrock (on *loss functions*, on **human choice**)



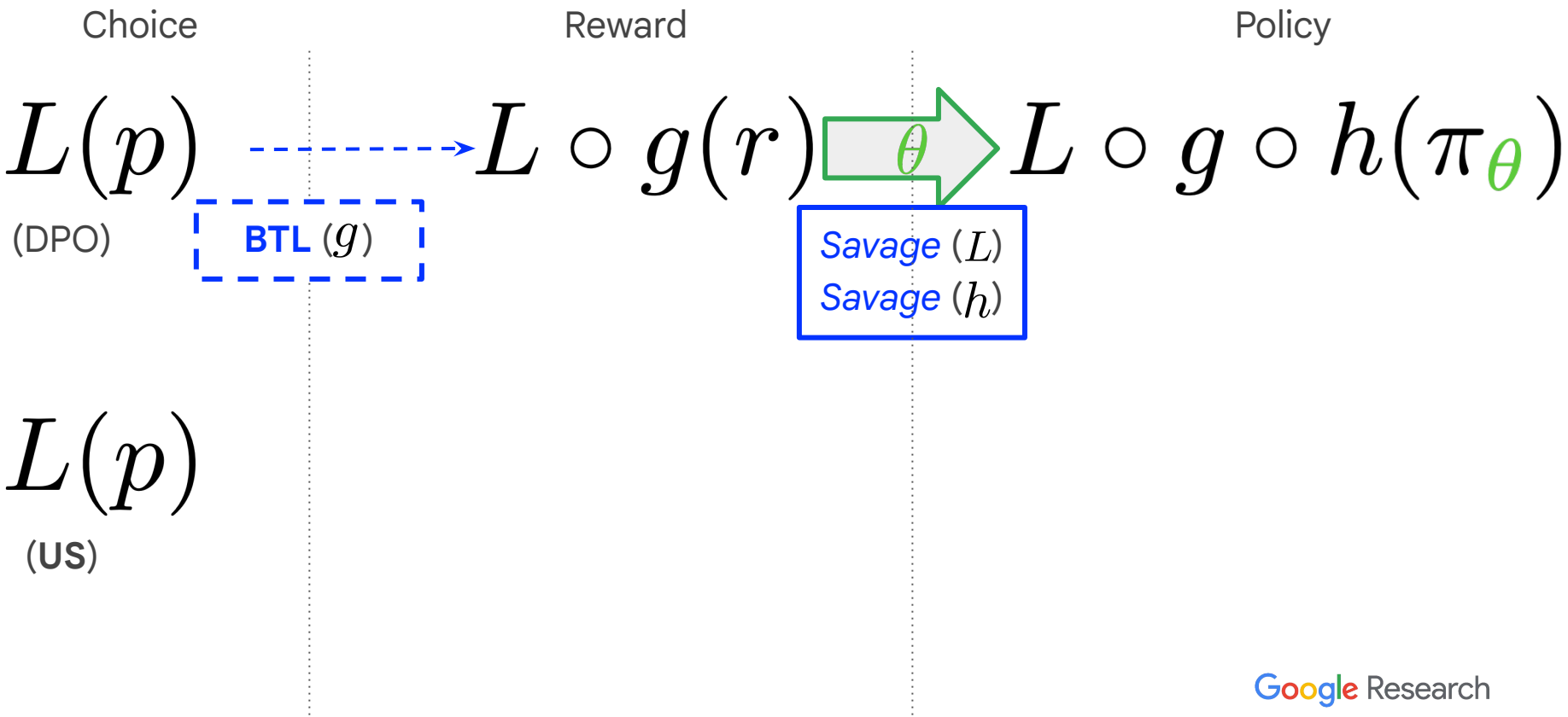
Google Research

Normative Bedrock (on *loss functions*, on **human choice**)



