Decentralized Aggregation of LLM Predictions via Advantage-Aligned Wagering

Yuhong Luo
Rutgers University
y.luo@rutgers.edu

David M. Pennock
DIMACS, Rutgers University
dpennock@dimacs.rutgers.edu

Xintong Wang
Rutgers University
xintong.wang@rutgers.edu

Abstract

We introduce WALLA, a family of advantage-aligned wagering mechanisms for aggregating LLM predictions without requiring central access to models' private inputs or hidden states.[1] Each participating LLM reports a probabilistic prediction and a wager, which the aggregator uses to weight that prediction. We design a net payoff function that combines a leave-one-out scoring baseline with a quadratic penalty on each agent's wager. We show that for any fixed wager, truthfully reporting one's belief is a dominant strategy, even when that belief incorporates other agents' reports. For any fixed prediction report, an agent's best-response wager reveals its expected score advantage over the baseline. Each agent can learn its wagering policy independently using feedback from its own realized payoffs. We instantiate two variants that trade off *normality* and *no-arbitrage*, both with the worst-case deficit of the mechanism bounded by a constant. We evaluate WALLA in three settings where predictions differ because of model heterogeneity, differences in private information, or both. Across four question-answering benchmarks and a synthetic forecasting task, WALLA achieves predictive performance comparable to that of centralized baselines such as learned routers and stacked generalization, while supporting decentralized, advantage-aligned aggregation, uncertainty awareness, and incentive-compatible prediction reporting.

1 Introduction

Large language models (LLMs) with distinct domain expertise or access to private tools and data increasingly work together on complex tasks. When no single model dominates across domains, combining their outputs can draw on their complementary strengths. In forecasting and question answering, this can take the form of aggregating their probabilistic predictions. A key challenge is deciding how much influence each model should have on the aggregate as the question and the information available to each model vary.

Centralized methods learn to route queries using features of the input [1, 2, 3, 4] or assign question-dependent weights using combined model representations [5]. When LLMs are operated independently, for example, by healthcare providers [6], financial institutions [7] or competing AI labs [8], their private contexts and hidden states may be unavailable to the aggregator. The aggregator must assign weights without directly observing what gives a model an advantage on the current question.

Uniform averaging weights predictions equally. Other methods adapt weights using prediction confidence [9] or input perplexity [10], while keeping inputs or model internals private. These signals can diverge from prediction quality, e.g., due to miscalibration. When such signals are self-reported, participants who value influence over the aggregate may also have an incentive to misreport them.

We frame decentralized aggregation as an elicitation problem, seeking each model to reveal how well it expects to answer a question given its expertise and any private information. Models that expect to

[1] Source code: `https://github.com/chailab-rutgers/WALLA`.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

outperform the pool should receive more weight, while those that expect to do worse receive little or none. We build on *wagering mechanisms* [11, 12, 13], in which each participant submits a probabilistic prediction and a nonnegative wager. Once the outcome is observed, the mechanism evaluates predictions using a proper scoring rule and determines payoffs from the scores scaled by wagers.

We propose using each model's wager as the weight on its prediction and design the payoff function so that an agent is incentivized to wager more when it expects to outperform its peers by a larger margin, giving its prediction more influence on the aggregate. The canonical *weighted-score wagering mechanism* (WSWM) [11] satisfies key incentive properties but has two limitations for our setting. First, its guarantee of truthful reporting relies on *immutable beliefs*, the assumption that agents would not revise their beliefs upon observing others' reports. Second, with predictions and opponents' wagers fixed, a rational agent will wager either the largest permitted amount if it expects to outperform or zero otherwise, producing a binary signal that is less informative as aggregation weights and as targets for learning wagering strategies that reflect the agent's advantage on a given question.

We introduce WALLA, a family of *advantage-aligned wagering mechanisms for LLM aggregation* that resolve both limitations through a new payoff function. Our net payoff function combines a leave-one-out scoring baseline with a quadratic penalty on each agent's wager and yields three desirable properties: (1) *dominant-strategy incentive compatibility for predictions*: reporting one's true belief maximizes expected payoff regardless of opponents' strategies, even when that belief incorporates information from others' reports; (2) *advantage–wager alignment*: an agent's best-response wager is proportional to its expected score advantage over the baseline; and (3) *wager optimization with fixed prediction reports*: the best-response wager is well-defined for any fixed prediction report, even a suboptimal one, so agents can learn their wagering policies while keeping their predictors fixed.

We instantiate two variants of WALLA that trade off two guarantees. WALLA I satisfies *normality*: if agent j's prediction improves from agent i's perspective, then agent i's expected net payoff weakly decreases, holding all wagers and other predictions fixed. WALLA II satisfies *no-arbitrage*, which rules out a strictly positive net payoff for an agent in every possible outcome. For both variants, the mechanism's worst-case deficit is bounded by a constant independent of the number of participants and the total amount wagered.

We evaluate WALLA with homogeneous models given private contexts, heterogeneous models given only shared queries, and heterogeneous models given private contexts; these settings let us study comparative advantages arising from private information, model expertise, or both. Each model learns its wagering policy from its own payoff feedback, using either its hidden states as inputs or embeddings of the question and any private context when only API access is available. Across four question-answering benchmarks (MMLU, MedMCQA, ARC-Challenge, and PubMedQA) and a synthetic forecasting task (BayesX), WALLA performs comparably to centralized baselines such as learned routers and stacked generalization. In BayesX, our mechanism assigns more weight to better-informed models even when private evidence makes them less certain about the outcome.

1.1 Related Work

LLM prediction aggregation. Methods for ensembling LLMs [14] include voting over answers, as in self-consistency [15, 16], and combining token-level predictions [17]. Multi-agent debate lets models revise their answers through interaction [18, 19]. LLM self-evaluations and hidden states can provide signals about answer reliability [20, 21, 22]. Some methods use prediction confidence or input perplexity to guide selection or weighting [9, 10]. Learned routers select models based on query features [1, 23, 2, 3, 4]. Other learned approaches combine model outputs through stacked generalization [5] or rank and fuse generated responses [24]. In forecasting, Schoenegger et al. [25] find that aggregating LLM forecasts can rival the accuracy of aggregated human forecasts.

Mechanism design for LLMs. Recent work uses auctions to incorporate preferences into LLM-generated content, including sponsored summaries and advertising [26, 27, 28]. Bergemann et al. [29] study the joint elicitation of preferences and private information, with applications to LLMs. Our setting concerns predictions about outcomes that are later observed, which lets us use scoring rules to reward truthful reports and elicit models' expected score advantages.

Wagering mechanisms and prediction markets. Prediction markets aggregate information through prices that evolve as participants trade [30, 31, 32, 33, 34]. Wagering mechanisms settle submitted predictions and wagers after the outcome is observed [35, 11, 12, 13]. Raja et al. [36] use wagers to weight forecasts for a buyer, and Freeman et al. [37] use wagering mechanisms to design incentive-compatible no-regret algorithms that update expert weights.

2 Preliminaries

We first review proper scoring rules and the *weighted-score wagering mechanism* on which we build, then discuss desirable properties and motivate additional ones for extending the framework to LLMs.

Scoring rules and wagering mechanisms. We consider a probabilistic prediction setting, where a mechanism elicits predictions on uncertain outcomes from multiple agents and evaluates them based on the quality of their predictions. Let $\mathcal{M} = \{1, 2, \ldots, M\}$ denote a set of participating agents. A *wagering mechanism* operates in two stages. In the first stage, given a question $x \in \mathcal{X}$ with an unknown outcome $y \in \mathcal{Y}$, each agent i submits a prediction $p_i \in \Delta(\mathcal{Y})$ and wagers some non-negative amount $w_i \in \mathbb{R}_{\geq 0}$ on their prediction. We denote the resulting *prediction* and *wager profiles* by $\mathbf{p} \coloneqq (p_1, \ldots, p_M)$ and $\mathbf{w} \coloneqq (w_1, \ldots, w_M)$, and write $\mathbf{p}_{-i}$ and $\mathbf{w}_{-i}$ for the corresponding profiles excluding agent i. In the second stage, after the outcome is revealed, the mechanism determines the *net payoff* (the payment to the agent at settlement minus its initial wager) to each agent, denoted $\pi_i(\mathbf{p}, \mathbf{w}, y) \in \mathbb{R}$, based on the quality of her prediction and the amount wagered. When the context is clear, we use "payoff" as shorthand for "net payoff."

The quality of a prediction is measured by *a scoring rule*: $s \colon \Delta(\mathcal{Y}) \times \mathcal{Y} \to \mathbb{R}$, which assigns a numerical score $s(p, y)$ to a probabilistic forecast p upon realization of outcome y. A *strictly proper scoring rule* incentivizes a *risk-neutral* agent to report her true belief, which uniquely maximizes the expected score.

Definition 2.1 (Strictly Proper Scoring Rule)**.** *A scoring rule s is* strictly proper *if and only if for all $p, q \in \Delta(\mathcal{Y})$ and $p \neq q$,*

$$\mathbb{E}_{Y \sim q}[s(q, Y)] > \mathbb{E}_{Y \sim q}[s(p, Y)].$$

A prediction of $p = q$ is truthful*, as the agent simply reports their belief q.*

Example 2.1 (Brier Score [38, 39])**.** *$s(p, y) = c_1 - c_2\|p - \delta_y\|_2^2$, where $c_1 \in \mathbb{R}$, $c_2 \in \mathbb{R}_+$, and $\delta_y \in \Delta(\mathcal{Y})$ denotes the Dirac measure at y, the indicator vector of the realized outcome y.*

The Brier score, also known as the quadratic score, is a simple and widely used strictly proper scoring rule; it is bounded and for an agent with belief q, the expected score $\mathbb{E}_{Y \sim q}[s(p, Y)]$ is monotonically decreasing in $\|p - q\|_2^2$, the squared distance between the prediction p and the belief q.

A scoring rule is used in a wagering mechanism's payoff function $\pi : (\Delta(\mathcal{Y}) \times \mathbb{R}_{\geq 0})^M \times \mathcal{Y} \to \mathbb{R}^M$.

Definition 2.2 (Weighted-Score Wagering Mechanism (WSWM) [11])**.** *Given a scoring rule s, the WSWM specifies the payoff to agent i as the difference between their score and the wager-weighted average of all agents' scores, scaled by their own wager, i.e.,*

$$\pi_i^{\text{WSWM}}(\mathbf{p}, \mathbf{w}, y) = w_i \left(s(p_i, y) - \frac{\sum_{j=1}^{M} w_j s(p_j, y)}{\sum_{j=1}^{M} w_j} \right). \tag{1}$$

The WSWM based on a Brier score is also referred to as the *Brier betting* mechanism.

Properties of wagering mechanisms. Appendix A reviews desirable properties, including *incentive compatibility* (IC; Def. A.1), *individual rationality* (IR; Def. A.2), *(weak) budget balance* (Def. A.3), *normality* (Def. A.4), and *no-arbitrage* (Def. A.5). WSWM is exactly budget-balanced. With a strictly proper scoring rule, it satisfies IR and normality. When agents also hold immutable beliefs, WSWM is IC for prediction reports [11]. It can admit arbitrage [12].

WSWM has two main limitations. First, with predictions and opponents' wagers fixed and at least one opponent wagering positively, the best-response wager is the wager cap if the agent expects to outscore its opponents' wager-weighted average and zero if it expects to fall below that average. With a common cap, models with small and large positive advantages wager the same amount and receive equal weight in the aggregate. Near zero, a small error in estimating the advantage can switch the estimated best response between the two endpoints.

Second, WSWM's IC assumes *immutable beliefs*, meaning that an agent would hold the same belief about the outcome even if it observed its opponents' reports. When beliefs are mutable, and one's expected opponents' wagers carry information about the outcome, WSWM can favor reports that assign more probability to outcomes associated with larger opposing wagers, breaking IC. We aim to close both gaps by designing advantage-aligned mechanisms (Section 3.2) where the best-response wager reflects expected score advantage and IC holds under an arbitrary belief structure.

Warm-up: WSWM with non-strategic agents. We analyze the wagering mechanism in a simplified, non-strategic setting where agents report truthful predictions and wager their full current budgets each round. We consider a variant of WSWM with a scaled linear scoring rule and linear pooling, and show that the resulting dynamics admit connections to gradient-based learning, including equivalence to multiplicative weights updates, and that the induced aggregation recovers Bayesian model averaging [40]. The formal statements and proofs are deferred to Appendix B.

3 Designing Wagering Mechanisms for LLM Aggregation

This section presents our proposed *advantage-aligned wagering mechanism for LLM aggregation (WALLA)*. We first analyze how agents choose predictions and wagers for a single question to maximize their expected net payoffs (Sections 3.1–3.2). We then show how each agent can learn a wagering policy from its own realized payoffs across questions (Section 3.3).

3.1 Setting, Information Model, and Strategies

We now formalize the setting, information model, and agent strategy. Let $(\Omega, \mathcal{F})$ be a measurable space, where Ω is the set of possible states of the world and $\mathcal{F}$ is the σ-algebra of measurable events. Each state $\omega \in \Omega$ encodes the question X, the outcome $Y \in \mathcal{Y}$, and all agents' private signals $(S_1, \ldots, S_M)$. The private signal S_i represents additional information available to agent i beyond the question itself, such as outputs from tool calls or extra context. The information available to agent i is captured by $\mathcal{F}_i = \sigma(X, S_i) \subseteq \mathcal{F}$: all agents observe the question X, but may differ in their private signals S_i. Each agent i holds a prior belief $\mathcal{Q}_i$, a probability measure on $(\Omega, \mathcal{F})$ that may differ from the data-generating probability measure $\mathcal{P}$. When agents share a common prior, we write $\mathcal{Q}_i = \mathcal{Q}$ for all i.

A strategy for agent i is a pair (p_i, w_i), where $p_i \colon \Omega \to \Delta(\mathcal{Y})$ is a probabilistic prediction and $w_i \colon \Omega \to \mathbb{R}_{\geq 0}$ is a wager. Both are $\mathcal{F}_i$-measurable: they depend on the state ω through the question X and agent i's private signal S_i.[2] Each agent is *risk-neutral* and maximizes its expected net payoff on each question, given its information. When the state is clear from context, we write p_i, w_i, and y for $p_i(\omega)$, $w_i(\omega)$, and $Y(\omega)$, respectively.

3.2 Advantage-Aligned Wagering Mechanisms

We discuss three design components: (1) a scoring rule s that evaluates individual predictions, (2) a net payoff function that determines each agent's net payoff, and (3) an aggregation rule that combines predictions into a single forecast. The net payoff function is the main object of design.

Definition 3.1 (Net Payoff Function)**.** *Given a strictly proper scoring rule s and a constant $c_3 > 0$, the net payoff to agent i upon the realization of outcome $y \in \mathcal{Y}$ is*

$$\pi_i(\mathbf{p}, \mathbf{w}, y) := w_i(s(p_i, y) - b_{-i}(\mathbf{p}_{-i}, \mathbf{w}_{-i}, y) - c_3 w_i)\,, \tag{2}$$

where $b_{-i} \colon \Delta(\mathcal{Y})^{M-1} \times \mathbb{R}_{\geq 0}^{M-1} \times \mathcal{Y} \to \mathbb{R}$ is a baseline score. It depends on other agents' predictions and wagers. We refer to b_{-i} as the leave-one-out baseline *and $s(p_i, y) - b_{-i}$ the* comparative advantage*.*

The specific form of $b_{-i}(\cdot)$ is left free. We instantiate two concrete choices in Appendix C.6: In WALLA I, the baseline is the wager-weighted average of opponents' scores; In WALLA II, it is the score of their wager-weighted average prediction. Appendix C.6 analyzes their tradeoff between normality and no-arbitrage. We show that, under the Brier score, each variant's worst-case deficit is bounded by a constant independent of the number of participants and the total amount wagered.

When at least two agents wager positively, the mechanism settles their wagers using Eq. (2). Agents who wager zero receive zero net payoff. With fewer than two positive wagers, the mechanism cancels the round, refunds any wagers, and issues no aggregate prediction. For the analysis below, we impose the *participation assumption* that the wagering strategies satisfy $\mathcal{Q}_i(\sum_{j \neq i} w_j > 0) = 1$: under agent i's belief, at least one other agent will wager positively, and the round will settle almost surely.

Since w_i is $\mathcal{F}_i$-measurable, it factors out of $\mathbb{E}[\cdot \mid \mathcal{F}_i]$ under any probability measure. The expected net payoff decomposes as

$$\mathbb{E}_{\mathcal{Q}_i}[\pi_i \mid \mathcal{F}_i] = w_i \Bigg(\underbrace{\mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) \mid \mathcal{F}_i]}_{\text{self-score}} - \underbrace{\mathbb{E}_{\mathcal{Q}_i}[b_{-i} \mid \mathcal{F}_i]}_{\text{leave-one-out baseline}} \Bigg) - \underbrace{c_3\, w_i^2}_{\text{regularization}} \,. \tag{3}$$

[2] With fixed model parameters and deterministic inference, an LLM's forecast is a function of (X, S_i).

We show two central properties below: the first establishes the optimal prediction for *any* wager; the second establishes that the best-response wager elicits the agent's *expected score advantage*.

Theorem 3.2 (DSIC of Prediction)**.** *For any belief $\mathcal{Q}_i$ such that $\mathcal{Q}_i(\sum_{j\neq i} w_j > 0) = 1$ and any opponents' strategies satisfying the participation assumption, truthful prediction is a dominant strategy: each agent maximizes its expected net payoff by reporting her own belief about Y. Formally, for any wager profile $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$ and any $\mathbf{p}_{-i} \in \Delta(\mathcal{Y})^{M-1}$,*

$$p_i^* := \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i) \;\in\; \arg\max_{p_i \in \Delta(\mathcal{Y})} \mathbb{E}_{\mathcal{Q}_i}[\pi_i((p_i, \mathbf{p}_{-i}), \mathbf{w}, Y) \mid \mathcal{F}_i]\,,$$

and the maximizer is unique when $w_i > 0$.

Proof. See Appendix C.1. □

We note that DSIC holds under mutable beliefs: if an agent revises $\mathcal{Q}_i$ after observing others' reports, the updated posterior remains optimal. Moreover, the optimal prediction does not depend on the agent's own wager ($w_i > 0$), so a learned (even suboptimal) wager does not distort the prediction incentive. Together with the property that the best-response wager is well defined for any fixed prediction (Theorem 3.3), this decoupling enables the learning pipeline of Section 3.3.

Theorem 3.3 (Advantage–Wager Alignment)**.** *For any fixed prediction and opponents' strategies satisfying the participation assumption, the best-response wager is proportional to the agent's expected score advantage. Formally, for any belief $\mathcal{Q}_i$ held by agent i such that $\mathcal{Q}_i(\sum_{j\neq i} w_j > 0) = 1$, any $\mathbf{p} \in \Delta(\mathcal{Y})^M$, and any $\mathbf{w}_{-i} \in \mathbb{R}^{M-1}_{\geq 0}$, the best-response wager for agent i is*

$$w_i^*(p_i) = \left(\frac{A_i}{2c_3}\right)^+; \quad A_i := \mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) - b_{-i} \mid \mathcal{F}_i], \tag{4}$$

where $(x)^+ := \max(x, 0)$ and A_i denotes agent i's expected score advantage *over the baseline.*

Proof. See Appendix C.2. □

Three features follow from Theorem 3.3. *First*, agents with $A_i \leq 0$ wager zero and are automatically excluded from aggregation. *Second*, the best-response wager is well-defined for any prediction p_i, not just the optimal one, allowing the wagers to be learned separately without altering the base model's predictions (more in Section 3.3). *Third*, under the Brier score and truthful prediction, $\mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) \mid \mathcal{F}_i] = c_1 - c_2 \operatorname{Var}_{\mathcal{Q}_i}(\delta_Y \mid \mathcal{F}_i)$, so higher predictive uncertainty reduces the expected self-score; it reduces the optimal wager holding the expected baseline fixed.[3] A zero wager guarantees zero net payoff, which establishes interim individual rationality (Def. C.3). Under the participation assumption, the best-response wager for a fixed prediction guarantees non-negative expected net payoff $\mathbb{E}_{\mathcal{Q}_i}[\pi_i \mid \mathcal{F}_i] = A_i w_i^*(p_i) - c_3(w_i^*(p_i))^2 = ((A_i)^+)^2/(4c_3)$.

Equilibrium Characterization. Heterogeneous predictions can stem from two sources: differences in information and differences in beliefs. We use Theorems 3.2 and 3.3 to characterize the equilibrium in each extreme case. In practice, both sources often appear together.

Example 3.1 (Homogeneous agents with private signals)**.** *Consider M agents sharing a common prior $\mathcal{Q}$ on $(\Omega, \mathcal{F})$. Each agent observes the same question X and its own private signal S_i, so its information is $\mathcal{F}_i = \sigma(X, S_i)$. Differences in private signals can lead to different posterior beliefs about Y even under a common prior. This setting corresponds to copies of the same base LLM answering a shared question with potentially different private contexts.*

Corollary 3.4 (BNE under common prior)**.** *Suppose agents share a common prior $\mathcal{Q}$ such that $\mathcal{Q}(\sum_{j\neq i} w_j > 0) = 1$ (Example 3.1). Let $p_i^* = \mathcal{Q}(Y = \cdot \mid \mathcal{F}_i)$ (Theorem 3.2) and $w_i^*(p_i^*) = (A_i/2c_3)^+$ with $A_i = \mathbb{E}[s(p_i^*, Y) - b_{-i} \mid \mathcal{F}_i]$ (Theorem 3.3). The strategy profile $\{(p_i^*, w_i^*(p_i^*))\}_{i=1}^M$ is a Bayes–Nash equilibrium: for every agent i and every alternative strategy $(\tilde{p}_i, \tilde{w}_i)$,*

$$\mathbb{E}_{\mathcal{Q}}\big[\pi_i\big((p_i^*, \mathbf{p}_{-i}^*), (w_i^*(p_i^*), \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big] \;\geq\; \mathbb{E}_{\mathcal{Q}}\big[\pi_i\big((\tilde{p}_i, \mathbf{p}_{-i}^*), (\tilde{w}_i, \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big]\,.$$

Proof. See Appendix C.4. □

[3] Overconfident models that underestimate this variance can over-wager; calibration on realized outcomes helps (Sec. 4.5).