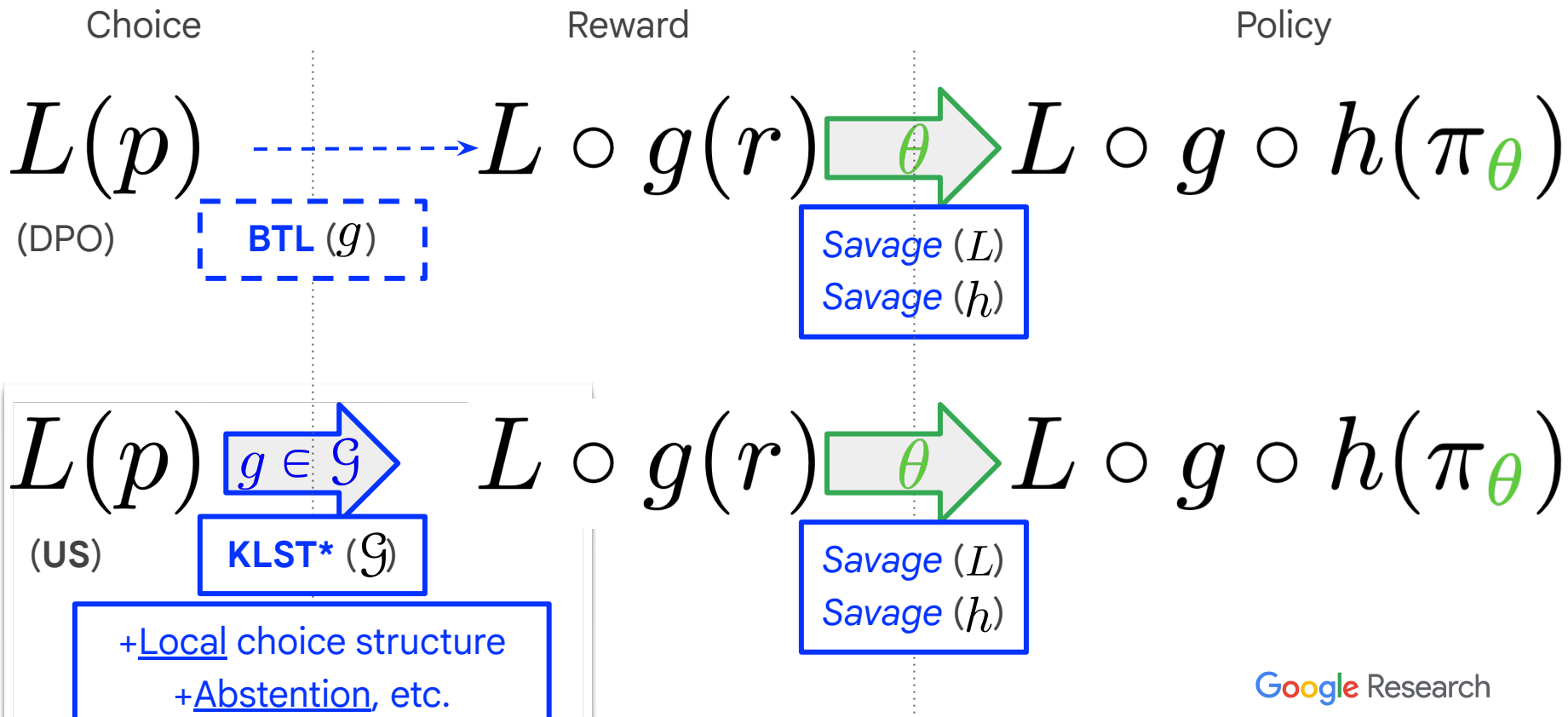
Google Research

Normative Bedrock (on *loss functions*, on **human choice**)



Google Research

Normative Bedrock (on *loss functions*, on **human choice**)



Google Research

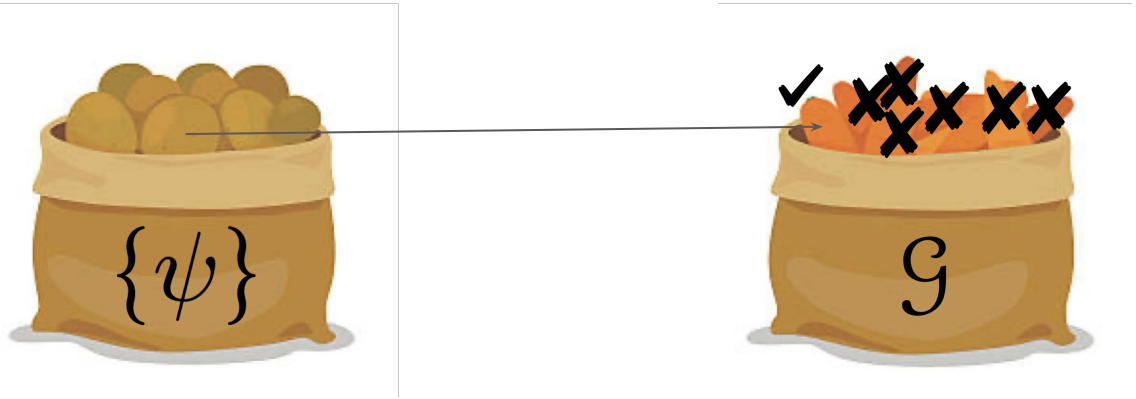
Surprise(s) – among others

- The key loss function is $L \circ g \doteq \psi$ (logistic loss for DPO)
- A classical issue with normative approaches is why should we pick one feasible option instead of another – especially true for human choice (e.g. one specific instance of **KLST***)