Example 3.2 (Heterogeneous agents)**.** *Consider M agents holding beliefs $\mathcal{Q}_1, \ldots, \mathcal{Q}_M$ that may differ. No assumptions are made regarding their relationship; the measures are permitted to be mutually singular, mutually absolutely continuous, or anywhere in between. Agent i does not know agent j's belief $\mathcal{Q}_j$. Heterogeneous predictions arise not from different information but from different perspectives on the same input. In the LLM setting, this corresponds to M distinct models—differing in architecture, training data, or fine-tuning procedure—each observing the same question with no additional context ($S_i = \emptyset$, so $\mathcal{F}_i = \sigma(X) = \mathcal{F}_j$).*

Corollary 3.5 (NE under heterogeneous beliefs)**.** *Suppose each agent i holds her own belief $\mathcal{Q}_i$ such that $\mathcal{Q}_i(\sum_{j\neq i} w_j > 0) = 1$ (Example 3.2). Let $p_i^* = \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i)$ (Theorem 3.2) and $w_i^*(p_i^*) = (A_i/2c_3)^+$ with $A_i = \mathbb{E}_{\mathcal{Q}_i}[s(p_i^*, Y) - b_{-i} \mid \mathcal{F}_i]$ (Theorem 3.3). The strategy profile $\{(p_i^*, w_i^*(p_i^*))\}_{i=1}^M$ is a Nash equilibrium: for every agent i and every alternative strategy $(\tilde{p}_i, \tilde{w}_i)$,*

$$\mathbb{E}_{\mathcal{Q}_i}\big[\pi_i\big((p_i^*, \mathbf{p}_{-i}^*), (w_i^*(p_i^*), \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big] \;\geq\; \mathbb{E}_{\mathcal{Q}_i}\big[\pi_i\big((\tilde{p}_i, \mathbf{p}_{-i}^*), (\tilde{w}_i, \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big]\,.$$

Proof. See Appendix C.5. □

The cancellation rule also admits an equilibrium in which all agents wager zero. We consider two standard aggregation functions, *linear pooling* and *logarithmic pooling* (see Appendix C.7).

3.3 Decentralized Learning of Wager Policies from Payoffs

Consider the *multiple-question* setting, where questions $x^{(t)}$ arrive sequentially and net payoffs $\pi_i^{(t)}$ are realized after each round t under the true data-generating probability measure $\mathcal{P}$. By observing these net payoffs, an agent can learn her comparative advantage, continuously updating her wager policy to approximate the best response.

We consider LLMs that form probabilistic forecasts from next-token logits. Agents have access to a privately owned model, private contexts for a question, or both. Without either private models or context, there is no privacy concern and no need for decentralization, and a central router suffices.

Learning procedure. Each model i has a forward pass f_i that maps input $(X^{(t)}, S_i^{(t)})$ to a hidden state $h_i^{(t)} := f_i(X^{(t)}, S_i^{(t)})$ where $h_i^{(t)} \in \mathcal{H}$. At each round, agent i produces (1) a prediction $p_i^{(t)} = \mathcal{Q}_i(Y^{(t)} \mid X^{(t)}, S_i^{(t)}) = g_{\phi_i}(h_i^{(t)})$ from the output layer $g_{\phi_i} : \mathcal{H} \to \Delta(\mathcal{Y})$ and (2) a wager $w_i^{(t)}$ from a learned wager network.

For open-weight models, the wager network can leverage the hidden states to produce a wager $w_i^{(t)} = g_{\theta_i}(h_i^{(t)})$ where $g_{\theta_i} \colon \mathcal{H} \to \mathbb{R}_{\geq 0}$. Otherwise, it can use the embeddings of questions and private context $Z_i^{(t)} = Z_i(X^{(t)}, S_i^{(t)}) \in \mathcal{Z}$, where Z_i is an embedding model, to produce $w_i^{(t)} = g_{\theta_i}(Z_i^{(t)})$, where $g_{\theta_i} \colon \mathcal{Z} \to \mathbb{R}_{\geq 0}$. An agent's own net payoff will be disclosed after the outcome is realized, while keeping other agents' predictions, wagers, and scores private.

Let $\mathcal{T}_i$ be the set of settled training rounds in which LLM i submitted a positive wager. For each $t \in \mathcal{T}_i$, it can recover the realized score gap from its own submitted wager and net payoff:

$$d_i^{(t)} := s(p_i^{(t)}, y^{(t)}) - b_{-i}^{(t)} = \frac{\pi_i^{(t)}}{w_i^{(t)}} + c_3 w_i^{(t)}.$$

For each $t \in \mathcal{T}_i$, hold the predictions, opponents' wagers, and realized outcome fixed. Agent i's net payoff from a candidate wager $w \geq 0$ is

$$\pi_{i,t}(w) := \pi_i\big(\mathbf{p}^{(t)}, (w, \mathbf{w}_{-i}^{(t)}), y^{(t)}\big) = w d_i^{(t)} - c_3 w^2.$$

The observed payoff is $\pi_i^{(t)} = \pi_{i,t}(w_i^{(t)})$, and the hindsight-optimal wager is $w_{i,t}^* = (d_i^{(t)})^+/(2c_3)$. Agent i trains its wager network by minimizing empirical hindsight regret,

$$\mathcal{L}_i(\theta_i) = \frac{1}{|\mathcal{T}_i|} \sum_{t \in \mathcal{T}_i} \Big[\pi_{i,t}(w_{i,t}^*) - \pi_{i,t}\big(g_{\theta_i}(z_i^{(t)})\big)\Big]\,,$$

where $z_i^{(t)}$ denote the input to agent i's wager network: $z_i^{(t)} = h_i^{(t)}$ when using hidden states and $z_i^{(t)} = Z_i^{(t)}$ when using input embeddings. For a fixed set of training records, the hindsight-optimal

payoff is independent of θ_i, so minimizing hindsight regret is equivalent to maximizing empirical net payoff. As agents update their wager policies simultaneously, this is a non-stationary multi-agent learning environment. The realized score differential $d_i^{(t)}$ serves as a noisy sample of the comparative advantage, where the randomness comes from the data-generating process and opponents' wager policies. We provide the implementation details used in our experiments in Appendix D.4.

Why learn wagers rather than improve predictions? Improving predictions requires approximating $\mathcal{P}(Y \mid X, S_i)$, a task constrained by the model's architecture, training data, and capacity. Learning wagers is comparatively lightweight: it requires only a small network operating on the hidden state or the input's embedding to learn when and to what extent the model possesses a comparative advantage, rather than how to generate better predictions. By separating wager optimization from prediction, WALLA allows smaller models to focus on domains of comparative advantage, rather than competing with larger generalist models across all tasks.

Will agents learn to stop participating? In the extreme, if one model dominates or each model perfectly identifies its best instances, only one model would have a positive wager and the round is cancelled with no aggregate prediction. In practice, non-stationarity, due to model updates, evolving query distributions, and retrieval augmentation, prevents comparative advantages from being perfectly learned, helping sustain multi-model participation and meaningful aggregation.

4 Empirical Evaluation

We evaluate whether wagers learned from agents' own net payoffs improve the aggregation of fixed LLM forecasts. We consider three settings: copies of the same LLM with private contexts, distinct LLMs given only the shared question, and distinct LLMs with private contexts.

4.1 Experiment Setup

LLM agents. We use four LLMs in the main experiments. Gemma-2-9B (Gemma2) [41] and Llama-3.1-8B (Llama3.1) [42] are general-purpose models, while Llama3-Aloe-8B (Aloe) [43] and BioMistral-7B [44] specialize in medical or biomedical tasks. Each model produces a probability distribution over the candidate outcomes from its next-token logits.

Datasets. We evaluate on four question-answering datasets, PubMedQA [45], MedMCQA [46], MMLU [47], and ARC-Challenge [48], and on BayesX, a synthetic Bayesian forecasting dataset. In BayesX, a private signal can move a concentrated prior toward uniform random, so a better-informed forecast can be less confident. PubMedQA and BayesX include question-related contexts, and thus let us vary which agents receive private context. Appendix D.1 provides dataset details.

Evaluation metrics. We use standard predictive performance metrics: area under the ROC curve (AUC) and accuracy (ACC), and expected calibration error (ECE). For BayesX, we report KL divergence (KLD) and total variation distance (TVD) to the reference posterior. We also evaluate whether aggregation weights reflect better-informed agents on each question, using per-question Brier loss to rank individual forecasts: (1) Kendall's Tau (K-Tau) measures the rank correlation between the ordering of models by their weights and by their per-question Brier scores; higher values indicate wagers correctly order models by quality. (2) Mean Reciprocal Rank (MRR) measures the average inverse rank of the best-performing model under the ordering by aggregation weights; MRR $= 1$ means the best model always receives the highest weight. (3) Dynamic Regret (D-Regret) averages the aggregate's Brier loss minus the lowest individual Brier loss on each question. Formal definitions are in Appendix D.2. Reported metrics in tables are scaled by $\times 100$; reported intervals ($\pm$) are 95% confidence intervals over five independent runs.

Baselines. We consider seven baselines spanning (1) pre-inference routers: RouterDC [2], NIRTRouter [4] and RouteLLM [3]; (2) training-free ensembles: PackLLM [10], SelfCertainty [9] and uniform averaging (UniformAvg); and (3) learning-based ensembles using LLM hidden states (StackedGen) [5]. StackedGen has access to all models' hidden states, serving as a strong centralized comparator with more information than any decentralized method. Table 1 categorizes all methods along four axes: uncertainty-awareness, decentralized aggregation, learned weights, and incentive

Table 1: Categorization of baselines. See Sec. 4.1 for discussion.

Methods	Uncertainty-Aware	Decentralized	Learning	IC / Manipulation-Resistant
Pre-inference Router [1, 2, 3, 4]	✗	✗	✓	–
Stacked Generalization [5]	✓	✗	✓	–
Uniform Averaging [15, 16, 17]	✗	✓	✗	✗
Weight by Model Confidence [9]	✓	✓	✗	✗
Weight by Context Perplexity [10]	✓	✓	✗	✗
WALLA (Ours)	✓	✓	✓	✓

compatibility. WALLA simultaneously achieves all four properties (Table 1). Detailed descriptions of baselines are in Appendix D.3.

Appendix D.4 discusses the implementation details of WALLA and Appendix D.5 provides other experiment setup details.

Table 2: Homogeneous models (Aloe) on PubMedQA with private contexts. For each question, one of $M \in \{4, 8, 12\}$ models is randomly selected to receive a relevant abstract; the rest receive none.[4]

	4					**8**					**12**				
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓
UniformAvg	70.29	71.66	2.27	-	20.80	64.85	67.20	3.12	-	25.35	63.10	66.17	4.49	-	27.30
SelfCertainty	71.78	79.23	2.67	86.76	13.62	69.31	76.92	3.33	84.73	16.50	68.05	75.66	3.65	84.07	18.12
PackLLM	78.23	85.22	10.23	88.64	8.01	75.65	81.71	9.38	87.85	11.99	73.65	78.97	7.44	87.67	14.64
RouterDC w/ ctx.	69.36$_{\pm 13.2}$	72.12$_{\pm 25.2}$	2.58$_{\pm 20.6}$	61.24$_{\pm 35.1}$	20.82$_{\pm 20.9}$	65.17$_{\pm 3.02}$	67.51$_{\pm 1.47}$	3.10$_{\pm 1.97}$	47.36$_{\pm 125}$	25.46$_{\pm 2.19}$	63.35$_{\pm 4.16}$	66.17$_{\pm 0.42}$	4.75$_{\pm 2.33}$	41.58$_{\pm 174}$	27.20$_{\pm 1.31}$
NIRTRouter w/ ctx.	70.19$_{\pm 1.32}$	71.71$_{\pm 0.63}$	2.15$_{\pm 1.17}$	52.26$_{\pm 14.9}$	20.80$_{\pm 0.01}$	64.85$_{\pm 0.00}$	67.53$_{\pm 0.00}$	3.18$_{\pm 0.00}$	33.38$_{\pm 24.1}$	25.57$_{\pm 0.00}$	63.10$_{\pm 0.01}$	66.17$_{\pm 0.00}$	4.91$_{\pm 0.19}$	24.29$_{\pm 1.11}$	27.30$_{\pm 0.00}$
RouteLLM w/ ctx.	82.32$_{\pm 0.07}$	87.48$_{\pm 0.06}$	7.24$_{\pm 0.07}$	87.81$_{\pm 0.83}$	3.51$_{\pm 0.08}$	82.22$_{\pm 0.04}$	87.50$_{\pm 0.03}$	7.21$_{\pm 0.02}$	85.77$_{\pm 1.05}$	3.54$_{\pm 0.02}$	82.52$_{\pm 0.05}$	87.39$_{\pm 0.03}$	7.17$_{\pm 0.02}$	84.15$_{\pm 0.99}$	3.55$_{\pm 0.02}$
RouteLLM w/o ctx.	69.49$_{\pm 0.39}$	72.01$_{\pm 0.19}$	2.43$_{\pm 0.70}$	52.07$_{\pm 0.11}$	20.64$_{\pm 0.00}$	65.08$_{\pm 0.38}$	67.44$_{\pm 0.19}$	3.24$_{\pm 0.49}$	33.29$_{\pm 1.65}$	25.63$_{\pm 0.01}$	63.85$_{\pm 0.35}$	65.80$_{\pm 0.17}$	4.72$_{\pm 0.14}$	28.31$_{\pm 2.59}$	27.34$_{\pm 0.02}$
StackedGen	81.99$_{\pm 0.28}$	87.76$_{\pm 0.11}$	7.53$_{\pm 0.11}$	88.01$_{\pm 0.25}$	3.29$_{\pm 0.13}$	82.65$_{\pm 0.74}$	87.44$_{\pm 0.18}$	7.48$_{\pm 0.38}$	85.25$_{\pm 1.20}$	3.49$_{\pm 0.25}$	82.26$_{\pm 0.32}$	87.36$_{\pm 0.29}$	7.29$_{\pm 0.09}$	83.76$_{\pm 1.05}$	3.66$_{\pm 0.31}$
WALLA I	82.32$_{\pm 0.02}$	87.60$_{\pm 0.00}$	7.35$_{\pm 0.01}$	88.42$_{\pm 0.24}$	3.35$_{\pm 0.00}$	82.74$_{\pm 0.10}$	87.38$_{\pm 0.03}$	7.32$_{\pm 0.09}$	84.63$_{\pm 0.61}$	3.56$_{\pm 0.04}$	82.44$_{\pm 0.04}$	87.47$_{\pm 0.00}$	7.16$_{\pm 0.01}$	83.72$_{\pm 1.06}$	3.57$_{\pm 0.00}$
WALLA II	82.24$_{\pm 0.03}$	87.63$_{\pm 0.00}$	7.44$_{\pm 0.03}$	88.41$_{\pm 0.16}$	3.37$_{\pm 0.02}$	82.74$_{\pm 0.07}$	87.38$_{\pm 0.03}$	7.31$_{\pm 0.04}$	84.97$_{\pm 1.23}$	3.56$_{\pm 0.03}$	82.45$_{\pm 0.07}$	87.47$_{\pm 0.00}$	7.17$_{\pm 0.04}$	83.76$_{\pm 0.65}$	3.57$_{\pm 0.01}$

Table 3: Heterogeneous models without context on three datasets. MedMCQA and MMLU are in-distribution; ARC-Challenge is out-of-distribution, with MMLU as the training set.

	MedMCQA (i.d.)					**MMLU (i.d.)**					**ARC-Challenge (o.o.d.)**				
Method	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
Gemma2	62.90	25.28	64.99	5.93	47.28	87.18	8.72	82.80	32.61	12.94	90.43	7.55	89.18	38.43	10.67
Aloe	79.77	3.18	59.49	14.42	15.77	80.23	8.77	53.42	3.68	21.34	80.35	8.58	51.17	-0.43	23.43
Llama3.1	78.97	2.64	51.56	4.47	15.97	81.99	2.52	40.88	-8.01	17.61	81.30	3.06	37.43	-12.20	21.01
BioMistral	47.47	17.10	37.71	-24.82	56.24	66.05	10.92	32.59	-28.27	39.95	66.61	10.95	30.68	-25.80	40.65
UniformAvg	76.45	8.10	-	0.00	21.05	85.15	5.10	-	0.00	13.87	87.39	7.40	-	0.00	14.23
SelfCertainty	74.42	4.58	79.78	43.61	22.08	86.58	1.36	88.75	63.21	11.86	89.22	2.70	91.59	66.96	10.71
PackLLM	79.00	9.93	50.77	8.85	18.93	84.94	5.13	37.79	-27.81	14.28	86.26	6.87	36.38	-21.80	14.93
RouterDC	78.95$_{\pm 5.46}$	8.77$_{\pm 0.36}$	55.65$_{\pm 1.86}$	27.65$_{\pm 1.30}$	18.35$_{\pm 8.02}$	85.28$_{\pm 0.22}$	4.75$_{\pm 0.24}$	61.71$_{\pm 22.9}$	30.16$_{\pm 20.0}$	13.61$_{\pm 0.15}$	87.48$_{\pm 0.19}$	7.22$_{\pm 0.22}$	63.21$_{\pm 30.1}$	27.50$_{\pm 25.3}$	13.98$_{\pm 0.18}$
NIRTRouter	81.15$_{\pm 0.83}$	6.99$_{\pm 0.22}$	55.64$_{\pm 1.77}$	27.67$_{\pm 1.17}$	14.56$_{\pm 0.14}$	86.91$_{\pm 5.79}$	2.15$_{\pm 0.22}$	82.72$_{\pm 0.93}$	43.72$_{\pm 4.81}$	11.51$_{\pm 5.78}$	90.17$_{\pm 2.21}$	5.12$_{\pm 1.75}$	89.18$_{\pm 0.00}$	46.38$_{\pm 0.00}$	10.65$_{\pm 3.17}$
RouteLLM	81.70$_{\pm 1.56}$	3.62$_{\pm 3.15}$	55.64$_{\pm 1.77}$	27.67$_{\pm 1.17}$	12.82$_{\pm 1.79}$	86.85$_{\pm 6.55}$	1.96$_{\pm 2.21}$	82.72$_{\pm 0.93}$	43.72$_{\pm 4.81}$	11.03$_{\pm 5.91}$	90.61$_{\pm 0.00}$	2.37$_{\pm 0.52}$	89.18$_{\pm 0.00}$	46.38$_{\pm 0.00}$	9.35$_{\pm 1.86}$
StackedGen	82.66$_{\pm 0.25}$	2.21$_{\pm 0.49}$	59.06$_{\pm 0.38}$	30.91$_{\pm 0.50}$	11.56$_{\pm 0.18}$	87.09$_{\pm 0.69}$	1.99$_{\pm 0.54}$	82.24$_{\pm 0.38}$	42.17$_{\pm 1.58}$	10.60$_{\pm 0.79}$	90.35$_{\pm 0.11}$	2.19$_{\pm 0.87}$	89.04$_{\pm 0.18}$	46.05$_{\pm 0.26}$	8.85$_{\pm 0.05}$
WALLA I	81.98$_{\pm 0.03}$	3.80$_{\pm 0.11}$	54.64$_{\pm 0.20}$	25.32$_{\pm 0.40}$	12.37$_{\pm 0.03}$	87.41$_{\pm 0.03}$	2.49$_{\pm 0.81}$	82.91$_{\pm 0.07}$	44.55$_{\pm 0.62}$	10.54$_{\pm 0.07}$	90.48$_{\pm 0.14}$	1.88$_{\pm 0.50}$	88.53$_{\pm 0.54}$	45.96$_{\pm 0.33}$	9.25$_{\pm 0.11}$
WALLA II	81.97$_{\pm 0.13}$	3.57$_{\pm 0.47}$	54.77$_{\pm 0.58}$	25.53$_{\pm 1.01}$	12.41$_{\pm 0.10}$	87.35$_{\pm 0.05}$	2.69$_{\pm 0.58}$	83.19$_{\pm 0.34}$	47.68$_{\pm 4.09}$	10.61$_{\pm 0.08}$	90.37$_{\pm 0.07}$	1.77$_{\pm 0.19}$	88.69$_{\pm 0.60}$	46.28$_{\pm 0.76}$	9.19$_{\pm 0.11}$
WALLA I log-pool	82.10$_{\pm 0.05}$	2.38$_{\pm 0.42}$	54.90$_{\pm 0.62}$	25.76$_{\pm 0.94}$	12.06$_{\pm 0.13}$	87.30$_{\pm 0.14}$	4.05$_{\pm 0.66}$	83.02$_{\pm 0.57}$	46.43$_{\pm 4.76}$	10.59$_{\pm 0.14}$	90.43$_{\pm 0.28}$	2.73$_{\pm 0.90}$	87.60$_{\pm 3.36}$	46.43$_{\pm 4.76}$	8.99$_{\pm 0.12}$
WALLA II log-pool	82.11$_{\pm 0.08}$	2.40$_{\pm 0.27}$	54.72$_{\pm 0.69}$	25.35$_{\pm 1.03}$	12.07$_{\pm 0.10}$	87.27$_{\pm 0.08}$	4.15$_{\pm 0.53}$	83.12$_{\pm 0.40}$	47.79$_{\pm 5.19}$	10.60$_{\pm 0.10}$	90.50$_{\pm 0.26}$	2.86$_{\pm 0.59}$	87.76$_{\pm 1.61}$	46.68$_{\pm 4.46}$	8.97$_{\pm 0.06}$
WALLA I calibrate	82.36$_{\pm 0.14}$	3.39$_{\pm 0.35}$	69.51$_{\pm 1.58}$	45.17$_{\pm 0.87}$	10.03$_{\pm 0.18}$	87.46$_{\pm 0.08}$	3.12$_{\pm 0.62}$	71.19$_{\pm 0.12}$	42.92$_{\pm 0.57}$	8.54$_{\pm 0.04}$	90.35$_{\pm 0.00}$	3.22$_{\pm 0.50}$	91.92$_{\pm 0.08}$	70.86$_{\pm 0.39}$	9.56$_{\pm 0.10}$

4.2 Scenario I: Homogeneous Models with Private Contexts

Participants use copies of the same LLM, as in Example 3.1. For each question, one randomly selected model receives the relevant context, while the rest receive only the question. We use Aloe on PubMedQA and Gemma2 on BayesX, with $M \in \{4, 8, 12\}$ agents.

On PubMedQA (Tables 2 and 8), WALLA achieves AUC and ACC comparable to StackedGen and RouteLLM (with context), both of which require centralized access to models' hidden states or private inputs, while outperforming all other baselines. The wager networks learn from payoff signals and hidden states to identify when the model has received relevant context, wagering high when it holds a comparative advantage and near zero otherwise (Fig. 2 and Fig. 6 in Appendix). As the pool grows from 4 to 12, learning-free methods degrade substantially (e.g., UniformAvg AUC drops from 70.3 to 63.1), while WALLA remains robust.

On BayesX (Table 7 in Appendix), WALLA outperforms all baselines except RouteLLM with context and StackedGen, both of which easily recognize agents with private context in inputs. Without access to context, RouteLLM drops to uniform averaging. SelfCertainty performs poorly because observing the signal makes the posterior more faithful but closer to uniform, inverting its confidence estimate.

[4]Aloe achieves AUC=82.39, ACC=87.57, ECE=7.38 with abstract and AUC=60.65, ACC=63.10, ECE=7.83 without.

4.3 Scenario II: Heterogeneous Models without Context

Participants receive the same question and use different LLMs, as in Example 3.2. We train a wager network for each LLM on the MedMCQA and MMLU datasets, and evaluate in-distribution on MedMCQA and MMLU, and out-of-distribution on ARC-Challenge. Results with four participants are in Tables 3 and 11; two-participant results are in Tables 9–10 in Appendix.

Table 3 shows that WALLA performs comparably to the centralized StackedGen across all three datasets. Aggregation can improve over individual models, especially on MedMCQA; on MMLU and ARC-Challenge, the aggregated prediction matches the performance of the strongest individual model (Gemma2). The wager networks here learn a different mapping than in Scenario I: rather than learning context assignment, they learn each model's domain-dependent comparative advantage from the hidden state and payoff signals.

Learning-free methods are notably less robust to pool composition. On MedMCQA, weaker models (Gemma2, BioMistral) cause performance drops for UniformAvg, PackLLM, and SelfCertainty, because miscalibrated confidence in weaker models can inflate their aggregation weight. This illustrates another limitation of confidence-based aggregation that holds even without strategic manipulation.

Table 4: Four heterogeneous models with private contexts on PubMedQA. For each question, randomly assign a model with a relevant context (left columns) or two models, each with a relevant or irrelevant context (right columns).

	One Relevant Context						**One Relevant Context & One Irrelevant Context**					
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
UniformAvg	73.43	76.46	11.51	-	0.00	24.28	72.76	74.60	11.32	-	0.00	27.66
SelfCertainty	69.84	77.25	3.13	86.11	46.30	21.60	66.58	71.40	10.89	83.61	39.68	31.20
PackLLM	77.85	83.60	15.38	66.42	12.32	18.05	71.95	77.02	12.86	58.23	5.39	26.41
RouterDC w/ ctx.	73.33$_{\pm 0.63}$	76.55$_{\pm 1.26}$	11.58$_{\pm 1.91}$	54.66$_{\pm 37.9}$	10.64$_{\pm 64.7}$	24.23$_{\pm 0.34}$	72.53$_{\pm 1.44}$	74.90$_{\pm 2.94}$	11.74$_{\pm 3.55}$	48.77$_{\pm 6.51}$	-2.19$_{\pm 11.9}$	27.65$_{\pm 0.52}$
NIRTRouter w/ ctx.	73.20$_{\pm 0.58}$	76.92$_{\pm 0.84}$	12.04$_{\pm 7.80}$	43.90$_{\pm 50.1}$	-7.29$_{\pm 152}$	24.15$_{\pm 3.73}$	72.53$_{\pm 0.34}$	75.40$_{\pm 0.00}$	12.27$_{\pm 4.23}$	43.15$_{\pm 69.6}$	-7.18$_{\pm 130}$	27.48$_{\pm 2.74}$
RouteLLM w/ ctx.	74.46$_{\pm 0.57}$	79.11$_{\pm 0.59}$	5.66$_{\pm 1.40}$	60.84$_{\pm 2.83}$	14.24$_{\pm 2.72}$	19.18$_{\pm 0.32}$	73.19$_{\pm 0.72}$	77.89$_{\pm 0.64}$	5.97$_{\pm 1.35}$	62.18$_{\pm 2.94}$	15.00$_{\pm 2.44}$	22.28$_{\pm 0.32}$
RouteLLM w/o ctx.	72.15$_{\pm 5.90}$	76.27$_{\pm 0.21}$	8.26$_{\pm 1.51}$	55.46$_{\pm 56.7}$	11.11$_{\pm 36.0}$	23.09$_{\pm 3.05}$	68.97$_{\pm 12.1}$	75.81$_{\pm 3.57}$	5.91$_{\pm 3.73}$	62.59$_{\pm 7.84}$	15.84$_{\pm 1.12}$	25.20$_{\pm 5.89}$
StackedGen	81.17$_{\pm 0.30}$	84.82$_{\pm 0.18}$	7.55$_{\pm 0.43}$	71.74$_{\pm 0.14}$	26.28$_{\pm 0.84}$	12.34$_{\pm 0.06}$	80.30$_{\pm 0.56}$	84.38$_{\pm 1.72}$	6.57$_{\pm 1.81}$	71.52$_{\pm 0.81}$	24.06$_{\pm 0.39}$	14.07$_{\pm 0.93}$
WALLA I	78.64$_{\pm 0.28}$	82.90$_{\pm 0.20}$	6.29$_{\pm 0.32}$	70.15$_{\pm 0.35}$	23.63$_{\pm 0.51}$	14.61$_{\pm 0.03}$	78.91$_{\pm 0.23}$	80.75$_{\pm 0.15}$	4.78$_{\pm 0.23}$	68.74$_{\pm 0.13}$	23.19$_{\pm 0.16}$	17.28$_{\pm 0.07}$
WALLA II	78.62$_{\pm 0.14}$	81.71$_{\pm 1.91}$	4.73$_{\pm 2.39}$	69.75$_{\pm 1.03}$	23.85$_{\pm 0.72}$	15.28$_{\pm 1.10}$	77.89$_{\pm 0.50}$	80.61$_{\pm 0.14}$	3.85$_{\pm 0.35}$	69.51$_{\pm 0.31}$	23.32$_{\pm 0.18}$	17.66$_{\pm 0.06}$
WALLA I on emb.	78.25$_{\pm 0.33}$	80.63$_{\pm 0.29}$	3.92$_{\pm 0.73}$	67.38$_{\pm 1.15}$	22.47$_{\pm 0.93}$	16.12$_{\pm 0.17}$	71.70$_{\pm 1.00}$	74.85$_{\pm 0.80}$	10.90$_{\pm 0.96}$	45.66$_{\pm 0.97}$	-1.46$_{\pm 2.26}$	27.27$_{\pm 0.42}$
WALLA I log-pool	78.56$_{\pm 0.37}$	82.95$_{\pm 0.19}$	5.95$_{\pm 0.20}$	70.22$_{\pm 0.14}$	23.56$_{\pm 0.32}$	14.48$_{\pm 0.02}$	77.94$_{\pm 0.37}$	80.69$_{\pm 0.07}$	3.29$_{\pm 0.14}$	68.88$_{\pm 0.27}$	23.30$_{\pm 0.21}$	17.47$_{\pm 0.07}$
WALLA I calibrate	87.15$_{\pm 0.14}$	83.76$_{\pm 0.31}$	3.46$_{\pm 0.38}$	75.78$_{\pm 0.23}$	27.76$_{\pm 0.92}$	11.67$_{\pm 0.14}$	84.85$_{\pm 0.14}$	82.05$_{\pm 0.07}$	3.69$_{\pm 0.15}$	73.46$_{\pm 0.34}$	23.43$_{\pm 0.48}$	13.61$_{\pm 0.08}$

4.4 Scenario III: Heterogeneous Models with Private Contexts

We use all four LLMs evaluated on PubMedQA. We consider two settings: (1) one randomly selected model receives the relevant context, as in Scenario I; and (2) one randomly selected model receives the relevant context while another randomly selected model receives an irrelevant context (an abstract randomly drawn from the test set). The results are provided in Tables 4 and 12.

StackedGen and WALLA stand out as the best-performing methods. RouteLLM with context and PackLLM are competitive when only one relevant context is present as context availability is easy to detect from input tokens, but their advantage diminishes when an irrelevant context is also introduced, as the method must now distinguish relevant from misleading context. WALLA on question and context embeddings only remains effective. It matches or outperforms centralized routers in prediction performance, but falls short of WALLA on hidden states and StackedGen, especially in ranking fidelity (KTau/MRR). This is consistent with embeddings capturing whether relevant context is present, but not how the model processed it.

The wager networks in this scenario face the most demanding learning among the three: they must jointly assess the model's intrinsic competence, the relevance of any received context, and how much that context improves the model's comparative advantage over the other participants in the pool.

4.5 More Analysis on Robustness, Advantage-Aligned Wagering, and Calibration

Robustness to mechanism variants. We evaluate WALLA II and log pooling as mechanism variants (Tables 2–4). Both lead to results similar to WALLA I with linear pooling. This indicates that the performance of WALLA is not driven by a particular choice of baseline scoring or aggregation rule. The similarity between WALLA I and WALLA II is also consistent with theory: both variants preserve advantage–wager alignment, differing mainly in the normality and no-arbitrage tradeoff. Likewise, the similarity between linear pooling and log pooling suggests that the learned wagers, rather than the aggregation rule, are the primary driver of the aggregated prediction quality.

Generalization to larger LLMs. WALLA applies to models of any scale: training the wager network on last-layer hidden states or input embeddings incurs minimal computational overhead regardless of model size. To demonstrate scalability, we conducted a new experiment adding Llama 3.1-70B to our heterogeneous model pool with private contexts, which leads to similar results as shown in Appendix Table 13 (discussion in Appendix D.6).

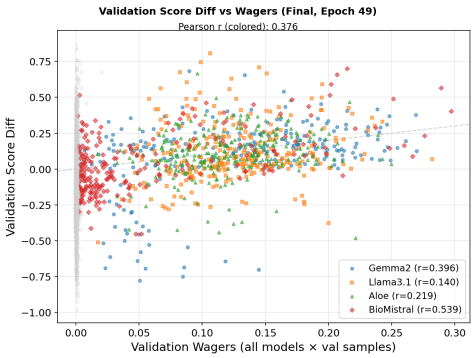


Figure 1: Score-wager alignment for WALLA I on Scenario III with heterogeneous models and private contexts. Colored dots highlight models with context. Pearson correlations are reported at the top of the panel. The plots for other settings are in Fig. 2 in the appendix.

Advantage–wager alignment. Figs 1 and 2 show that learned wagers are generally positively correlated with realized comparative advantages in Scenarios I and III, providing evidence that models learn to wager in proportion to their expected score advantage. As in Scenario II, specialized models dominate over the entire datasets; dominated models tend to wager close to zero. This leads to lower empirical advantage–wager correlation as the baseline scores become hard to estimate. We conduct further experiments to verify that the hidden states contain sufficient information to separately predict both the model's own score and the baseline score; we find that the difference between these predictions aligns closely with the learned wager (see Figs 3–5 in the Appendix).

Calibration. The mechanism incentivizes participants to calibrate their predictions: even when a model's belief $\mathcal{Q}_i \neq \mathcal{P}$, calibration aligns expected scores better with realized scores, so that the learned wager better reflects *actual* comparative advantage. We conduct calibration via temperature scaling [49], which predicts an appropriate temperature conditional on the last-layer hidden state. In Table 18 (Appendix), we show that when one model is calibrated (Gemma2) while the others remain unchanged, the calibrated model receives improved net payoffs because its expected score advantage more accurately reflects its realized comparative advantage. This suggests that rational participants under WALLA have an incentive to calibrate, thus discouraging both overconfidence and underconfidence. We also report WALLA I results when all participants are calibrated for Scenarios II and III (Tables 3, 4, and 9–12). The results show that calibrated aggregation can substantially improve prediction performance on in-distribution tasks, while maintaining similar performance on out-of-distribution tasks.

5 Conclusion

We introduced WALLA, a family of wagering mechanisms that use each LLM's advantage-aligned wager to weight its probabilistic forecast. The proposed mechanism yields three desirable properties: (1) DSIC of predictions under general belief structures: under the participation assumption, truthful reporting maximizes expected payoff for any fixed wager, (2) advantage–wager alignment: the best-response wager is proportional to its expected score advantage over a leave-one-out baseline, and (3) wager optimization for any fixed prediction: each LLM can train its wager policy from its own submitted wagers and realized payoffs while keeping its predictor fixed.

Across four question-answering benchmarks, WALLA achieves predictive performance comparable to centralized baselines with access to model internals or private inputs. On BayesX, WALLA outperforms confidence-based weighting in settings where relevant private evidence makes forecasts informed but less confident. Future work could test whether LLMs can use their own past questions, wagers, and payoffs as in-context examples to estimate their expected score advantage and choose wagers on new questions. Another direction is to extend WALLA to more complex collaboration settings, for example, involving multi-step reasoning beyond aggregating probabilistic predictions.

6 Acknowledgments

We thank the anonymous reviewers for constructive feedback. This work was supported in part by Rutgers SAS Research Grant in Academic Themes.

References

[1] Keming Lu, Hongyi Yuan, Runji Lin, Junyang Lin, Zheng Yuan, Chang Zhou, and Jingren Zhou. Routing to the expert: Efficient reward-guided ensemble of large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1964–1974, 2024.

[2] Shuhao Chen, Weisen Jiang, Baijiong Lin, James Kwok, and Yu Zhang. RouterDC: Query-based router by dual contrastive learning for assembling large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[3] Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs from preference data. In *The Thirteenth International Conference on Learning Representations*, 2025.

[4] Wei Song, Zhenya Huang, Cheng Cheng, Weibo Gao, Bihan Xu, GuanHao Zhao, Fei Wang, and Runze Wu. IRT-router: Effective and interpretable multi-LLM routing via item response theory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

[5] David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

[6] Zhen Ling Teo, Liyuan Jin, Nan Liu, Yong Liu, Rick Siow Mong Goh, and Daniel Shu Wei Ting. Federated machine learning in healthcare: A systematic review on clinical applications and technical architecture. *Cell Reports Medicine*, 5(2):101419, 2024.

[7] Yueyue Shi, Hengjie Song, and Jun Xu. Responsible and effective federated learning in financial services: A comprehensive survey. pages 4229–4236, 12 2023. doi: 10.1109/CDC49753.2023.10384119.

[8] Justin D. Harris and Bo Waggoner. Decentralized and Collaborative AI on Blockchain . In *2019 IEEE International Conference on Blockchain (Blockchain)*, pages 368–375, 2019.

[9] Zhewei Kang, Xuandong Zhao, and Dawn Song. Scalable best-of-n selection for large language models via self-certainty. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[10] Costas Mavromatis, Petros Karypis, and George Karypis. Pack of LLMs: Model fusion at test-time via perplexity optimization. In *First Conference on Language Modeling*, 2024.

[11] Nicolas S. Lambert, John Langford, Jennifer Wortman Vaughan, Yiling Chen, Daniel M. Reeves, Yoav Shoham, and David M. Pennock. An axiomatic characterization of wagering mechanisms. *Journal of Economic Theory*, 156:389–416, 2015.

[12] Yiling Chen, Nikhil R. Devanur, David M. Pennock, and Jennifer Wortman Vaughan. Removing arbitrage from wagering mechanisms. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, page 377–394, 2014.

[13] Rupert Freeman, David M. Pennock, and Jennifer Wortman Vaughan. The double clinching auction for wagering. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, page 43–60, 2017.

[14] Zhijun Chen, Jingzheng Li, Pengpeng Chen, Zhuoran Li, Kai Sun, Yuankai Luo, Qianren Mao, Ming Li, Likang Xiao, Dingqi Yang, Yikun Ban, Hailong Sun, and Philip S. Yu. Harnessing multiple large language models: A survey on LLM ensemble. *arXiv preprint arXiv:2502.18036*, 2025.

[15] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023.

[16] Junyou Li, Qin Zhang, Yangbin Yu, Qiang Fu, and Deheng Ye. More agents is all you need. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856.

[17] Yao-Ching Yu, Chun Chih Kuo, Ye Ziqi, Chang Yucheng, and Yueh-Se Li. Breaking the ceiling of the LLM community by treating token generation as a classification for ensembling. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1826–1839, 2024.

[18] Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*, 2023.

[19] Hyeong Kyu Choi, Jerry Zhu, and Sharon Li. Debate or vote: Which yields better decisions in multi-agent large language models? In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[20] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Nova Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[21] Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, 2023.

[22] Zeguan Xiao, Diyang Dou, Boya Xiong, Yun Chen, and Guanhua Chen. Enhancing uncertainty estimation in LLMs with expectation of aggregated internal belief. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(40):34043–34051, Mar. 2026.

[23] Tal Shnitzer, Anthony Ou, Mírian Silva, Kate Soule, Yuekai Sun, Justin Solomon, Neil Thompson, and Mikhail Yurochkin. Large language model routing with benchmark datasets. In *First Conference on Language Modeling*, 2024.

[24] Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178, 2023.

[25] Philipp Schoenegger, Indre Tuminauskaite, Peter S. Park, Rafael Valdece Sousa Bastos, and Philip E. Tetlock. Wisdom of the silicon crowd: LLM ensemble prediction capabilities rival human crowd accuracy. *Science Advances*, 10(45):eadp1528, 2024.

[26] Paul Dütting, Vahab Mirrokni, Renato Paes Leme, Haifeng Xu, and Song Zuo. Mechanism design for large language models. In *Proceedings of the ACM Web Conference 2024*, page 144–155, 2024.

[27] Avinava Dubey, Zhe Feng, Rahul Kidambi, Aranyak Mehta, and Di Wang. Auctions with LLM summaries. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, page 713–722, 2024.

[28] Ermis Soumalias, Michael Curry, and Sven Seuken. Truthful aggregation of LLMs with an application to online advertising. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[29] Dirk Bergemann, Marek Bojko, Paul Duetting, Renato Paes Leme, Haifeng Xu, and Song Zuo. Data-driven mechanism design: Jointly eliciting preferences and information. In *Proceedings of the 26th ACM Conference on Economics and Computation*, page 507, 2025.

[30] Jacob Abernethy and Rafael M. Frongillo. A collaborative mechanism for crowdsourcing prediction problems. In *Advances in Neural Information Processing Systems*, 2011.

[31] Alina Beygelzimer, John Langford, and David M. Pennock. Learning performance of prediction markets with kelly bettors. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 3*, page 1317–1318, 2012.

[32] Rafael Frongillo, Nicholás Della Penna, and Mark D Reid. Interpreting prediction markets: a stochastic approach. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

[33] Jacob Abernethy, Yiling Chen, and Jennifer Wortman Vaughan. Efficient market making via convex optimization, and a connection to online learning. *ACM Transactions on Economics and Computation*, 1(2), 2013.

[34] Jinli Hu and Amos Storkey. Multi-period trading prediction markets with connections to machine learning. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1773–1781, 2014.

[35] Nicolas S. Lambert, John Langford, Jennifer Wortman, Yiling Chen, Daniel Reeves, Yoav Shoham, and David M. Pennock. Self-financed wagering mechanisms for forecasting. In *Proceedings of the 9th ACM Conference on Electronic Commerce*, page 170–179, 2008.

[36] Aitazaz Ali Raja, Pierre Pinson, Jalal Kazempour, and Sergio Grammatico. A market for trading forecasts: A wagering mechanism. *International Journal of Forecasting*, 40(1):142–159, 2024.

[37] Rupert Freeman, David M. Pennock, Chara Podimata, and Jennifer Wortman Vaughan. No-regret and incentive-compatible online learning. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

[38] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78:1–3, 1950.

[39] Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

[40] Jennifer A Hoeting, David Madigan, Adrian E Raftery, and Chris T Volinsky. Bayesian model averaging: a tutorial. *Statistical science*, 14(4):382–417, 1999.