Google Research

Surprise(s) – among others

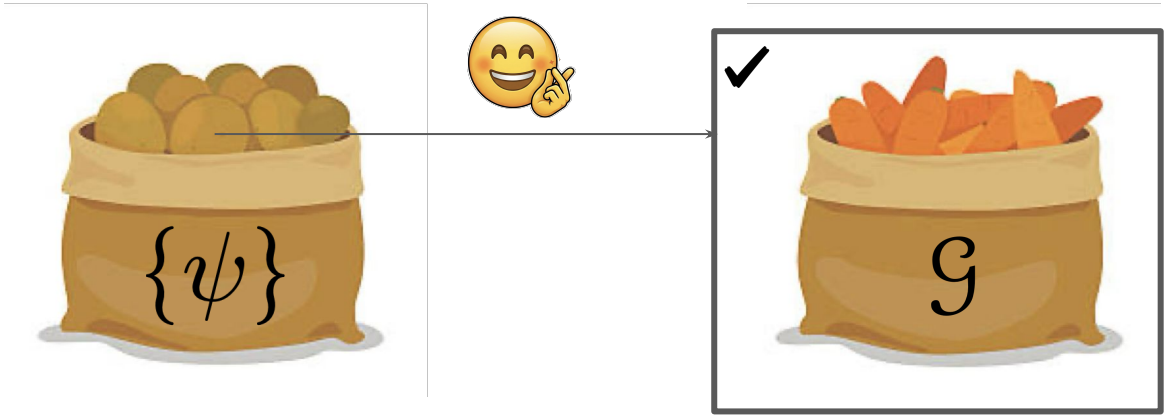
- The key loss function is $L \circ g \doteq \psi$ (logistic loss for DPO)
- A classical issue with normative approaches is why should we pick one feasible option instead of another – especially true for human choice (e.g. one specific instance of **KLST***)
- We were expecting this issue showing in "full force" for the design of ψ ...



Google Research

Surprise(s) – among others

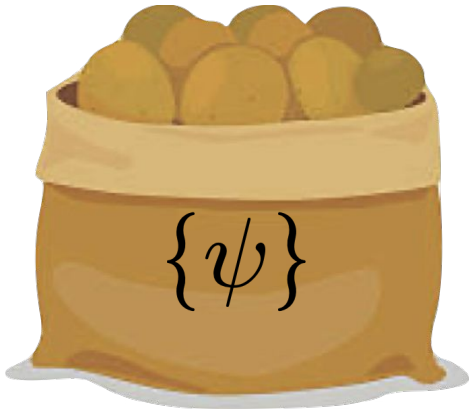
- The key loss function is $L \circ g \doteq \psi$ (logistic loss for DPO)
- A classical issue with normative approaches is why should we pick one feasible option instead of another – especially true for human choice (e.g. one specific instance of **KLST***)
- We were expecting this issue showing in "full force" for the design of ψ ...
  but instead got the possibility for *any* applicable loss ψ to map to *any* human choice model...



Google Research

Surprise(s) – among others

- The key loss function is $L \circ g \doteq \psi$ (logistic loss for DPO)
- A classical issue with normative approaches is why should we pick one feasible option instead of another – especially true for human choice (e.g. one specific instance of **KLST***)
- We were expecting this issue showing in "full force" for the design of ψ ...
  but instead got the possibility for *any* applicable loss ψ to map to *any* human choice model...
  and the set of applicable losses to be larger than expected (e.g. no convexity requirement)



Google Research

Thank You

Wenxuan Zhou	Google DeepMind
Shujian Zhang	Google DeepMind
Brice Magdalou	CEE-M, Univ. Montpellier, CNRS, INRAé, Institut Agro
John Lambert	Google DeepMind
Ehsan Amid	Google DeepMind (former)
Richard Nock	Google Research
Andrew Hard	Google DeepMind

Google Research