[41] Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

[42] Aaron Grattafiori et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[43] Ashwin Kumar Gururajan, Enrique Lopez-Cuena, Jordi Bayarri-Planas, Adrian Tormos, Daniel Hinjos, Pablo Bernabeu-Perez, Anna Arias-Duart, Pablo Agustin Martin-Torres, Lucia Urcelay-Ganzabal, Marta Gonzalez-Mallo, Sergio Alvarez-Napagao, Eduard Ayguadé-Parra, Ulises Cortés, and Dario Garcia-Gasulla. Aloe: A family of fine-tuned open healthcare LLMs. *arXiv preprint arXiv:2405.01886*, 2024.

[44] Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. BioMistral: A collection of open-source pretrained large language models for medical domains. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 5848–5864, 2024.

[45] Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, 2019.

[46] Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. MedMCQA: A large-scale multi-subject multi-choice dataset for medical domain question answering. In *Proceedings of the Conference on Health, Inference, and Learning*, volume 174, pages 248–260, 2022.

[47] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

[48] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

[49] Johnathan Xie, Annie S Chen, Yoonho Lee, Eric Mitchell, and Chelsea Finn. Calibrating language models with adaptive temperature scaling. In *EMNLP*, 2024.

[50] Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8(6):121–164, 2012. doi: 10.4086/toc.2012.v008a006.

[51] Christian Genest. A Characterization Theorem for Externally Bayesian Groups. *The Annals of Statistics*, 12(3):1100 – 1105, 1984.

[52] Pengcheng He, Jianfeng Gao, and Weizhu Chen. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*, 2023.

[53] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019.

A Desirable Properties of Wagering Mechanisms

Definition A.1 (Incentive Compatibility (IC))**.** *A wagering mechanism is incentive-compatible if all players maximize their expected net payoff when predicting their true beliefs. Formally, $\forall i \in \mathcal{M}$, $q_i \in \Delta(\mathcal{Y})$ which reflects agent i's true belief, $p_i \in \Delta(\mathcal{Y})$ where $q_i \neq p_i$, $\mathbf{p}_{-i} \in \Delta(\mathcal{Y})^{M-1}$, and $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$ with $w_i > 0$,*

$$\mathbb{E}_{Y\sim q_i}[\pi_i((q_i, \mathbf{p}_{-i}), \mathbf{w}, Y)] > \mathbb{E}_{Y\sim q_i}[\pi_i((p_i, \mathbf{p}_{-i}), \mathbf{w}, Y)].$$

Definition A.2 (Individual Rationality (IR))**.** *Every agent prefers participating to not participating in the mechanism (i.e., the expected net payoff of participating is nonnegative). Formally, $\forall i \in \mathcal{M}$, $q_i \in \Delta(\mathcal{Y})$ which reflects agent i's true belief and $w_i \in \mathbb{R}_{\geq 0}$, there exists some $p_i \in \Delta(\mathcal{Y})$ such that for all $\mathbf{p}_{-i} \in \Delta(\mathcal{Y})^{M-1}$ and $\mathbf{w}_{-i} \in \mathbb{R}^{M-1}_{\geq 0}$,*

$$\mathbb{E}_{Y\sim q_i}[\pi_i((p_i, \mathbf{p}_{-i}), (w_i, \mathbf{w}_{-i}), Y)] \geq 0.$$

Definition A.3 (Weak Budget Balance)**.** *The mechanism does not lose money but can potentially make a profit. Formally, $\forall \mathbf{p} \in \Delta(\mathcal{Y})^M, \mathbf{w} \in \mathbb{R}^M_{\geq 0}$ and $y \in \mathcal{Y}$,*

$$\sum_{i=1}^{M} \pi_i(\mathbf{p}, \mathbf{w}, y) \leq 0.$$

Definition A.4 (Normality)**.** *If from agent i's perspective the prediction of another agent j improves, then agent i's expected net payoff decreases. Formally, for every $i \neq j \in \mathcal{M}$, $q_i \in \Delta(\mathcal{Y})$ which reflects agent i's true belief, $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$ and $\mathbf{p} \in \Delta(\mathcal{Y})^M$, defining $\mathbf{p}'$ by $p'_j = q_i$ and $p'_k = p_k$ for $k \neq j$,*

$$\mathbb{E}_{Y\sim q_i}[\pi_i(\mathbf{p}, \mathbf{w}, Y)] \geq \mathbb{E}_{Y\sim q_i}[\pi_i(\mathbf{p}', \mathbf{w}, Y)],$$

with strict inequality whenever $p_j \neq q_i$, $w_i > 0$ and $w_j > 0$.

Definition A.5 (No-Arbitrage [12])**.** *No agent can guarantee a strictly positive payoff regardless of the realized outcome. Formally, a mechanism is arbitrage-free if there exists no agent $i \in \mathcal{M}$, prediction profile $\mathbf{p} \in \Delta(\mathcal{Y})^M$, and $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$ such that $\pi_i(\mathbf{p}, \mathbf{w}, y) > 0$ for all $y \in \mathcal{Y}$.*

B Theoretical Properties of WSWM Using a Scaled Linear Score

Consider the repeated setting where question $x^{(t)}$ arrives at time t with outcome $y^{(t)}$. Suppose all participants are non-strategic: reporting the predictions $p_i^{(t)}$ truthfully and wagering the total cumulative payoff, namely, their budget, denoted as $B_i^{(t-1)}$. Assume $B_i^{(0)} \geq 0$, $\sum_i B_i^{(0)} = 1$, and $p^{(t)}_{\text{agg}}[y^{(t)}] > 0$ at every round.

Consider a WSWM that adopts a scaled linear score $s_{\text{lin}} \in \mathcal{S} : (\Delta(\mathcal{Y}) \times \mathbb{R}_{\geq 0})^M$ such that

$$s_i^{(t)} = \frac{p_i^{(t)}[y]}{\sum_{j=1}^{M} w_j^{(t)} p_j^{(t)}[y]}$$

and the linear pooling (i.e., $p^{(t)}_{\text{agg}} = \frac{\sum_{i=1}^{M} w_i^{(t)} p_i^{(t)}}{\sum_{i=1}^{M} w_i^{(t)}}$). We first show that the net payoff equals the negative cross-entropy gradient scaled by the agent's wager. We then show the mechanism exactly recovers Bayesian Model Averaging (BMA) [40].

For notation simplicity, we remove the timestamp and simplify $p_{\text{agg}}[y]$ as p_{agg} and $p_i[y]$ as p_i when the context is clear.

B.1 WSWM Recovers the Scaled Negative Gradient of Cross-Entropy Loss

We connect this mechanism to centralized supervised learning by analyzing the cross-entropy loss of the aggregated prediction.

Theorem B.1. *Suppose WSWM utilizes the scoring function s_{lin}. The net payoff function, defined as $\pi_i = w_i \left(s_i - \frac{\sum_{k=1}^{M} s_k w_k}{\sum_{k=1}^{M} w_k} \right)$, satisfies $\pi_i = -w_i \frac{\partial L}{\partial w_i}$ where $L = -\log p_{agg}$, the cross-entropy loss.*

Proof. Expanding the logarithm of the quotient yields the loss function:

$$L = -\log\left(\sum_{i=1}^{M} w_i p_i\right) + \log\left(\sum_{i=1}^{M} w_i\right).$$

Taking the partial derivative of L with respect to a specific weight w_i gives:

$$\frac{\partial L}{\partial w_i} = -\frac{p_i}{\sum_{k=1}^{M} w_k p_k} + \frac{1}{\sum_{k=1}^{M} w_k}.$$

Substituting the given scoring rule s_k into $\sum_{k=1}^{M} s_k w_k$ gives:

$$\begin{aligned}\sum_{k=1}^{M} s_k w_k &= \sum_{k=1}^{M}\left(\frac{p_k}{\sum_{j=1}^{M} w_j p_j}\right) w_k \\ &= \frac{\sum_{k=1}^{M} w_k p_k}{\sum_{j=1}^{M} w_j p_j} \\ &= 1.\end{aligned}$$

We substitute this result back into the net payoff function π_i:

$$\begin{aligned}\pi_i &= w_i\left(s_i - \frac{1}{\sum_{k=1}^{M} w_k}\right) \\ &= w_i\left(\frac{p_i}{\sum_{j=1}^{M} w_j p_j} - \frac{1}{\sum_{k=1}^{M} w_k}\right) \\ &= -w_i\left(-\frac{p_i}{\sum_{j=1}^{M} w_j p_j} + \frac{1}{\sum_{k=1}^{M} w_k}\right).\end{aligned}$$

We observe that the expression inside the parentheses is exactly the partial derivative $\frac{\partial L}{\partial w_i}$ derived previously. Therefore, by direct substitution:

$$\pi_i = -w_i \frac{\partial L}{\partial w_i}.$$

□

Remark B.1. *This relationship demonstrates that the budget update* $B_i^{(t)} = w_i^{(t)} + \pi_i^{(t)} = w_i^{(t)}\left(1 - \frac{\partial L^{(t)}}{\partial w_i^{(t)}}\right) = w_i^{(t)} - w_i^{(t)}\frac{\partial L^{(t)}}{\partial w_i^{(t)}}$ *has the form of a Multiplicative Weights Update (MWU) algorithm [50]. It can be interpreted as executing decentralized gradient descent where the learning rate is scaled dynamically by the current weight. It also mirrors the exponentiated gradient step* $B_i^{(t)} \approx w_i^{(t)} e^{-\frac{\partial L^{(t)}}{\partial w_i^{(t)}}}$ *under a first-order Taylor approximation.*

B.2 Equivalence to Bayesian Model Averaging

We now show that the WSWM under this setting is equivalent to Bayesian model averaging (BMA) [40]. We start with the review of BMA.

Definition B.2 (Bayesian Model Averaging (BMA))**.** *Let* $\theta_1, \ldots, \theta_M$ *be discrete parameters for the* M *models. Given a history of observations* $D^{(1:t-1)} = \{(x^{(1)}, y^{(1)}), \ldots, (x^{(t-1)}, y^{(t-1)})\}$*, the posterior probability of parameter* θ_i *is* $p(\theta_i|D^{(1:t-1)})$*. We condition on the questions and update model weights only after observing their labels. Each model assigns probability* $p(y^{(t)} \mid x^{(t)}, \theta_i)$ *to the observed label, independently of previous question–answer pairs. Thus* $p(\theta_i|D^{(1:t-1)}) \propto p(\theta_i)\prod_{\tau=1}^{t-1} p(y^{(\tau)}|x^{(\tau)}, \theta_i)$*. The BMA predictive probability for the true label* $y^{(t)}$ *is:*

$$p_{BMA}(y^{(t)}|x^{(t)}, D^{(1:t-1)}) = \sum_{i=1}^{M} p(y^{(t)}|x^{(t)}, \theta_i) p(\theta_i|D^{(1:t-1)}).$$

The mechanism recovers the model posterior defined by these conditional likelihoods; this does not require the model mixture to equal the true data-generating distribution.

Theorem B.3. *Under the normalized-budget setting above and the conditional-likelihood formulation in Definition B.2, suppose each model reports $p_i^{(t)}[y] = p(y|x^{(t)}, \theta_i)$. If the initial budgets are set to the prior probabilities $B_i^{(0)} = p(\theta_i)$, the aggregated prediction satisfies $p_{agg}^{(t)} = p_{BMA}(y^{(t)}|x^{(t)}, D^{(1:t-1)})$ for all $t \geq 1$, and the budgets track the exact posterior $B_i^{(t)} = p(\theta_i|D^{(1:t)})$.*

Proof. We proceed by induction. The base case holds by initialization $B_i^{(0)} = p(\theta_i)$. Assume the inductive hypothesis $B_i^{(t-1)} = p(\theta_i|D^{(1:t-1)})$. Because $w_i^{(t)} = B_i^{(t-1)}$, substituting the hypothesis into the aggregated prediction yields $p_{\text{agg}}^{(t)} = \sum_{j=1}^{M} p(\theta_j|D^{(1:t-1)})p(y^{(t)}|x^{(t)}, \theta_j)$, which exactly matches Definition B.2.

The inductive hypothesis gives $\sum_i w_i^{(t)} = 1$, so $\pi_i^{(t)} = w_i^{(t)}(s_i^{(t)} - 1)$. Hence the budget update is:

$$
\begin{aligned}
B_i^{(t)} = w_i^{(t)} s_i^{(t)} \quad &= w_i^{(t)} \frac{p_i^{(t)}}{\sum_{j=1}^{M} w_j^{(t)} p_j^{(t)}} \quad = B_i^{(t-1)} \frac{p_i^{(t)}}{p_{\text{agg}}^{(t)}} \\
&= \frac{p(\theta_i|D^{(1:t-1)})p(y^{(t)}|x^{(t)}, \theta_i)}{p_{\text{BMA}}(y^{(t)}|x^{(t)}, D^{(1:t-1)})} \\
&= \frac{p(\theta_i|D^{(1:t-1)})p(y^{(t)}|x^{(t)}, \theta_i)}{\sum_j p(\theta_j|D^{(1:t-1)})p(y^{(t)}|x^{(t)}, \theta_j)} \\
&= p(\theta_i|D^{(1:t)}) \quad \text{(conditional-likelihood Bayes update).}
\end{aligned}
$$

This completes the proof. □

C Deferred Content from Sec. 3

C.1 Proof of DSIC of Prediction

Theorem C.1 (DSIC of Prediction)**.** *For any belief $\mathcal{Q}_i$ such that $\mathcal{Q}_i(\sum_{j\neq i} w_j > 0) = 1$ and any opponents' strategies satisfying the participation assumption, truthful prediction is a dominant strategy: each agent maximizes its expected net payoff by reporting her own belief about Y. Formally, for any wager profile $\mathbf{w} \in \mathbb{R}_{\geq 0}^{M}$ and any $\mathbf{p}_{-i} \in \Delta(\mathcal{Y})^{M-1}$,*

$$
p_i^* := \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i) \;\in\; \arg\max_{p_i \in \Delta(\mathcal{Y})} \mathbb{E}_{\mathcal{Q}_i}[\pi_i((p_i, \mathbf{p}_{-i}), \mathbf{w}, Y) \mid \mathcal{F}_i]\,,
$$

and the maximizer is unique when $w_i > 0$.

Proof. Under the participation assumption preceding (3) and for $w_i > 0$, the round runs almost surely. From the decomposition (3), since b_{-i} does not depend on p_i, the optimization reduces to

$$
p_i^* \in \arg\max_{p_i \in \Delta(\mathcal{Y})} \mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) \mid \mathcal{F}_i].
$$

By strict properness of s, this is uniquely attained at $p_i^* := \mathbb{E}_{\mathcal{Q}_i}[\delta_Y \mid \mathcal{F}_i] = \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i)$, the agent's belief about Y given her observation of the question X and private signal S_i. When $w_i = 0$, the payoff is identically zero regardless of the prediction. □

C.2 Proof of Advantage–Wager Alignment

Theorem C.2 (Advantage–Wager Alignment)**.** *For any fixed prediction and opponents' strategies satisfying the participation assumption, the best-response wager is proportional to the agent's expected score advantage. Formally, for any belief $\mathcal{Q}_i$ held by agent i such that $\mathcal{Q}_i(\sum_{j\neq i} w_j > 0) = 1$, any $\mathbf{p} \in \Delta(\mathcal{Y})^M$, and any $\mathbf{w}_{-i} \in \mathbb{R}_{\geq 0}^{M-1}$, the best-response wager for agent i is*

$$
w_i^*(p_i) = \left(\frac{A_i}{2c_3}\right)^+; \quad A_i := \mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) - b_{-i} \mid \mathcal{F}_i], \tag{5}
$$

where $(x)^+ := \max(x, 0)$ and A_i denotes agent i's expected score advantage *over the baseline.*

Proof. For any fixed p_i, $\mathbf{p}_{-i}$, and $\mathbf{w}_{-i}$, the expected payoff (3) as a function of $w_i \geq 0$ is $\mathbb{E}[\pi_i \mid \mathcal{F}_i] = w_i\, A_i - c_3\, w_i^2$. Since $\frac{\partial^2}{\partial w_i^2}\mathbb{E}[\pi_i \mid \mathcal{F}_i] = -2c_3 < 0$, this is strictly concave. The unique unconstrained maximizer is $w_i = A_i/(2c_3)$, and the constrained maximizer over $w_i \geq 0$ is $w_i^* = (A_i/(2c_3))^+$. □

C.3 Definition of Interim Individual Rationality

Definition C.3 (Interim Individual Rationality)**.** *The expected net payoff of participating upon receiving the question and private signal is nonnegative. Formally, $\forall i \in \mathcal{M}$, any $\mathcal{Q}_i$ and $\mathcal{F}_i$, there exist some $p_i \in \Delta(\mathcal{Y})$ and $w_i \in \mathbb{R}_{\geq 0}$ such that for all $\mathbf{p}_{-i} \in \Delta(\mathcal{Y})^{M-1}$ and $\mathbf{w}_{-i} \in \mathbb{R}_{\geq 0}^{M-1}$,*

$$\mathbb{E}_{\mathcal{Q}_i}[\pi_i((p_i, \mathbf{p}_{-i}), (w_i, \mathbf{w}_{-i}), Y) \mid \mathcal{F}_i] \geq 0.$$

C.4 Proof of BNE under Common Prior

Corollary 3.4 (BNE under common prior)**.** *Suppose agents share a common prior $\mathcal{Q}$ such that $\mathcal{Q}(\sum_{j\neq i} w_j > 0) = 1$ (Example 3.1). Let $p_i^* = \mathcal{Q}(Y = \cdot \mid \mathcal{F}_i)$ (Theorem 3.2) and $w_i^*(p_i^*) = (A_i/2c_3)^+$ with $A_i = \mathbb{E}[s(p_i^*, Y) - b_{-i} \mid \mathcal{F}_i]$ (Theorem 3.3). The strategy profile $\{(p_i^*, w_i^*(p_i^*))\}_{i=1}^M$ is a Bayes–Nash equilibrium: for every agent i and every alternative strategy $(\tilde{p}_i, \tilde{w}_i)$,*

$$\mathbb{E}_{\mathcal{Q}}\big[\pi_i\big((p_i^*, \mathbf{p}_{-i}^*), (w_i^*(p_i^*), \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big] \;\geq\; \mathbb{E}_{\mathcal{Q}}\big[\pi_i\big((\tilde{p}_i, \mathbf{p}_{-i}^*), (\tilde{w}_i, \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big].$$

Proof. Fix opponents' equilibrium strategies $(\mathbf{p}_{-i}^*, \mathbf{w}_{-i}^*)$. By Theorem 3.2, $p_i^* = \mathcal{Q}(Y = \cdot \mid \mathcal{F}_i)$ uniquely maximizes $\mathbb{E}_{\mathcal{Q}}[s(p_i, Y) \mid \mathcal{F}_i]$ over all p_i, and hence maximizes $A_i = \mathbb{E}_{\mathcal{Q}}[s(p_i, Y) - b_{-i} \mid \mathcal{F}_i]$. Given p_i^*, Theorem 3.3 gives the best-response wager $w_i^*(p_i^*) = (A_i/2c_3)^+$. When $A_i > 0$, the pair $(p_i^*, w_i^*(p_i^*))$ is the unique maximizer of the expected net payoff. When $A_i \leq 0$, $w_i^*(p_i^*) = 0$ and the net payoff is zero regardless of p_i; the agent is excluded from aggregation. □

C.5 Proof of NE under Heterogeneous Beliefs

Corollary 3.5 (NE under heterogeneous beliefs)**.** *Suppose each agent i holds her own belief $\mathcal{Q}_i$ such that $\mathcal{Q}_i(\sum_{j\neq i} w_j > 0) = 1$ (Example 3.2). Let $p_i^* = \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i)$ (Theorem 3.2) and $w_i^*(p_i^*) = (A_i/2c_3)^+$ with $A_i = \mathbb{E}_{\mathcal{Q}_i}[s(p_i^*, Y) - b_{-i} \mid \mathcal{F}_i]$ (Theorem 3.3). The strategy profile $\{(p_i^*, w_i^*(p_i^*))\}_{i=1}^M$ is a Nash equilibrium: for every agent i and every alternative strategy $(\tilde{p}_i, \tilde{w}_i)$,*

$$\mathbb{E}_{\mathcal{Q}_i}\big[\pi_i\big((p_i^*, \mathbf{p}_{-i}^*), (w_i^*(p_i^*), \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big] \;\geq\; \mathbb{E}_{\mathcal{Q}_i}\big[\pi_i\big((\tilde{p}_i, \mathbf{p}_{-i}^*), (\tilde{w}_i, \mathbf{w}_{-i}^*), Y\big) \mid \mathcal{F}_i\big].$$

Proof. Fix opponents' equilibrium strategies $(\mathbf{p}_{-i}^*, \mathbf{w}_{-i}^*)$. By Theorem 3.2, p_i^* maximizes $\mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) \mid \mathcal{F}_i]$ and hence maximizes $A_i(p_i) = \mathbb{E}_{\mathcal{Q}_i}[s(p_i, Y) - b_{-i} \mid \mathcal{F}_i]$ (since b_{-i} depends on opponents' strategies, not on p_i). By Theorem 3.3, for any fixed p_i, the expected net payoff is maximized at $w_i^*(p_i) = (A_i(p_i)/(2c_3))^+$. Since p_i^* maximizes $A_i(p_i)$ and the optimized expected net payoff $(A_i(p_i)^+)^2/(4c_3)$ is nondecreasing in $A_i(p_i)$, the pair $(p_i^*, w_i^*(p_i^*))$ maximizes the expected net payoff over all $(\tilde{p}_i, \tilde{w}_i)$ jointly. When $A_i > 0$, the pair $(p_i^*, w_i^*(p_i^*))$ is the unique maximizer of the expected net payoff. The NE does not require information about others' beliefs because $(p_i^*, w_i^*(p_i^*))$ maximizes the expected net payoff with respect to any opponents' beliefs and profile $(\mathbf{p}_{-i}, \mathbf{w}_{-i})$. □

C.6 Two Variants of Leave-One-Out Score Baseline

Theorems 3.2 and 3.3 hold for any leave-one-out baseline b_{-i} that does not depend on (p_i, w_i). We first instantiate two concrete choices that make different design tradeoffs, and then show that both variants admit a worst-case deficit bound independent of the number of participants and the amount wagered. These baselines are evaluated only for agents with positive wagers and require at least two such agents. Agents with zero wagers do not participate and receive zero net payoff.

Definition C.4 (Weighted-Score Baseline of WALLA I)**.** *The baseline is the wager-weighted average of other agents' scores:*

$$b_{-i}^{\mathrm{I}}(\mathbf{p}_{-i}, \mathbf{w}_{-i}, y) := \frac{\sum_{j\neq i} w_j\, s(p_j, y)}{\sum_{j\neq i} w_j}. \tag{6}$$

Definition C.5 (Weighted-Prediction Baseline of WALLA II)**.** *The baseline evaluates the scoring rule at the wager-weighted pool of other agents' predictions:*

$$b_{-i}^{\mathrm{II}}(\mathbf{p}_{-i}, \mathbf{w}_{-i}, y) := s(q_{-i}, y), \qquad q_{-i} := \frac{\sum_{j \neq i} w_j \, p_j}{\sum_{j \neq i} w_j} \in \Delta(\mathcal{Y}). \tag{7}$$

In short, WALLA I averages scores, whereas WALLA II scores an average prediction. The two variants trade off normality (Def. C.6) and no-arbitrage (Def. A.5). These two properties are in tension under known wagering mechanism designs. Lambert et al. [11] showed that WSWM satisfies normality, while Chen et al. [12] showed it admits arbitrage and proposed the no-arbitrage wagering mechanism (NAWM), which eliminates arbitrage but sacrifices normality. Our two variants exhibit the same tradeoff. We first adapt the definition of normality (Def. A.4) for our information model.

Definition C.6 (Normality)**.** *A mechanism satisfies normality if, for every $i \neq j \in \mathcal{M}$, any belief $\mathcal{Q}_i$ over the measurable space $(\Omega, \mathcal{F})$ such that $\mathcal{Q}_i(\sum_{j \neq i} w_j > 0) = 1$, $\mathcal{F}_i \subseteq \mathcal{F}$, $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$ and $\mathbf{p} \in \Delta(\mathcal{Y})^M$, defining $\mathbf{p}'$ by $p'_j = p_i^* := \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i)$ and $p'_k = p_k$ for $k \neq j$,*

$$\mathbb{E}_{\mathcal{Q}_i}[\pi_i(\mathbf{p}, \mathbf{w}, Y) \mid \mathcal{F}_i] \;\geq\; \mathbb{E}_{\mathcal{Q}_i}[\pi_i(\mathbf{p}', \mathbf{w}, Y) \mid \mathcal{F}_i],$$

with strict inequality whenever $p_j \neq \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i), w_i > 0,$ and $w_j > 0$. That is, if from agent i's perspective (i.e., $\mathcal{Q}_i$), the prediction of another agent j improves, then agent i's expected net payoff does not increase.

We defer proofs of this section to Appendix C.6.1.

Proposition C.7 (Normality of WALLA I)**.** *WALLA I satisfies normality.*

Proposition C.8 (Arbitrage in WALLA I)**.** *WALLA I does not satisfy no-arbitrage. When predictions from other positive-wager agents are not all identical, there exists a report $\hat{p}_i$ and a sufficiently small wager w_i such that $\pi_i > 0$ for all outcomes $y \in \mathcal{Y}$.*

Proposition C.9 (No-Arbitrage of WALLA II)**.** *WALLA II satisfies no-arbitrage.*

Proposition C.10 (Normality Failure of WALLA II)**.** *WALLA II does not satisfy normality: by adopting p_i^*, agent j can change the pooled baseline q_{-i} in a direction that increases player i's expected net payoff.*

Both variants satisfy DSIC under any belief structure (Theorem 3.2), advantage–wager alignment at best response (Theorem 3.3), and prediction-agnostic wager optimization, and the choice between variants is a design decision, depending on the presence of strategic behavior. However, both compromise the budget balance. We show below that the worst-case deficit of our mechanism is bounded and is independent of the number of participants M. For our net payoff function using the Brier score (Example 2.1), the worst-case deficit is c_2^2/c_3.

Proposition C.11 (Bounded Worst-Case Deficit)**.** *Under the net payoff function* (2) *with a bounded scoring rule ($s \in [\underline{s}, \overline{s}]$) for WALLA I, or a bounded and concave scoring rule for WALLA II (e.g., the Brier score), the total net payoff satisfies*

$$\sum_{i=1}^{M} \pi_i \;\leq\; \left(\frac{\overline{s} - \underline{s}}{W} - c_3 \right) \sum_{i=1}^{M} w_i^2 \;\leq\; \frac{(\overline{s} - \underline{s})^2}{4c_3} \tag{8}$$

where $W = \sum_i w_i$, for any prediction profile $\mathbf{p} \in \Delta(\mathcal{Y})^M$, any wager profile $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$, and realized outcome $y \in \mathcal{Y}$.

Compared to WSWM with a global average baseline [11], which achieves exact budget balance, our leave-one-out baseline inflates each agent's deviation by W/W_{-i}, producing the leave-one-out inflation term in (11). The regularization $c_3 \sum_i w_i^2$ serves to offset this inflation (e.g., by taxing wagers). The regularization constant c_3 is the designer's lever: a larger c_3 reduces the worst-case deficit at the cost of suppressing wagers.

C.6.1 Proofs of Properties of the Two Variants

Proposition C.7 (Normality of WALLA I)**.** *WALLA I satisfies normality.*

Proof. By strict properness, $p_i^* := \mathcal{Q}_i(Y = \cdot \mid \mathcal{F}_i)$ uniquely maximizes $\mathbb{E}[s(p, Y) \mid \mathcal{F}_i]$ when $w_i \geq 0$. With $w_j > 0$, replacing p_j with p_j' a prediction closer to p_i^* increases $\mathbb{E}[s(p_j, Y) \mid \mathcal{F}_i]$, which raises the baseline $\mathbb{E}[b^{\mathrm{I}}_{-i} \mid \mathcal{F}_i]$ and hence reduces A_i and the net payoff. When $w_i = 0$ or $w_j = 0$, the expected net payoff remains the same regardless of player j's prediction. □

Proposition C.8 (Arbitrage in WALLA I)**.** *WALLA I does not satisfy no-arbitrage. When predictions from other positive-wager agents are not all identical, there exists a report $\hat{p}_i$ and a sufficiently small wager w_i such that $\pi_i > 0$ for all outcomes $y \in \mathcal{Y}$.*

Proof. By Theorem 3.3 of Chen et al. [12], when at least two predictions differ ($p_j \neq p_k$ for some $j, k \neq i$), there exists $\hat{p}_i$ with $s(\hat{p}_i, y) > b^{\mathrm{I}}_{-i}$ for all y. Let $\delta(y) := s(\hat{p}_i, y) - b^{\mathrm{I}}_{-i} > 0$ and $\delta_{\min} := \min_y \delta(y) > 0$. Then $\pi_i = w_i(\delta(y) - c_3 w_i)$. Choosing any $w_i \in (0,\, \delta_{\min}/c_3)$ ensures $c_3 w_i < \delta(y)$ for all y, hence $\pi_i > 0$ for all y. □

Proposition C.9 (No-Arbitrage of WALLA II)**.** *WALLA II satisfies no-arbitrage.*

Proof. By Theorem 4.1 of Chen et al. [12], the base net payoff $\tilde{\pi}_i := w_i(s(p_i, y) - s(q_{-i}, y))$ has the form of a no-arbitrage wagering mechanism: there exists y^* with $\tilde{\pi}_i(y^*) \leq 0$ for any $\mathbf{p}$ and $\mathbf{w}$. Since $\pi_i = \tilde{\pi}_i - c_3 w_i^2 \leq \tilde{\pi}_i$, it follows that $\pi_i(y^*) \leq 0$. □

Proposition C.10 (Normality Failure of WALLA II)**.** *WALLA II does not satisfy normality: by adopting p_i^*, agent j can change the pooled baseline q_{-i} in a direction that increases player i's expected net payoff.*

Proof. We construct a counterexample with the Brier score $s(p, y) = c_1 - c_2\|p - \delta_y\|_2^2$ (Example 2.1).

Recall that normality (Def. C.6) is asserted for every *fixed* prediction profile $\mathbf{p} \in \Delta(\mathcal{Y})^M$ and wager profile $\mathbf{w} \in \mathbb{R}^M_{\geq 0}$; these are treated as deterministic parameters, and the conditional expectation $\mathbb{E}_{\mathcal{Q}_i}[\cdot \mid \mathcal{F}_i]$ is taken solely over the randomness in Y. Consequently, for any fixed $\mathbf{p}$ and $\mathbf{w}$, the pooled baseline $q_{-i} = \sum_{j \neq i} w_j p_j / W_{-i}$ is a fixed vector in $\Delta(\mathcal{Y})$ and $p_1 = \mathbb{E}_{\mathcal{Q}_i}[\delta_Y \mid \mathcal{F}_1]$ is $\mathcal{F}_1$-measurable,

$$\mathbb{E}_{\mathcal{Q}_i}\left[\|q_{-1} - \delta_Y\|_2^2 \mid \mathcal{F}_1\right] = \|q_{-1} - p_1\|_2^2 + \mathbb{E}_{\mathcal{Q}_i}\left[\|p_1 - \delta_Y\|_2^2 \mid \mathcal{F}_1\right],$$

where the cross-term $2\langle q_{-1} - p_1,\, \mathbb{E}_{\mathcal{Q}_i}[p_1 - \delta_Y \mid \mathcal{F}_1]\rangle = 2\langle q_{-1} - p_1,\, p_1 - p_1\rangle = 0$ vanishes.

Hence

$$\mathbb{E}_{\mathcal{Q}_i}[s(q_{-1}, Y) \mid \mathcal{F}_1] = c_1 - c_2\|q_{-1} - p_1\|_2^2 - c_2\mathbb{E}_{\mathcal{Q}_i}\left[\|p_1 - \delta_Y\|_2^2 \mid \mathcal{F}_1\right]. \tag{9}$$

Let $\mathcal{Y} = \{1, 2, 3\}$, $M = 3$, and consider normality for player $i = 1$. Fix the following deterministic profiles:

$$p_1 = (0.5,\, 0.5,\, 0), \quad p_2 = (1,\, 0,\, 0), \quad p_3 = (0,\, 1,\, 0), \quad w_2 = 1, \quad w_3 = 2,$$

so $W_{-1} = 3$.

The WALLA II baseline for player 1 is

$$q_{-1} = \frac{w_2 p_2 + w_3 p_3}{W_{-1}} = \tfrac{1}{3}(1, 0, 0) + \tfrac{2}{3}(0, 1, 0) = \left(\tfrac{1}{3}, \tfrac{2}{3}, 0\right).$$

The squared distance is

$$\|q_{-1} - p_1\|_2^2 = \left\|\left(-\tfrac{1}{6}, \tfrac{1}{6}, 0\right)\right\|_2^2 = \frac{1}{36} + \frac{1}{36} = \frac{1}{18}.$$

Now let $\mathbf{p}'$ be the profile obtained by setting $p_2' = p_1 = (0.5, 0.5, 0)$ (player 2's prediction improves to match player 1's true belief) while keeping $p_3' = p_3$ and all wagers unchanged.

The new pooled baseline is

$$q_{-1}' = \tfrac{1}{3}(0.5, 0.5, 0) + \tfrac{2}{3}(0, 1, 0) = \left(\tfrac{1}{6}, \tfrac{5}{6}, 0\right),$$

and the new squared distance to p_1 is

$$\|q'_{-1} - p_1\|_2^2 = \left\|\left(-\tfrac{1}{3}, \tfrac{1}{3}, 0\right)\right\|_2^2 = \frac{1}{9} + \frac{1}{9} = \frac{2}{9} > \frac{1}{18} = \|q_{-1} - p_1\|_2^2.$$

By (9),

$$\mathbb{E}_{\mathcal{Q}_i}[s(q'_{-1}, Y) \mid \mathcal{F}_1] < \mathbb{E}_{\mathcal{Q}_i}[s(q_{-1}, Y) \mid \mathcal{F}_1].$$

Because the net payoff subtracts the baseline score, and $w_1 > 0$, the expected payoff of player 1 strictly increases:

$$\mathbb{E}_{\mathcal{Q}_i}[\pi_1^{\mathrm{II}}(\mathbf{p}', \mathbf{w}, Y) \mid \mathcal{F}_1] > \mathbb{E}_{\mathcal{Q}_i}[\pi_1^{\mathrm{II}}(\mathbf{p}, \mathbf{w}, Y) \mid \mathcal{F}_1].$$

This is a contradiction to Def. C.6 and hence WALLA II fails normality. □

Proposition C.11 (Bounded Worst-Case Deficit)**.** *Under the net payoff function* (2) *with a bounded scoring rule ($s \in [\underline{s}, \overline{s}]$) for WALLA I, or a bounded and concave scoring rule for WALLA II (e.g., the Brier score), the total net payoff satisfies*

$$\sum_{i=1}^{M} \pi_i \;\le\; \left(\frac{\overline{s} - \underline{s}}{W} - c_3\right) \sum_{i=1}^{M} w_i^2 \;\le\; \frac{(\overline{s} - \underline{s})^2}{4c_3} \tag{8}$$

where $W = \sum_i w_i$, for any prediction profile $\mathbf{p} \in \Delta(\mathcal{Y})^M$, any wager profile $\mathbf{w} \in \mathbb{R}^M_{\ge 0}$, and realized outcome $y \in \mathcal{Y}$.

Proof. We first consider *WALLA I*. Write $s_i := s(p_i, y)$, $\bar{s}_w := \frac{\sum_j w_j s_j}{W}$ and $W_{-i} = \sum_{j \neq i} w_j$. From $W\bar{s}_w = w_i s_i + W_{-i} b^{\mathrm{I}}_{-i}$, we have:

$$s_i - b^{\mathrm{I}}_{-i} = \frac{W(s_i - \bar{s}_w)}{W_{-i}}, \qquad s_i - \bar{s}_w = \frac{W_{-i}}{W}(s_i - b^{\mathrm{I}}_{-i}). \tag{10}$$

Writing $\frac{W}{W_{-i}} = 1 + \frac{w_i}{W_{-i}}$ and using $\sum_i w_i(s_i - \bar{s}_w) = 0$, we write our net payoff function (2) as the difference of the leave-one-out (LOO) inflation term and the regularization term:

$$\sum_{i=1}^{M} \pi_i = \underbrace{\sum_{i=1}^{M} \frac{w_i^2(s_i - \bar{s}_w)}{W_{-i}}}_{\text{LOO inflation}} - \underbrace{c_3 \sum_{i=1}^{M} w_i^2}_{\text{regularization}}. \tag{11}$$

Using (10) and $s_i - b^{\mathrm{I}}_{-i} \le \overline{s} - \underline{s}$, we have

$$\frac{w_i^2(s_i - \bar{s}_w)}{W_{-i}} = \frac{w_i^2(s_i - b^{\mathrm{I}}_{-i})}{W} \le \frac{w_i^2(\overline{s} - \underline{s})}{W}.$$

Therefore, the LOO inflation is at most $\frac{\overline{s} - \underline{s}}{W} \sum_i w_i^2$, giving the first inequality in (8).

For the second: since $\sum_i w_i^2 / W \le w_{\max}$ and $\sum_i w_i^2 \ge w_{\max}^2$,

$$\frac{(\overline{s} - \underline{s}) \sum_i w_i^2}{W} - c_3 \sum_i w_i^2 \;\le\; (\overline{s} - \underline{s})\, w_{\max} - c_3\, w_{\max}^2 \;\le\; \frac{(\overline{s} - \underline{s})^2}{4c_3},$$

where the last step maximized at $(\overline{s} - \underline{s})/(2c_3)$ with value $(\overline{s} - \underline{s})^2/(4c_3)$.

WALLA II. When $s(\cdot, y)$ is concave in p, Jensen's inequality gives $b^{\mathrm{II}}_{-i} \ge b^{\mathrm{I}}_{-i}$ and thus $\pi_i^{\mathrm{II}} \le \pi_i^{\mathrm{I}}$ for all prediction and wager profiles, so WALLA I's bound applies. □

C.7 Aggregation

After eliciting predictions $\{p_i\}_{i=1}^M$ and wagers $\{w_i\}_{i=1}^M$, the platform aggregates individual reports into a single probabilistic forecast $\hat{p} \in \Delta(\mathcal{Y})$ using wagers as weights.

Linear pooling. The wager-weighted arithmetic mean

$$\hat{p}_{\text{lin}} := \frac{\sum_{i=1}^{M} w_i \, p_i}{W} \in \Delta(\mathcal{Y}) \tag{12}$$

is the unique minimizer of the wager-weighted sum of squared Euclidean distances $\sum_i w_i \|q - p_i\|_2^2$ over $q \in \Delta(\mathcal{Y})$, or equivalently, the minimizer of the wager-weighted KL divergence $\sum_i w_i \, \mathrm{KL}(p_i \| q)$. Linear pooling is computationally simple and tends to preserve disagreement among forecasters.

Logarithmic pooling. The wager-weighted geometric mean

$$\hat{p}_{\log}[y] \propto \prod_{i=1}^{M} p_i[y]^{w_i/W}, \qquad y \in \mathcal{Y}, \tag{13}$$

minimizes the wager-weighted reverse KL divergence $\sum_i w_i \, \mathrm{KL}(q \| p_i)$. With pooling weights held fixed, logarithmic pooling is externally Bayesian under a common likelihood update [51], but emphasizes regions of consensus among forecasters, typically yielding a sharper aggregated distribution and exhibiting sensitivity to probabilities near zero.

D Deferred Content from Sec. 4

D.1 Datasets

Table 5 shows basic information about each dataset.

Table 5: Dataset attributes.

	PubMedQA [45]	BayesX	MedMCQA [46]	MMLU [47]	ARC-Challenge [48]
Size	37.82K	10K	183K	116K	1.17K
Task	Binary Classification	Distribution Matching	Multiple Choice	Multiple Choice	Multiple Choice
Private Context	✓	✓	✗	✗	✗

PubMedQA [45][5] is a specialized biomedical research dataset designed for answering questions based on medical abstracts. The primary task is to provide a "yes," "no," or "maybe" response to a clinical question by synthesizing information from an associated PubMed abstract. For our experiment, we convert this into a binary classification by focusing on "yes" and "no" questions. We also balance the dataset such that there is a similar number of "yes" and "no" questions. We extract the abstract per question and randomly assign relevant or irrelevant abstracts to a subset of participating models.

MedMCQA [46][6] is a large-scale, multiple-choice question-answering dataset specifically curated from Indian medical entrance exams (AIIMS and NEET-PG). It covers 21 diverse medical subjects, such as anatomy, pathology, and pharmacology. Because these questions are designed to test human medical students, they require deep domain knowledge, making MedMCQA a standard tool for measuring the clinical competency of general and medical-purpose LLMs.

MMLU [47][7] is one of the most comprehensive benchmarks for evaluating the general knowledge and problem-solving capabilities of language models. It spans 57 subjects across STEM, the humanities, social sciences, and more, testing both world knowledge and analytical skill. The questions are multiple-choice and range in difficulty from elementary levels to advanced professional standards (such as law or medicine). MMLU has become the industry standard for reporting a model's foundational "intelligence" and its ability to handle a vast breadth of human knowledge.

ARC-Challenge [48][8] is a high-difficulty subset of the AI2 Reasoning Challenge, consisting of grade-school science questions that are particularly difficult for automated systems to solve. The

[5] We use the publicly available PubMedQA dataset on Hugging Face: `https://huggingface.co/datasets/qiaojin/PubMedQA`.

[6] We use the publicly available MedMCQA dataset on Hugging Face: `https://huggingface.co/datasets/openlifescienceai/medmcqa`.

[7] We use the publicly available MMLU dataset on Hugging Face: `https://huggingface.co/datasets/cais/mmlu`.

[8] We use the publicly available ARC-Challenge dataset on Hugging Face: `https://huggingface.co/datasets/allenai/ai2_arc`.

Challenge set contains questions that cannot be answered by simple keyword matching or statistical co-occurrence alone; they often require a degree of logical reasoning, common sense, or a basic understanding of scientific processes. It remains a key metric for determining how well a model can move beyond rote memorization toward actual conceptual understanding.

BayesX is a new dataset we constructed to capture a setting in which additional information reduces epistemic uncertainty about the world while revealing greater aleatoric uncertainty in the outcome-generating process. Existing QA datasets primarily evaluate whether additional context or expert knowledge improves prediction accuracy. Forecasting presents a different challenge: additional information may reduce epistemic uncertainty about the world while simultaneously revealing greater aleatoric uncertainty in the outcome-generating process. BayesX is designed to capture this phenomenon. Its priors can be highly concentrated, but private context often shifts the posterior toward a more uniform distribution, reflecting a better-informed yet less confident prediction. As a result, BayesX tests whether an aggregation method can recognize the informed model even when additional information makes its forecast appear less certain.

Each example in BayesX describes a binary cluster-saturation forecasting problem. The prior probability is either 0.1 or 0.9, and the context contains a binary warning signal with sampled true-positive and false-positive rates. The evidence is selected so that receiving additional information can move the posterior closer to a uniform distribution. The ground truth is therefore the posterior probability rather than only a hard binary label.[9]

D.2 Formal Definitions of Metrics

We formally define Kendall's Tau, MRR, and dynamic regret in terms of the Brier score as follows.

Definition D.1 (Kendall's Tau (K-Tau))**.** *We use K-Tau to measure the ordinal association between two orderings: (1) the ordering of models by their per-question Brier-score performance and (2) the ordering of models by their wagers. Formally,*

$$\tau = \frac{\textit{(number of concordant pairs) - (number of discordant pairs)}}{\textit{(number of pairs)}}.$$

Definition D.2 (Mean Reciprocal Rank (MRR) of the Best Experts)**.** *The per-sample reciprocal rank is the inverse rank of the best expert in terms of the Brier score. Formally,*

$$\textit{MRR} = \frac{1}{N}\sum_{n=1}^{N} \frac{1}{\textit{rank}(\arg\min_j \|\delta_y^{(n)} - p_j^{(n)}\|_2^2)},$$

where N is the number of samples and the rank is induced by the aggregation weights.

Definition D.3 (Dynamic Regret (D-Regret) in the Brier Score)**.** *The D-Regret is defined as*

$$\frac{1}{N}\sum_{n=1}^{N} \|\delta_y^{(n)} - p_{agg}^{(n)}\|_2^2 - \min_j \|\delta_y^{(n)} - p_j^{(n)}\|_2^2.$$

It measures the Brier-score gap between the aggregated prediction and the best individual model on each question.

D.3 Baselines

We consider seven baselines spanning learning-free and learning-based methods.

Among *learning-free methods*, UniformAvg performs equally weighted aggregation. SelfCertainty [9] uses the predicted distribution to estimate confidence and treats it as the aggregation weight. Pack-LLM [10] uses the inverse perplexity on input tokens as weights, requiring access to model parameters. These methods can be considered decentralized, and some are uncertainty-aware, but they do not learn from data, may be sensitive to miscalibrated confidence estimates, and are not incentive-compatible—participants can misreport confidence or perplexity to gain disproportionate weight.

Learning-based methods, including pre-inference routers [2, 3, 4] and StackedGen [5], require centralized control to train routing or aggregation networks. Pre-inference routers learn weights conditional on input features but do not capture each model's internal uncertainty on the specific

[9] We include the BayesX dataset in the repo `https://github.com/chailab-rutgers/WALLA`.

question. StackedGen trains on concatenated hidden states and is therefore uncertainty-aware, but requires centralized access to all models' internals. Since StackedGen has access to all models' hidden states, it serves as a strong centralized reference point, with more information than any decentralized method.

We describe each method in detail below.

Uniform averaging (UniformAvg) is a simple method that averages the predicted probability distributions with equal weights (i.e., $p_{\text{agg}} = \frac{\sum_{i=1}^{M} p_i}{M}$).

SelfCertainty [9] is a learning-free ensembling method that uses model confidence as weight for averaging. We use the variant that quantifies confidence through the Kullback-Leibler (KL) divergence between the model's predicted token probability distribution and a uniform distribution, as it performs best.[10]

PackLLM [10] is a learning-free ensembling method that aims to minimize the perplexity of the input prompt for the ensemble of all models. As PackLLM relies on the perplexity of the input tokens for each model, it requires open-weight models. In a decentralized setting where different providers control different models, providers can misreport perplexities to gain disproportionate weights.[11]

RouterDC [2] is a pre-inference router that finetunes a DeBERTaV3-base [52] language model via dual-contrastive learning. It aligns a query's embedding with the top-K most capable LLMs for that specific task while distancing it from the last-K models.[12]

Song et al. [4] propose a pre-inference router inspired by Item Response Theory (IRT) called IRTRouter. It treats user queries as "test items" and LLMs as "respondents," and explicitly models the relationship between a specific query's difficulty and an individual model's latent capabilities. It uses a frozen BERT model to retrieve the query embedding. We adopt NIRTRouter, a variant that trains neural networks for the IRT implementation.[13]

RouteLLM [3] was designed to choose between a more capable but expensive model and a weaker and cheaper model using human pairwise preference as feedback. Its base router is a BERT model that is finetuned using preference data. We extend this framework for routing on more LLMs using cross-entropy loss and pairwise ranking loss.[14]

Stacked generalization (StackedGen) [5] is a classical ensembling technique that learns a neural network to combine diverse predictive models to achieve greater overall accuracy than any single model could produce alone. We train a two-hidden-layer MLP with cross-entropy loss. For consistency with WALLA, we use the LLMs' last-layer hidden states as inputs.

Table 6: The mechanism parameters and learning parameters applied to all learning-based methods.

Mechanism Parameters					Learning Parameters				
c_1	c_2	c_3	Aggregation	Scoring rule	Batch size	Dataset split	MLP hidden sizes	Optimizer	Activation
1	1/2	1/2	Linear pooling	Brier score	100	8:1:1	512, 256	Adam	ReLU

D.4 Implementation Details of the Decentralized Wager Policy Learning

In WALLA, LLM participants produce predictions via their forward passes and learn wager policies as described in Sec. 3.3. The goal is for each agent to learn her comparative advantage: *on which or what type of questions it is likely to outperform the pool, and by how much*. We allow all participants' wager networks to be trained simultaneously without assuming that opponents' strategies are fixed. This procedure simulates how, in a multi-agent system, each LLM participant would independently learn to wager in a deployed wagering mechanism, using only her own model or context representations and observed payoffs. A key property of the payoff function (2) is that each agent can recover the *hindsight best-response wager* from her own net payoff alone: since $\pi_i = w_i(s_i - b_{-i} - c_3 w_i)$, the realized

[10] We use the public implementation of SelfCertainty from `https://github.com/backprop07/Self-Certainty`.

[11] We use the public implementation of PackLLM from `https://github.com/cmavro/PackLLM`.

[12] We use the public implementation of RouterDC from `https://github.com/shuhao02/RouterDC`. We use the pretrained DeBERTaV3 from `https://huggingface.co/microsoft/deberta-v3-base`.

[13] We use the public implementation of NIRTRouter from `https://github.com/Mercidaiha/IRT-Router`. We adopt the pretrained BERT from `https://huggingface.co/google-bert/bert-base-uncased`.

[14] We adapt the public implementation of RouteLLM from `https://github.com/lm-sys/RouteLLM.git`. We adopt the pretrained BERT from `https://huggingface.co/google-bert/bert-base-uncased`.

comparative advantage $s_i - b_{-i} = \pi_i/w_i + c_3 w_i$ ($w_i > 0$) and thus the hindsight best-response wager is computable from the agent's observed payoff and wager, without access to other agents' predictions, wagers, or scores: $w^* = ((s_i - b_{-i})/2c_3)^+$. For the main experiment, we train the wager network to minimize the hindsight-regret of failing to play the best response wager (as discussed in Sec. 3.3). In Appendix D.7, we provide the result for an alternative training objective that minimizes the mean-squared error (MSE) between the predicted wager and the hindsight best-response wager.

The base model parameters are frozen while only the wager network is trained. Our main experiment assumes access to LLM internals. We extract the final-token last-layer hidden state after processing the input tokens (i.e., the question X and context S). In the scenario with heterogeneous models and private context (Sec. 4.4), we also report results that do not assume access to model internals to demonstrate generalization to models accessible only via APIs. We use BERT [53] embeddings of the input question and context in these cases to train wager networks.

We then pass these representations to a two-hidden-layer MLP to output a non-negative scalar wager. To preserve the feasible range of w^* while improving numerical stability, we apply a sigmoid transformation to the output of the wager head. The mechanism parameters and learning hyperparameters are reported in Table 6.

D.5 Additional Experiment Setup Details

The learning parameters provided in Table 6 are common to all methods. For every dataset, we use an $8{:}1{:}1$ train-validation-test split and fix the batch size to 100. For learning-based methods, we use the Adam optimizer, tune the learning rate with grid search and adopt early stopping to prevent overfitting. For baselines, we stop at the epoch that achieves the best validation performance. For WALLA, because the training procedure is decentralized, we use the Brier score on the most recent training batches as the early stopping criterion, ensuring that the number of training-batch samples used for early stopping matches the validation-set size. Unless specified otherwise, all methods use linear pooling for aggregation. In all experiments, the aggregated predictions are evaluated on a held-out test set, where no further updates to the wagering strategy are allowed.

Experiments are run on 4 NVIDIA RTX PRO 6000 Blackwell Server Edition GPUs (96 GiB VRAM each) and a dual-socket AMD EPYC 9354 setup (64 CPU cores total). It provides about 1.1 TiB system RAM (plus 33GiB swap).

Table 7: Homogeneous models (Gemma2) on BayesX with private contexts. For each question, one of $M \in \{4, 8, 12\}$ models is randomly selected to receive evidence.[15]

	4				8				12			
Method	KLD ↓	TVD ↓	MRR ↑	DRegret ↓	KLD ↓	TVD ↓	MRR ↑	DRegret ↓	KLD ↓	TVD ↓	MRR ↑	DRegret ↓
UniformAvg	25.64	32.96	-	19.65	41.09	38.93	-	28.79	48.67	40.76	-	32.02
SelfCertainty	61.44	43.99	25.00	37.37	68.46	44.58	12.50	38.74	70.80	44.59	8.33	38.93
PackLLM	21.58	30.70	100.00	16.66	37.36	37.75	100.00	26.75	45.32	39.93	100.00	30.60
RouterDC w/ ctx.	22.88$_{\pm 0.26}$	30.00$_{\pm 0.38}$	65.55$_{\pm 12.27}$	16.77$_{\pm 0.18}$	39.96$_{\pm 14.38}$	38.38$_{\pm 6.81}$	39.01$_{\pm 64.03}$	28.02$_{\pm 9.66}$	48.96$_{\pm 0.60}$	40.73$_{\pm 0.12}$	26.15$_{\pm 3.64}$	32.00$_{\pm 0.22}$
NIRTRouter w/ ctx.	22.49$_{\pm 17.12}$	30.72$_{\pm 16.17}$	52.11$_{\pm 0.32}$	17.11$_{\pm 15.72}$	40.50$_{\pm 0.49}$	38.76$_{\pm 0.03}$	33.97$_{\pm 0.00}$	-0.35$_{\pm 1.46}$	45.18$_{\pm 0.11}$	38.97$_{\pm 0.02}$	25.91$_{\pm 0.11}$	29.74$_{\pm 0.11}$
RouteLLM w/ ctx.	0.64$_{\pm 0.07}$	3.75$_{\pm 0.92}$	100.00$_{\pm 0.00}$	-2.31$_{\pm 0.07}$	0.83$_{\pm 2.32}$	4.67$_{\pm 13.61}$	100.00$_{\pm 0.00}$	-2.12$_{\pm 2.25}$	0.93$_{\pm 2.46}$	5.27$_{\pm 10.68}$	100.00$_{\pm 0.00}$	-2.02$_{\pm 2.40}$
RouteLLM w/o ctx.	25.67$_{\pm 0.22}$	32.96$_{\pm 0.01}$	52.08$_{\pm 0.00}$	19.66$_{\pm 0.08}$	41.11$_{\pm 0.20}$	38.92$_{\pm 0.00}$	33.97$_{\pm 0.00}$	28.78$_{\pm 0.01}$	48.68$_{\pm 0.15}$	40.76$_{\pm 0.06}$	25.91$_{\pm 0.05}$	32.01$_{\pm 0.10}$
StackedGen	1.27$_{\pm 0.04}$	6.45$_{\pm 0.11}$	100.00$_{\pm 0.00}$	-1.69$_{\pm 0.04}$	1.51$_{\pm 0.03}$	7.06$_{\pm 0.08}$	100.00$_{\pm 0.00}$	-1.46$_{\pm 0.03}$	1.82$_{\pm 0.04}$	7.93$_{\pm 0.10}$	100.00$_{\pm 0.00}$	-1.13$_{\pm 0.04}$
WALLA I	1.99$_{\pm 0.06}$	8.23$_{\pm 0.13}$	100.00$_{\pm 0.00}$	-1.01$_{\pm 0.03}$	1.96$_{\pm 0.09}$	8.20$_{\pm 0.17}$	100.00$_{\pm 0.00}$	-1.00$_{\pm 0.09}$	2.23$_{\pm 0.12}$	8.91$_{\pm 0.23}$	100.00$_{\pm 0.00}$	-0.77$_{\pm 0.10}$
WALLA II	2.01$_{\pm 0.07}$	8.29$_{\pm 0.16}$	100.00$_{\pm 0.00}$	-0.99$_{\pm 0.05}$	2.27$_{\pm 0.12}$	8.83$_{\pm 0.24}$	100.00$_{\pm 0.00}$	-0.70$_{\pm 0.12}$	2.23$_{\pm 0.12}$	8.91$_{\pm 0.23}$	100.00$_{\pm 0.00}$	-0.77$_{\pm 0.10}$

Table 8: Homogeneous models (Aloe) on PubMedQA with private contexts. Questions that all models can answer correctly are filtered out. For each question, one of $M \in \{4, 8, 12\}$ models is randomly selected to receive a relevant abstract; the rest receive none.

	4					8					12				
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓
UniformAvg	26.05	30.44	30.89	54.57	39.27	19.29	20.36	43.32	39.10	48.38	19.92	17.03	47.68	31.92	51.66
SelfCertainty	54.42	48.94	23.12	67.94	31.05	51.38	43.15	27.94	62.45	36.41	49.71	39.82	30.81	60.65	39.30
PackLLM	78.23	63.72	4.72	89.58	15.18	48.75	55.15	9.35	88.83	22.87	41.43	48.42	41.43	88.45	28.00
RouterDC w/ ctx.	24.58$_{\pm 27.7}$	31.57$_{\pm 61.8}$	30.69$_{\pm 59.0}$	61.57$_{\pm 366}$	39.32$_{\pm 39.6}$	19.86$_{\pm 2.61}$	20.32$_{\pm 3.61}$	43.30$_{\pm 4.81}$	48.12$_{\pm 146}$	48.17$_{\pm 4.22}$	20.22$_{\pm 6.03}$	17.03$_{\pm 1.03}$	47.61$_{\pm 0.21}$	41.98$_{\pm 186}$	51.47$_{\pm 2.50}$
NIRTRouter w/ ctx.	25.81$_{\pm 3.03}$	30.56$_{\pm 1.55}$	30.77$_{\pm 1.54}$	52.02$_{\pm 8.94}$	39.27$_{\pm 0.04}$	19.29$_{\pm 0.00}$	20.36$_{\pm 0.00}$	43.32$_{\pm 0.00}$	33.89$_{\pm 17.09}$	48.38$_{\pm 0.00}$	19.92$_{\pm 0.03}$	17.03$_{\pm 0.00}$	47.67$_{\pm 0.00}$	24.50$_{\pm 1.64}$	51.66$_{\pm 0.01}$
RouteLLM w/ ctx.	72.54$_{\pm 0.15}$	69.27$_{\pm 0.15}$	4.89$_{\pm 0.09}$	89.30$_{\pm 0.76}$	6.47$_{\pm 0.14}$	72.55$_{\pm 0.07}$	72.55$_{\pm 0.07}$	4.99$_{\pm 0.26}$	87.92$_{\pm 0.79}$	6.53$_{\pm 0.05}$	73.11$_{\pm 0.11}$	69.08$_{\pm 0.08}$	5.15$_{\pm 0.08}$	86.59$_{\pm 0.89}$	6.55$_{\pm 0.04}$
RouteLLM w/o ctx.	23.03$_{\pm 0.59}$	29.69$_{\pm 0.49}$	31.82$_{\pm 0.50}$	52.08$_{\pm 0.42}$	39.24$_{\pm 0.01}$	19.52$_{\pm 0.83}$	20.00$_{\pm 0.46}$	43.68$_{\pm 0.46}$	33.43$_{\pm 0.90}$	48.52$_{\pm 0.02}$	22.00$_{\pm 0.81}$	16.26$_{\pm 0.41}$	48.47$_{\pm 0.40}$	27.78$_{\pm 2.65}$	51.64$_{\pm 0.06}$
StackedGen	71.57$_{\pm 0.36}$	69.25$_{\pm 0.27}$	4.27$_{\pm 0.20}$	89.18$_{\pm 0.44}$	6.26$_{\pm 0.34}$	73.38$_{\pm 1.09}$	69.15$_{\pm 0.45}$	5.02$_{\pm 0.58}$	87.10$_{\pm 1.11}$	6.48$_{\pm 0.61}$	72.68$_{\pm 0.42}$	69.05$_{\pm 0.70}$	4.70$_{\pm 0.50}$	85.99$_{\pm 1.10}$	6.90$_{\pm 0.65}$
WALLA I	72.21$_{\pm 0.03}$	68.85$_{\pm 0.00}$	5.08$_{\pm 0.11}$	89.59$_{\pm 0.23}$	6.32$_{\pm 0.01}$	73.54$_{\pm 0.13}$	69.01$_{\pm 0.07}$	5.30$_{\pm 0.11}$	86.80$_{\pm 0.51}$	6.48$_{\pm 0.10}$	72.89$_{\pm 0.05}$	69.31$_{\pm 0.00}$	4.94$_{\pm 0.04}$	86.03$_{\pm 0.86}$	6.56$_{\pm 0.01}$
WALLA II	71.94$_{\pm 0.09}$	68.94$_{\pm 0.00}$	4.94$_{\pm 0.04}$	89.55$_{\pm 0.22}$	6.35$_{\pm 0.03}$	73.54$_{\pm 0.10}$	69.01$_{\pm 0.07}$	5.27$_{\pm 0.06}$	87.15$_{\pm 1.15}$	6.48$_{\pm 0.08}$	72.89$_{\pm 0.09}$	69.31$_{\pm 0.00}$	4.94$_{\pm 0.04}$	86.17$_{\pm 0.66}$	6.56$_{\pm 0.03}$

[15] Gemma2 achieves KLD=3.24 and TVD=10.27 with evidence, and KLD=78.30 and TVD=45.46 without.

Table 9: 2 Heterogeneous models (Llama3.1 and Aloe) without context on three datasets.

	MedMCQA (i.d.)					MMLU (i.d.)					ARC-Challenge (o.o.d.)				
Method	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
Llama3.1	78.97	2.64	69.51	-21.95	10.62	81.99	2.52	68.51	-25.96	7.37	81.30	3.06	68.39	-26.43	8.89
Aloe	79.77	3.18	80.49	21.95	10.42	80.23	8.77	81.49	25.96	11.10	80.35	8.58	81.61	26.43	11.31
UniformAvg	81.76	4.60	69.51	0.00	7.58	82.22	2.47	68.51	0.00	6.66	82.87	3.05	68.39	0.00	7.05
SelfCertainty	81.73	2.24	88.64	54.56	7.32	82.15	5.07	88.57	54.29	7.56	82.87	3.94	87.74	50.96	7.34
PackLLM	81.54	4.18	79.04	16.17	7.62	82.27	2.49	72.53	-9.90	6.65	82.70	2.80	75.70	2.78	7.15
RouterDC	82.01±3.58	4.36±0.62	80.18±3.91	20.71±15.6	7.49±1.11	82.21±0.04	2.54±0.13	76.30±8.83	5.19±35.3	6.68±0.03	82.89±0.05	2.86±0.45	76.32±8.99	5.29±35.9	7.07±0.03
NIRTRouter	81.98±3.61	4.27±0.98	80.56±0.94	22.24±3.75	7.49±1.13	82.36±0.57	2.23±0.23	68.51±0.03	-25.97±0.13	6.54±0.55	82.78±0.00	3.42±0.00	68.39±0.00	-26.43±0.00	7.00±0.00
RouteLLM	81.82±2.71	3.89±1.74	80.56±0.94	22.24±3.75	7.51±1.14	82.56±0.19	1.86±1.28	68.51±0.03	-25.97±0.13	6.37±0.07	83.13±0.00	2.13±0.00	68.39±0.00	-26.43±0.00	6.94±0.00
StackedGen	82.61±0.28	2.16±0.32	85.51±0.39	42.04±1.57	6.35±0.11	82.52±0.21	2.29±0.21	77.47±1.03	9.88±4.12	6.21±0.19	83.33±0.21	3.07±0.35	75.22±0.37	0.87±1.46	6.61±0.03
WALLA I	82.01±0.06	3.64±0.20	77.03±0.74	8.10±2.98	7.03±0.08	82.51±0.09	1.68±0.13	68.61±0.05	-25.57±0.21	6.37±0.03	82.13±0.24	2.34±0.20	68.92±0.03	-24.30±0.14	7.19±0.12
WALLA II	81.94±0.17	3.62±0.58	77.16±2.79	8.60±11.18	7.07±0.12	82.52±0.05	1.85±0.15	68.81±0.29	-24.78±1.17	6.42±0.06	82.22±0.14	2.73±0.57	68.95±0.03	-24.22±0.14	7.06±0.07
WALLA I log-pool	81.87±0.31	2.71±0.26	72.19±4.83	-11.26±19.33	7.02±0.30	82.55±0.09	3.05±0.19	68.67±0.04	-25.33±0.17	6.50±0.02	82.22±0.18	2.90±0.23	68.91±0.00	-24.35±0.00	7.12±0.22
WALLA II log-pool	82.05±0.28	2.51±0.23	75.42±4.74	1.66±18.95	6.79±0.34	82.54±0.15	3.03±0.15	68.68±0.02	-25.28±0.09	6.50±0.02	82.30±0.18	2.81±0.51	68.91±0.00	-24.35±0.00	7.12±0.22
WALLA I calibrate	82.40±0.04	4.16±1.31	76.43±3.11	5.74±12.43	6.67±0.42	82.75±0.08	2.19±0.08	76.70±0.47	6.79±1.89	5.80±0.06	82.11±0.26	5.90±1.45	88.88±0.25	55.52±0.99	7.46±0.03

Table 10: 2 Heterogeneous models (Llama3.1 and Aloe) without context on three datasets, with questions that both models can answer correctly filtered out.

	MedMCQA (i.d.)					MMLU (i.d.)					ARC-Challenge (o.o.d.)				
Method	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
Llama3.1	26.38	30.11	77.69	10.78	24.86	26.94	38.89	81.74	26.94	21.41	27.85	38.59	80.37	21.48	24.50
Aloe	29.20	36.31	72.31	-10.78	32.14	19.79	54.53	68.26	-26.94	41.29	24.16	50.18	69.63	-21.48	39.75
UniformAvg	36.17	17.01	77.69	0.00	20.68	27.87	34.11	81.74	0.00	22.88	33.89	31.10	80.37	0.00	22.62
SelfCertainty	36.03	23.67	72.61	-9.55	22.65	27.59	40.88	67.33	-30.68	28.38	33.89	34.01	69.46	-22.15	25.80
PackLLM	35.40	18.32	73.39	-6.45	21.08	28.08	33.83	78.34	13.37	22.79	33.22	31.28	73.32	-6.71	23.04
RouterDC	36.44±5.21	16.94±6.40	72.11±2.45	-11.55±9.82	20.66±1.91	27.81±0.17	34.39±0.46	73.65±9.16	-5.39±36.6	22.97±0.17	33.96±0.19	29.69±0.15	73.93±7.30	-4.30±29.2	22.71±0.17
NIRTRouter	36.34±5.34	17.10±6.61	72.04±3.38	-11.84±13.5	20.73±1.92	28.68±5.62	33.36±1.67	81.28±5.82	25.11±23.2	22.06±4.98	33.56±0.00	28.53±0.01	80.37±0.00	21.48±0.00	22.30±0.02
RouteLLM	35.78±2.09	18.07±2.44	72.04±3.38	-11.84±13.5	21.03±1.65	29.49±2.49	32.43±1.57	81.28±5.82	25.11±23.2	20.73±2.98	34.90±0.00	26.61±0.02	80.37±0.00	21.48±0.00	21.33±0.04
StackedGen	38.65±1.16	20.23±1.00	78.26±0.29	13.02±1.15	13.02±1.15	29.80±0.60	32.88±0.88	79.54±0.40	18.17±1.61	20.44±0.67	35.65±0.80	29.35±1.20	80.29±0.15	21.14±0.62	20.42±0.10
WALLA I	36.76±0.23	20.37±0.77	77.43±0.71	9.72±2.83	19.72±0.43	29.37±0.81	31.47±1.36	81.04±0.99	24.17±3.96	20.85±0.66	34.40±0.53	26.95±0.46	80.16±0.26	20.64±1.02	21.00±0.03
WALLA II	36.55±0.88	16.34±0.47	75.67±2.61	2.68±10.4	20.64±0.28	29.12±0.75	31.71±0.88	81.01±0.84	24.03±3.37	21.00±0.85	34.06±0.69	26.87±0.74	80.20±0.38	20.81±1.51	21.03±0.09
WALLA I log-pool	36.27±1.07	20.26±1.74	77.14±0.97	8.57±3.87	20.87±0.79	29.00±0.37	35.38±0.33	81.86±0.10	27.44±0.42	22.54±0.47	31.83±0.68	32.67±0.88	81.86±0.10	22.67±0.00	21.93±0.32
WALLA II log-pool	36.91±1.00	21.26±1.61	77.53±1.09	10.12±4.35	20.40±0.88	28.95±0.60	35.45±0.48	81.91±0.04	27.63±0.15	22.46±0.47	32.17±0.68	32.31±0.73	80.67±0.00	22.67±0.00	21.91±0.28
WALLA I calibrate	38.14±0.15	14.47±3.20	76.22±2.03	4.88±8.13	18.33±0.89	29.84±0.32	26.01±0.13	77.18±1.12	8.70±4.49	18.33±0.23	31.42±1.00	24.96±3.12	69.88±0.25	-20.50±1.02	21.38±1.45

D.6 Result and Discussion on Including Llama3.1-70B

As shown in Table 13, adding the 70B model to the pool matches the overall aggregation performance with improved ECE, even though the 70B model's standalone performance falls short of the aggregated output. This occurs because private context is randomly distributed among agents; under such decentralized information sharing, a larger model does not inherently dominate. This observation further underscores the necessity of decentralized aggregation.

D.7 Result and Discussion on an Alternative Wager Policy Optimization Objective

In our main experiment, the learning objective for the wager policy is to minimize the hindsight regret of failing to play the best-response wager (Sec. 3.3). We explore an alternative training objective that minimizes the mean-squared error (MSE) between the predicted wager and the hindsight best-response wager. This objective is equivalent to minimizing the hindsight regret of failing to play w^* in each round when $w^* > 0$. When $w^* = 0$, the MSE objective yields a less aggressive update toward zero than direct regret minimization, as the penalty is quadratic rather than linear, thereby encouraging continued participation.

We provide the results in Tables 14–17, denoted with BR-MSE, for the comparison to hindsight regret minimization. As the MSE objective imposes a less aggressive update when the comparative advantage is negative, more wagers are well-above zero. This leads to worse ECE and D-Regret as the worse performing models can have non-trivial contribution to the aggregation. However, it enables an extended time horizon to explore the comparative advantages across samples. Hence, Fig. 7 shows improved alignment between the hindsight comparative advantage and the predicted wager compared to regret minimization(Fig. 2); Fig. 8 further demonstrates that the predicted wagers strongly correlate with the expected score advantages computed with a pair of independent prediction heads on own scores and the baseline scores. Future work can explore other approaches to trade off exploration and exploitation.

Table 11: Heterogeneous models (Gemma2, Aloe, Llama3.1, and BioMistral) without context on three datasets, with questions that all models can answer correctly filtered out.

	MedMCQA (i.d.)					MMLU (i.d.)					ARC-Challenge (o.o.d.)				
Method	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
Gemma2	43.25	40.85	56.83	-8.50	71.61	70.20	21.74	76.62	21.61	29.71	78.04	17.82	83.15	28.84	24.36
Aloe	69.05	7.86	60.32	16.12	23.26	54.05	26.10	50.61	-3.01	47.75	54.89	24.89	48.85	-5.09	52.09
Llama3.1	67.82	2.57	56.19	11.84	22.25	58.15	15.59	43.17	0.69	37.34	57.09	15.80	39.87	-2.76	43.73
BioMistral	19.64	38.15	41.48	-19.47	78.88	21.09	45.46	38.26	-19.29	82.59	23.35	42.36	35.84	-20.99	84.41
UniformAvg	63.97	8.20	58.06	0.00	30.79	65.48	9.12	76.33	0.00	30.32	71.06	12.69	82.58	0.00	30.86
SelfCertainty	60.87	9.05	71.25	19.89	33.34	68.81	7.37	78.04	27.88	27.03	75.25	7.46	83.53	36.66	24.16
PackLLM	67.87	10.75	55.85	21.00	27.51	64.99	8.92	41.64	-15.83	31.16	68.46	11.31	38.52	-16.10	32.40
RouterDC	67.42±3.61	9.29±0.91	59.79±0.83	30.02±1.96	27.16±8.06	65.78±0.51	8.28±0.51	59.41±18.5	21.66±13.6	29.83±0.32	71.26±0.43	12.43±0.47	60.92±25.7	22.91±19.1	30.36±0.38
NIRTRouter	70.82±3.04	6.77±0.10	59.79±0.81	30.03±1.91	21.47±2.98	70.08±7.19	4.81±0.11	76.51±1.37	32.92±8.11	25.38±7.51	77.45±5.07	8.87±0.99	83.15±0.00	39.79±0.00	23.58±6.45
RouteLLM	71.67±1.79	2.09±4.02	59.79±0.81	30.03±1.91	18.77±0.39	69.95±8.91	8.92±4.87	76.51±1.37	32.92±8.11	24.56±7.84	78.44±0.00	6.28±3.05	83.15±0.00	39.79±0.00	20.90±3.11
StackedGen	73.13±0.30	1.47±0.59	64.16±0.20	33.91±0.07	17.09±0.16	70.59±1.23	8.42±1.15	76.20±0.58	32.87±1.56	23.62±1.43	77.84±0.26	7.41±1.70	82.96±0.31	39.40±0.18	19.97±0.10
WALLA I	72.44±0.05	2.61±0.18	59.66±0.17	29.82±0.17	17.85±0.05	70.70±0.07	9.78±1.56	76.54±0.05	33.51±0.15	23.99±0.22	78.23±0.32	5.62±0.61	82.51±0.72	39.36±0.39	20.69±0.18
WALLA II	72.43±0.20	2.44±0.24	59.72±0.49	29.85±0.41	17.93±0.24	70.58±0.12	10.15±1.12	76.51±0.09	33.54±0.26	24.18±0.23	77.98±0.16	5.35±0.28	82.61±0.68	39.23±0.27	20.59±0.18
WALLA I log-pool	72.63±0.07	2.22±0.30	59.86±0.50	30.02±0.41	17.65±0.24	70.46±0.33	11.98±1.28	76.41±0.22	33.34±0.92	24.41±0.35	78.13±0.63	7.97±1.83	75.94±1.27	28.08±2.08	20.90±0.07
WALLA II log-pool	72.64±0.12	2.08±0.34	59.72±0.67	29.83±0.54	17.63±0.18	70.39±0.18	12.18±1.04	76.42±0.18	33.38±1.11	24.43±0.22	78.28±0.60	8.20±1.24	81.46±1.52	38.77±0.86	20.41±0.14
WALLA I calibrate	73.27±0.07	10.17±0.14	67.35±0.52	34.12±0.52	16.57±0.10	70.83±0.18	2.89±0.63	70.02±0.04	31.64±0.50	18.43±0.07	77.93±0.00	6.99±2.51	84.21±0.19	38.80±0.44	21.38±0.42

Table 12: Heterogeneous models (Gemma2, Aloe, Llama3.1, and BioMistral) with private contexts on the PubMedQA dataset, with questions that all models can answer correctly filtered out. For each question, we randomly assign one model with a relevant context (left columns) or two models, one with a relevant context and the other with irrelevant context (right columns).

	1 Relevant Context						1 Relevant Context & 1 Irrelevant Context					
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
UniformAvg	62.26	67.04	5.93	61.04	0.00	29.64	65.29	68.43	7.61	65.47	0.00	31.42
SelfCertainty	64.02	68.15	9.67	80.56	25.57	29.57	63.05	64.45	16.27	79.63	25.50	38.47
PackLLM	73.28	77.04	11.34	69.89	22.08	20.68	66.20	71.43	9.66	59.43	7.04	29.60
RouterDC w/ ctx.	62.16±1.03	67.18±1.76	6.04±2.34	54.05±21.8	7.18±13.0	29.58±0.83	65.03±1.62	68.80±3.66	8.11±4.30	50.04±1.33	-0.76±14.6	31.39±0.78
NIRTRouter w/ ctx.	62.06±2.79	67.69±1.18	6.56±8.74	46.88±40.5	0.07±83.4	29.39±2.96	65.08±1.26	69.42±0.00	8.77±3.77	45.02±68.4	-2.09±81.1	31.17±1.96
RouteLLM w/ ctx.	70.38±0.62	70.75±0.82	2.97±0.68	62.28±1.39	19.27±0.41	23.45±0.67	70.39±0.59	72.52±0.80	3.13±0.81	63.46±2.28	17.10±1.38	25.22±0.57
RouteLLM w/o ctx.	62.54±11.9	66.78±0.29	2.74±1.75	54.72±32.9	7.50±6.67	28.87±3.44	64.62±21.3	69.93±4.44	2.90±1.45	61.40±4.40	12.26±14.4	28.98±10.1
StackedGen	78.87±0.39	78.94±0.25	4.54±0.31	73.21±0.22	29.97±0.57	14.51±0.06	77.62±0.76	79.44±2.26	4.14±1.87	72.15±1.23	25.05±2.25	16.65±1.21
WALLA I	77.26±0.55	76.28±0.28	3.04±0.37	71.92±0.19	26.13±0.35	16.87±0.04	77.18±0.32	74.67±0.20	2.58±0.34	70.47±0.06	22.25±0.21	20.17±0.14
WALLA II	76.92±1.20	74.63±2.66	3.27±0.18	70.48±2.91	23.89±3.78	18.20±2.41	75.82±0.62	74.49±0.19	2.21±0.24	70.80±0.17	21.30±0.24	20.84±0.13
WALLA I on emb.	75.49±0.36	73.13±0.41	3.06±0.23	67.86±0.56	21.71±1.35	19.86±0.35	65.35±0.72	66.91±1.05	4.87±1.55	48.66±0.67	4.34±0.32	30.48±0.13
WALLA I log-pool	76.92±0.61	76.35±0.27	3.00±0.50	71.99±0.07	26.01±0.29	16.94±0.06	76.02±0.52	74.59±0.10	2.26±0.38	70.52±0.16	22.23±0.04	20.72±0.18
WALLA I calibrate	85.21±0.28	77.48±0.43	4.57±0.52	76.07±0.17	24.94±0.56	14.72±0.18	83.20±0.15	76.38±0.09	4.27±0.12	73.75±0.32	20.51±0.34	16.10±0.08

Table 13: Heterogeneous models with private contexts on the PubMedQA dataset including Llama3.1-70B. The first row is repeated from Table 4.

	One Relevant Context						One Relevant Context & One Irrelevant Context					
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
WALLA I	78.64±0.28	82.90±0.20	6.29±0.32	70.15±0.35	23.63±0.51	14.61±0.03	78.91±0.23	80.75±0.15	4.78±0.23	68.74±0.13	23.19±0.16	17.28±0.07
WALLA I add 70B	79.20±0.48	82.58±0.31	1.63±0.50	66.46±0.59	23.78±0.20	18.59±0.13	78.53±0.41	79.36±0.36	2.55±1.12	69.26±0.64	21.26±0.30	22.04±0.50
70B w/ context	75.81±0.00	67.43±0.00	24.46±0.00	65.04±0.00	7.35±0.00	37.65±0.00	71.75±0.00	65.64±0.00	17.67±0.00	63.53±0.00	5.63±0.00	41.54±0.00

Table 14: Comparison between hindsight regret minimization and best-response wager MSE minimization on homogeneous models (Aloe) on PubMedQA with private contexts. First two rows repeat Table 2.

	4					8					12				
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	DRegret ↓
WALLA I	82.32±0.02	87.60±0.00	7.35±0.01	88.42±0.24	3.35±0.00	82.74±0.10	87.38±0.03	7.32±0.09	84.63±0.61	3.56±0.04	82.44±0.04	87.47±0.00	7.16±0.01	83.72±1.06	3.57±0.00
WALLA II	82.24±0.03	87.63±0.00	7.44±0.03	88.41±0.16	3.37±0.02	82.74±0.07	87.38±0.03	7.31±0.04	84.97±1.23	3.56±0.03	82.45±0.07	87.47±0.00	7.17±0.04	83.76±0.65	3.57±0.01
WALLA I (BR-MSE)	81.07±0.58	87.00±0.47	9.76±0.33	86.97±1.89	5.19±0.64	84.47±0.28	87.09±0.14	8.35±0.48	81.90±0.86	9.32±0.22	82.10±0.35	86.66±0.32	10.89±0.80	77.14±1.94	12.06±0.62
WALLA II (BR-MSE)	81.07±0.19	86.68±0.11	9.83±0.14	88.46±0.22	5.49±0.05	79.85±0.34	85.95±0.39	10.45±0.29	85.07±1.39	7.15±0.33	78.65±0.60	85.17±0.45	10.61±0.37	83.84±0.57	8.46±0.26

Table 15: Comparison between hindsight regret minimization and best-response wager MSE minimization on homogeneous models (Gemma2) on BayesX with private contexts. The first two rows repeat Table 7.

	4				8				12			
Method	KLD ↓	TVD ↓	MRR ↑	DRegret ↓	KLD ↓	TVD ↓	MRR ↑	DRegret ↓	KLD ↓	TVD ↓	MRR ↑	DRegret ↓
WALLA I	1.99±0.06	8.23±0.13	100.00±0.00	-1.01±0.03	1.96±0.09	8.20±0.17	100.00±0.00	-1.00±0.09	2.23±0.12	8.91±0.23	100.00±0.00	-0.77±0.10
WALLA II	2.01±0.07	8.29±0.16	100.00±0.00	-0.99±0.05	2.27±0.12	8.83±0.24	100.00±0.00	-0.70±0.12	2.23±0.12	8.91±0.23	100.00±0.00	-0.77±0.10
WALLA I (BR-MSE)	2.22±0.11	8.69±0.22	100.00±0.00	-0.74±0.11	5.87±0.50	15.97±0.82	100.00±0.00	2.86±0.49	12.59±0.98	24.01±0.90	100.00±0.00	9.19±0.88
WALLA II (BR-MSE)	2.22±0.07	8.68±0.14	100.00±0.00	-0.74±0.07	5.91±0.08	16.05±0.12	100.00±0.00	2.90±0.08	13.20±1.26	24.57±1.13	100.00±0.00	9.74±1.12

Table 16: Comparison between hindsight regret minimization and best-response wager MSE minimization on heterogeneous models without context on three datasets. The first two rows repeat Table 3.

	MedMCQA (i.d.)					MMLU (i.d.)					ARC-Challenge (o.o.d.)				
Method	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
WALLA I	81.98$_{\pm 0.03}$	3.80$_{\pm 0.11}$	54.64$_{\pm 0.20}$	25.32$_{\pm 0.40}$	12.37$_{\pm 0.03}$	87.41$_{\pm 0.03}$	2.49$_{\pm 0.81}$	82.91$_{\pm 0.07}$	44.55$_{\pm 0.62}$	10.54$_{\pm 0.07}$	90.48$_{\pm 0.14}$	1.88$_{\pm 0.50}$	88.53$_{\pm 0.54}$	45.96$_{\pm 0.33}$	9.25$_{\pm 0.11}$
WALLA II	81.97$_{\pm 0.13}$	3.57$_{\pm 0.47}$	54.77$_{\pm 0.58}$	25.53$_{\pm 1.01}$	12.41$_{\pm 0.10}$	87.35$_{\pm 0.05}$	2.69$_{\pm 0.58}$	83.19$_{\pm 0.34}$	47.68$_{\pm 4.09}$	10.61$_{\pm 0.08}$	90.37$_{\pm 0.07}$	1.77$_{\pm 0.19}$	88.69$_{\pm 0.60}$	46.28$_{\pm 0.76}$	9.19$_{\pm 0.11}$
WALLA I (BR-MSE)	81.17$_{\pm 0.29}$	7.59$_{\pm 0.36}$	55.26$_{\pm 0.55}$	24.17$_{\pm 1.16}$	14.86$_{\pm 0.47}$	86.78$_{\pm 0.49}$	3.34$_{\pm 0.35}$	82.60$_{\pm 0.62}$	45.11$_{\pm 4.05}$	11.89$_{\pm 0.43}$	89.89$_{\pm 0.31}$	6.24$_{\pm 0.39}$	86.76$_{\pm 0.97}$	44.90$_{\pm 1.38}$	11.06$_{\pm 0.14}$
WALLA II (BR-MSE)	81.00$_{\pm 0.34}$	7.85$_{\pm 0.61}$	55.30$_{\pm 0.75}$	21.95$_{\pm 1.94}$	15.36$_{\pm 0.51}$	86.98$_{\pm 0.54}$	3.01$_{\pm 0.38}$	82.52$_{\pm 0.52}$	40.42$_{\pm 7.92}$	11.70$_{\pm 0.49}$	90.28$_{\pm 0.24}$	6.17$_{\pm 0.81}$	88.07$_{\pm 0.75}$	42.23$_{\pm 7.02}$	10.59$_{\pm 0.11}$

Table 17: Comparison between hindsight regret minimization and best-response wager MSE minimization on four heterogeneous models with private contexts on PubMedQA. First two rows repeat Table 4.

	One Relevant Context						One Relevant Context & One Irrelevant Context					
Method	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓	AUC ↑	ACC ↑	ECE ↓	MRR ↑	K-Tau ↑	DRegret ↓
WALLA I	78.64$_{\pm 0.28}$	82.90$_{\pm 0.20}$	6.29$_{\pm 0.32}$	70.15$_{\pm 0.35}$	23.63$_{\pm 0.51}$	14.61$_{\pm 0.03}$	78.91$_{\pm 0.23}$	80.75$_{\pm 0.15}$	4.78$_{\pm 0.23}$	68.74$_{\pm 0.13}$	23.19$_{\pm 0.16}$	17.28$_{\pm 0.07}$
WALLA II	78.62$_{\pm 0.14}$	81.71$_{\pm 1.91}$	4.73$_{\pm 2.39}$	69.75$_{\pm 1.03}$	23.85$_{\pm 0.72}$	15.28$_{\pm 1.10}$	77.89$_{\pm 0.50}$	80.61$_{\pm 0.14}$	3.85$_{\pm 0.35}$	69.51$_{\pm 0.31}$	23.32$_{\pm 0.18}$	17.66$_{\pm 0.06}$
WALLA I (BR-MSE)	79.76$_{\pm 0.65}$	82.11$_{\pm 0.18}$	11.58$_{\pm 0.52}$	72.59$_{\pm 0.75}$	28.06$_{\pm 0.59}$	16.95$_{\pm 0.05}$	77.98$_{\pm 0.37}$	81.15$_{\pm 0.18}$	11.88$_{\pm 0.35}$	74.98$_{\pm 0.81}$	28.38$_{\pm 0.32}$	20.09$_{\pm 0.09}$
WALLA II (BR-MSE)	79.73$_{\pm 0.23}$	82.19$_{\pm 0.31}$	11.30$_{\pm 0.58}$	72.31$_{\pm 0.47}$	27.02$_{\pm 0.59}$	16.72$_{\pm 0.09}$	77.60$_{\pm 0.05}$	80.84$_{\pm 0.24}$	10.81$_{\pm 0.43}$	75.36$_{\pm 0.74}$	26.29$_{\pm 0.78}$	19.93$_{\pm 0.05}$

Table 18: Average wagers and net payoffs in Scenario II when no model is calibrated vs. when Gemma2 is calibrated. When Gemma2 is calibrated, it increases net payoff on both MedMCQA and MMLU. On MedMCQA, it improves performance after calibration, leading to higher wager and net payoffs; on MMLU, it turns net payoffs from negative to positive as it performs best compared to others. Both wagers and net payoffs $\times 100$.

	MedMCQA				MMLU			
	All Uncalibrated		Gemma2 Calibrated		All Uncalibrated		Gemma2 Calibrated	
Method	Wagers	Net Payoffs	Wagers	Net Payoffs	Wagers	Net Payoffs	Wagers	Net Payoffs
Gemma2	13.42$_{\pm 1.33}$	-0.23$_{\pm 0.04}$	19.49$_{\pm 2.16}$	-0.18$_{\pm 0.04}$	47.97$_{\pm 7.60}$	-0.17$_{\pm 0.30}$	51.11$_{\pm 2.74}$	0.07$_{\pm 0.14}$
Llama3.1	42.19$_{\pm 1.26}$	0.08$_{\pm 0.01}$	38.56$_{\pm 3.89}$	0.06$_{\pm 0.01}$	24.13$_{\pm 4.47}$	-0.18$_{\pm 0.09}$	22.93$_{\pm 2.22}$	-0.18$_{\pm 0.10}$
Aloe	42.40$_{\pm 1.72}$	0.09$_{\pm 0.01}$	39.91$_{\pm 2.90}$	0.07$_{\pm 0.02}$	19.89$_{\pm 3.18}$	-0.25$_{\pm 0.12}$	19.26$_{\pm 1.56}$	-0.22$_{\pm 0.08}$
BioMistral	1.99$_{\pm 0.16}$	-0.04$_{\pm 0.00}$	2.04$_{\pm 0.41}$	-0.04$_{\pm 0.01}$	8.01$_{\pm 1.77}$	-0.26$_{\pm 0.06}$	6.70$_{\pm 1.88}$	-0.19$_{\pm 0.07}$

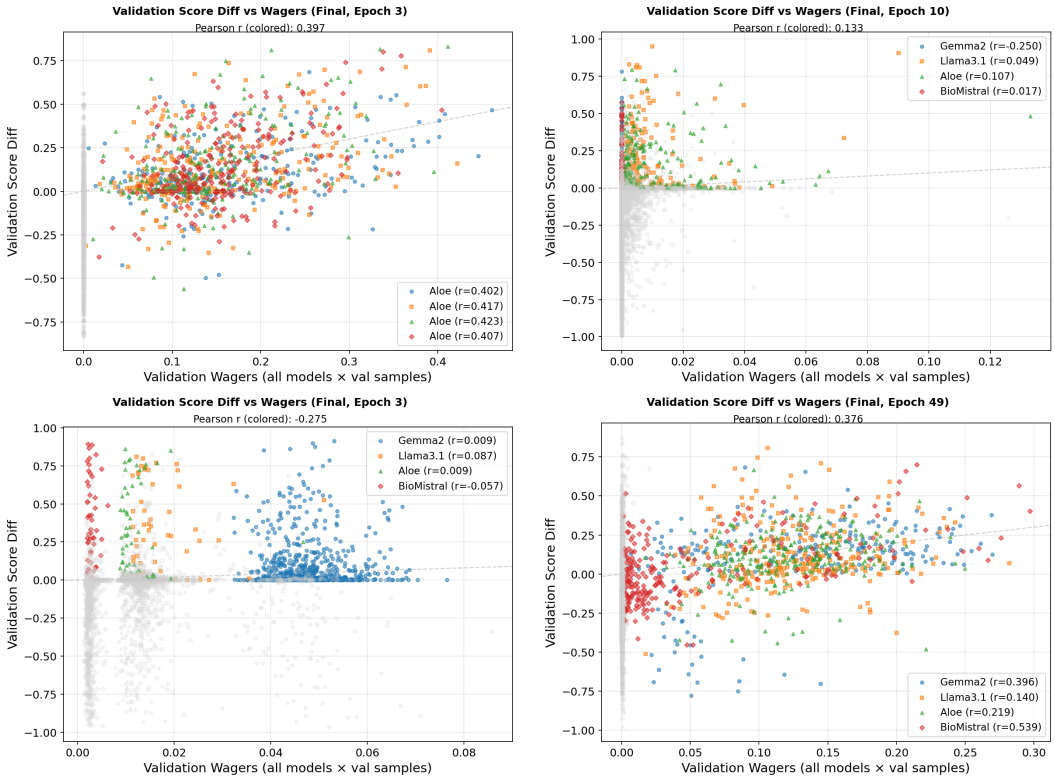


Figure 2: Advantage–wager alignment for WALLA I. Learned wager (x-axis) vs. realized score differentials $s(p_i, y) - b_{-i}$ (y-axis), whose positive parts equal the hindsight optimal wagers under our parameter choice ($c_1 = 1, c_2 = 1/2, c_3 = 1/2$). For Figures 2–5, each panel corresponds to one scenario: homogeneous models on PubMedQA (top-left), heterogeneous models on MedMCQA (top-right), MMLU (bottom-left), and PubMedQA (bottom-right). Colored dots highlight models with context (PubMedQA) or the best-performing model (other datasets). Pearson correlations are reported at the top of each panel. Learned wagers are positively associated with realized score differentials in the two PubMedQA (top-left and bottom-right) settings; the association is weaker on MedMCQA and negative overall on MMLU. On PubMedQA, models successfully differentiate whether they received relevant context, wagering near zero when they did not.

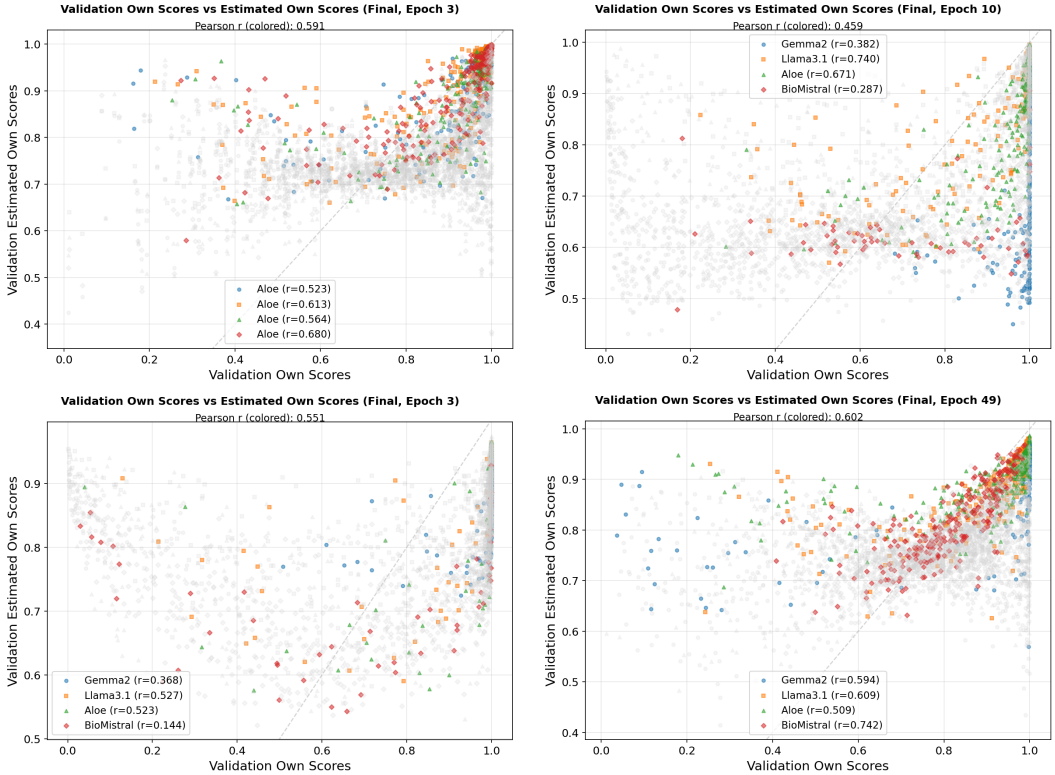


Figure 3: The alignment between the actual Brier scores (x-axis) and the predicted Brier scores trained on last-layer hidden states.

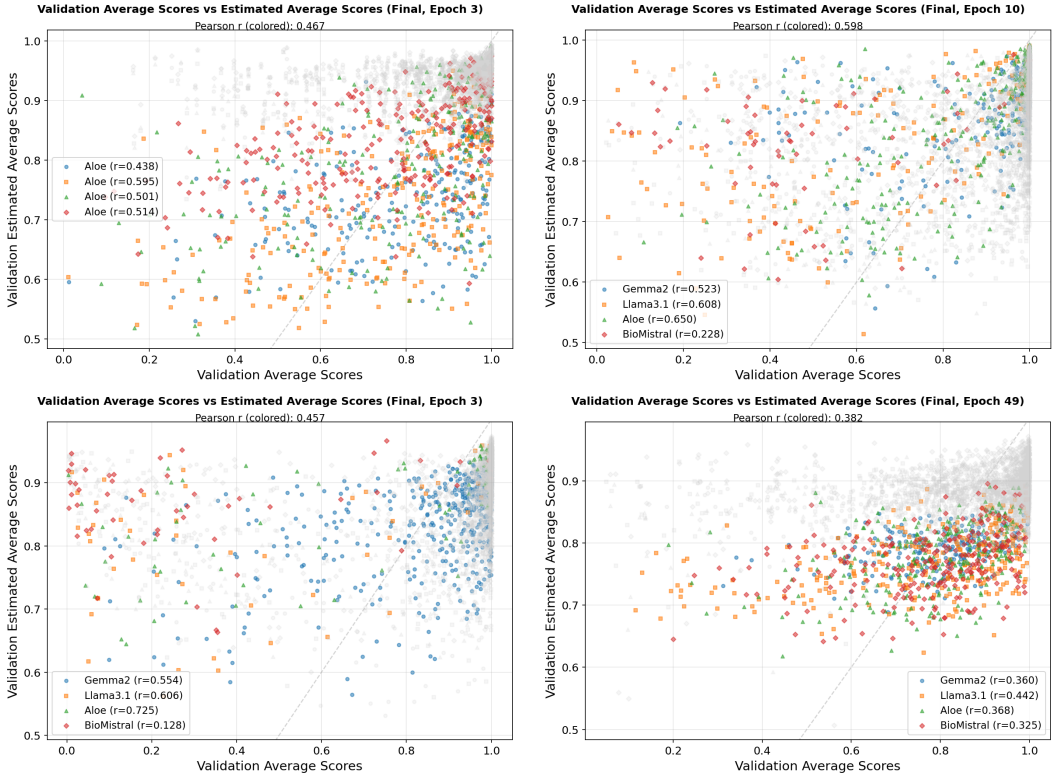


Figure 4: The alignment between the actual baseline average scores (b_{-i}, x-axis) and the predicted baseline average scores trained on last-layer hidden states.

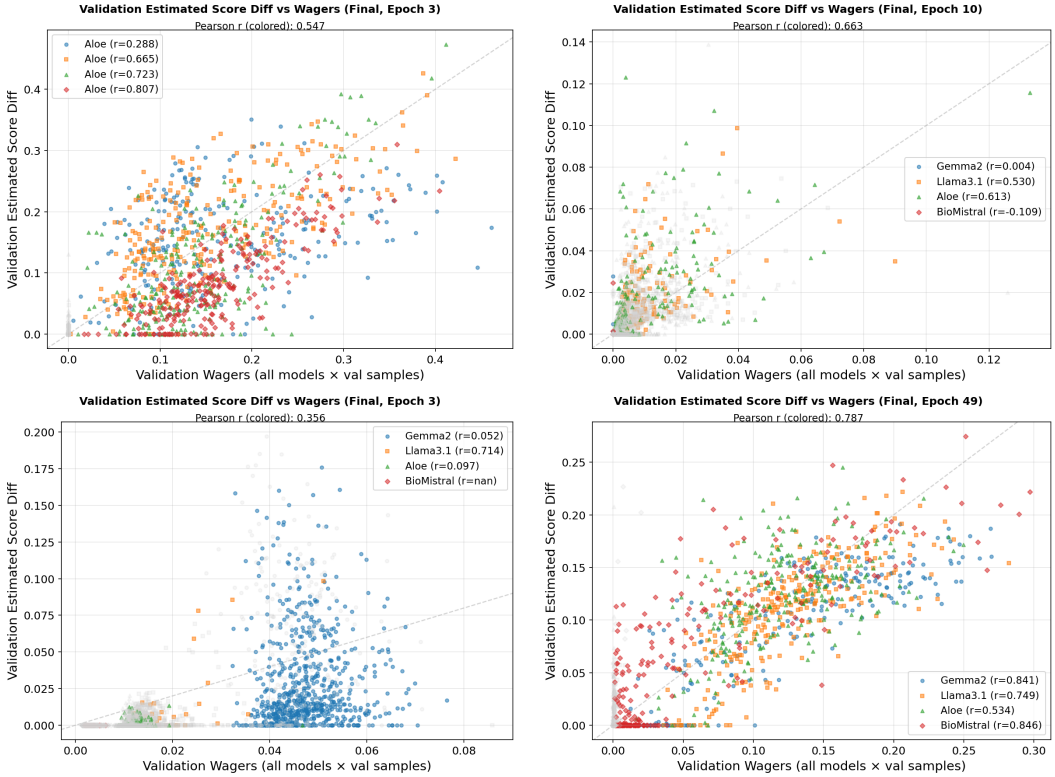


Figure 5: The alignment between the wagers (x-axis) and the differences between the predicted own scores and the predicted baseline average scores (y-axis), which can be interpreted as the estimated expected score advantage, whose positive part gives an estimated best-response wager for $c_3 = 1/2$. The alignment is strong even though the wagers, the predicted own scores, and the predicted baseline average scores are trained with separate heads.



Figure 6: Dynamic regret, mean reciprocal rank of the best expert (MRR), and accuracy (ACC) over time of WALLA under Scenario I on PubMedQA. They demonstrate that WALLA's performance gradually improves over time, though in a decentralized learning environment. A similar trend can be observed in other scenarios or datasets.

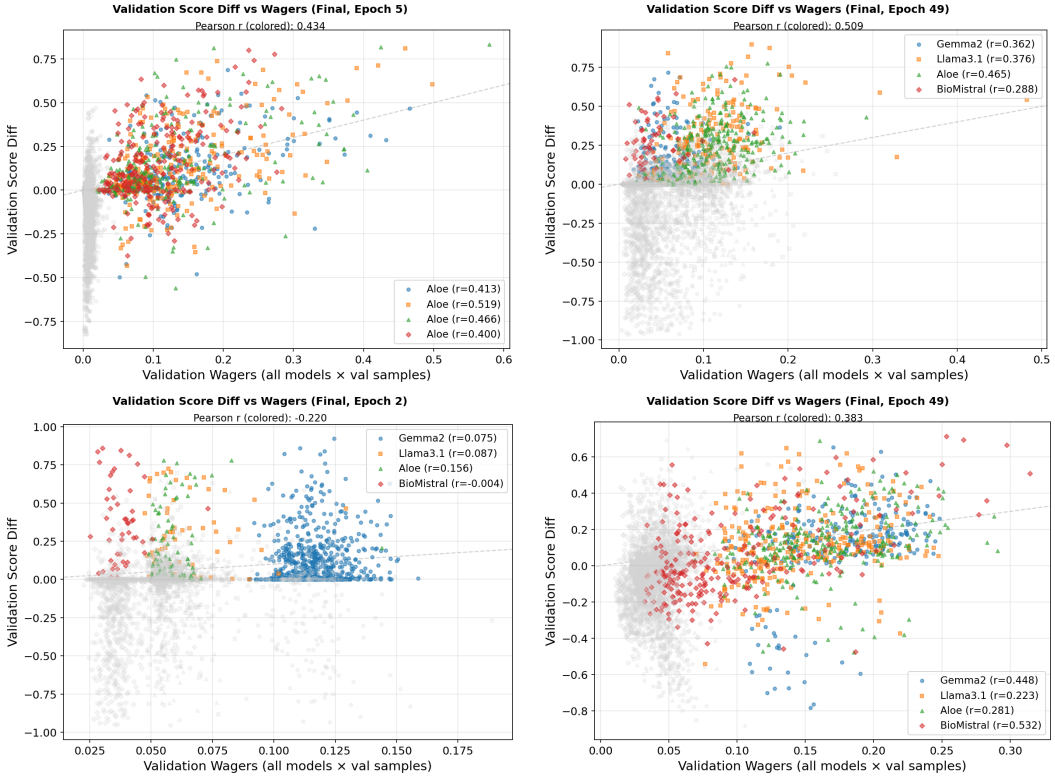


Figure 7: Advantage–wager alignment for WALLA I learning the best-response wager with MSE minimization. Learned wager (x-axis) vs. realized score differentials $s(p_i, y) - b_{-i}$ (y-axis).

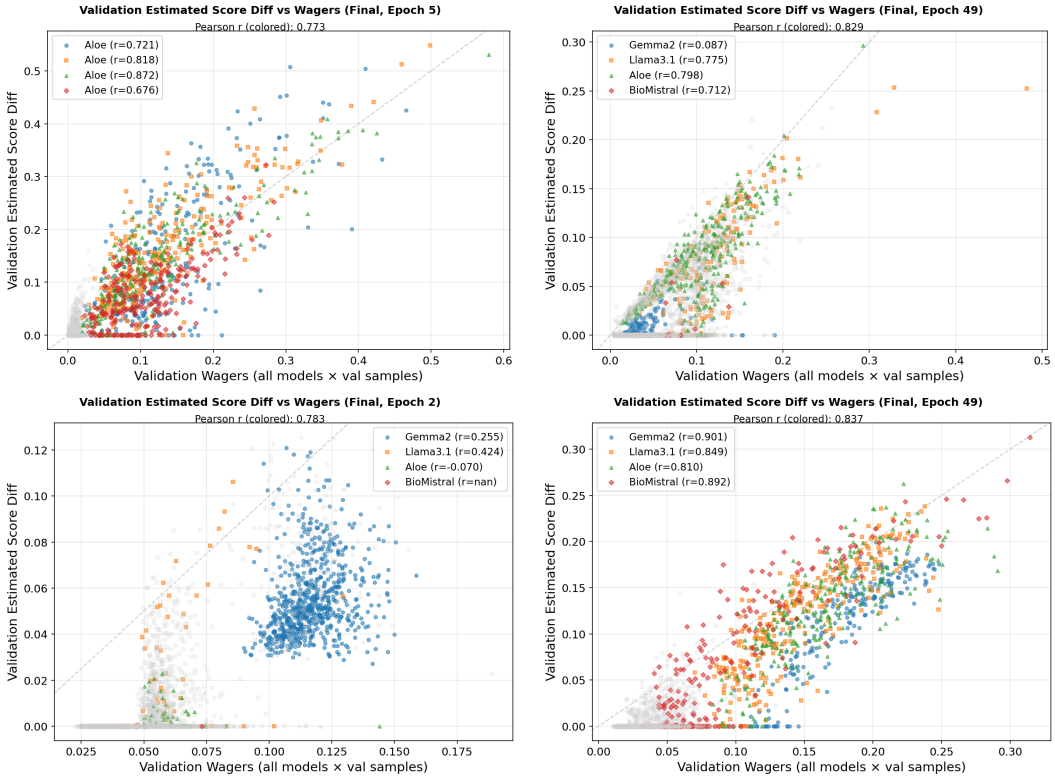


Figure 8: For WALLA I learning the best-response wager with MSE minimization, the alignment between the wagers (x-axis) and the differences between the predicted own scores and the predicted baseline average scores (y-axis) shows stronger correlation than WALLA with hindsight regret minimization.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our abstract and introduction describe the scope and the main contribution to the wagering mechanism design for LLM.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of the work in the proof assumptions and describe scenarios where our mechanism can be applicable.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For each theoretical result, the paper provides the complete set of assumptions and the complete and correct proof.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper fully discloses the source code, the source of datasets, and the commands run for the main experiment results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The paper provides open access to the data and code with sufficient instructions to faithfully reproduce the main experiments.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: the paper specifies all the training and test details including data splits, hyperparameters, how they were chosen, and the type of optimizer.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports 95% confidence intervals for all experiment results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper indicates the type of computing workers, CPU and GPU the experiments run on.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conforms, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses some potential positive societal impact for allowing decentralized collaboration among proprietary large language models.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: the datasets are either publicly available online or are synthesized.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The creators of code, data, and model are properly cited and credited.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We introduce a new synthesized dataset, and it is well-documented.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs are used for our experiment as our subject for evaluation.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.