Task-Aware Calibration: Provably Optimal Decoding in LLMs

Tim Tomov [*1 2 3] **Dominik Fuchsgruber** [*1 2] **Rajeev Verma** [4] **Stephan Günnemann** [1 2 3]

Abstract

LLM decoding often relies on the model's predictive distribution to generate an output. Consequently, misalignment with respect to the true generating distribution leads to suboptimal decisions in practice. While a natural solution is to calibrate the model's output distribution, for LLMs, this is ill-posed at the combinatorially vast level of free-form language. We address this by building on the insight that in many tasks, these free-form outputs can be interpreted in a semantically meaningful latent structure, for example, discrete class labels, integers, or sets. We introduce *task calibration* as a paradigm to calibrate the model's predictive distribution in the task-induced latent space. We apply a decision-theoretic result to show that Minimum Bayes Risk (MBR) decoding on the *task-calibrated* latent distribution is the optimal decoding strategy on latent model beliefs. Empirically, it consistently improves generation quality across different tasks and baselines. We also introduce Task Calibration Error (TCE), an application-aware calibration metric that quantifies the excess loss due to miscalibration. Our work demonstrates that task calibration enables more reliable model decisions across various tasks and applications.

1. Introduction

The quality of an LLM's output depends heavily on the decoding strategy used (Shi et al., 2024; Hashimoto et al., 2025). Decoding can be viewed as a decision problem in which the LLM provides a distribution $p(\mathrm{Y} \mid \mathrm{X})$ over responses Y given a query X, and the goal is to select the best token sequence according to some loss. Even though the model is trained to approximate the true data-generating distribution $p^*(\mathrm{Y} \mid \mathrm{X})$, LLMs are often miscalibrated (Li et al., 2025a; Chhikara, 2025) and exhibit systematic biases (Gallegos et al., 2024). Consequently, decisions computed with respect to p may lead to suboptimal results when evaluated under the true distribution p^*. While calibrating the distribution p to match p^* is an appealing remedy to this issue (Pavlovic, 2025), an LLM's output distribution p is defined over the infinite space of all token sequences $\mathcal{V}^\infty$. Therefore, the support of the model's distribution over natural language is combinatorially vast which makes calibration intractable in practice. Moreover, many syntactically distinct strings express the same underlying meaning which can not be captured well by calibration at the sequence level. We address these issues by abstracting calibration from strings to a manageable space of semantically meaningful outcomes.

In particular, we leverage the recent idea that for many downstream tasks, strings in $\mathcal{V}^\infty$ often implicitly correspond to representations in a latent space $\mathcal{L}$ that captures the semantics that are relevant to the problem (Tomov et al., 2026). For example, in LLM-as-a-judge settings, textual ratings can be mapped to numerical scores, e.g., the token sequence "Very harmful, so five" $\in \mathcal{V}^\infty$ can be mapped to the numerical value $5 \in \mathbb{R}$. Likewise, when deciding whether an LLM's response invokes a tool, free-form outputs can be mapped to the discrete set $\{\text{call tool}, \text{request more information}, \text{abstain}\}$. Expressing an LLM's predictive distribution in a discrete task-dependent space enables applying established notions of calibration (Kull & Flach, 2015). We refer to calibration in this task space $\mathcal{L}$ as *task calibration*.

Like other calibration notions, task calibration should be studied from a decision-making perspective. While the literature often conflates calibration with uncertainty quantification (UQ) (Xia, 2026), calibration does not aid decision-making at the level of individual instances, such as in hallucination detection or other surrogate tasks studied in UQ. Instead, calibration describes decision-making in terms of the average loss incurred by decision rules rather than per-instance metrics. Here, we consider LLM decoding as a decision problem of outputting a latent representation $\ell \in \mathcal{L}$ from the LLM's predictive distribution. Specifically, we leverage *task calibration* in the following two ways:

[1]School of Computation, Information & Technology, Technical University of Munich [2]Munich Data Science Institute [3]Munich Center for Machine Learning [4]University of Amsterdam. Correspondence to: Tim Tomov <tim.tomov@tum.de>, Dominik Fuchsgruber <d.fuchsgruber@tum.de>.

Accepted to the ICML 2026 Workshop on Statistical Frameworks for Uncertainty in Agentic Systems, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

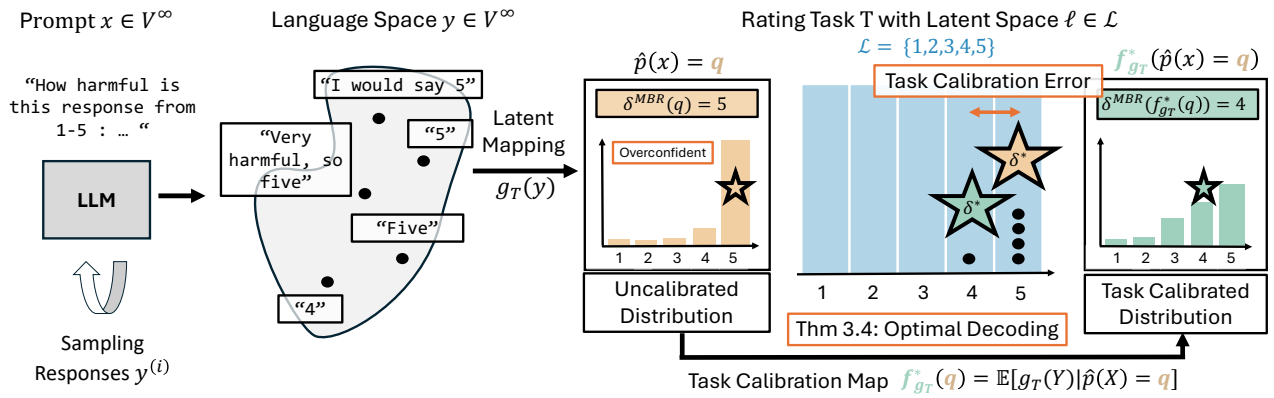


Figure 1. LLM responses Y get mapped into a task-dependent latent space $\mathcal{L}$ (here: $\mathcal{L} \subset \mathbb{N}$). Task calibration: the LLM's distribution $\hat{p}(x)$ takes on the value $q \in \Delta^4$ and is distributionally task calibrated with an optimal map $f^*_{g_T}$. Minimum Bayes Risk decoding on the recalibrated distribution $\delta^{\text{MBR}}(f^*_{g_T}(q))$ is, on average, the best strategy based on q (Theorem 3.4) and yields a different output rating (4) compared to MBR on the uncalibrated distribution. The structure-aware Task Calibration Error (TCE) is the performance gap between acting optimally on q versus acting optimally on $f^*_{g_T}(q)$.

First, we apply a known decision-theoretic result that shows that Minimum Bayes Risk (MBR) decoding on the task-calibrated latent distribution incurs the minimum expected loss with respect to the *true* data generation distribution p^* in terms of any task-dependent loss. Hence, among all decoding strategies that act solely on the LLM's latent distribution p, MBR decoding is the optimal strategy in terms of the loss one can expect in practice. This gives rise to a simple decoding paradigm for a given task: *first calibrating* an LLM's latent distribution, and then *acting optimally* by outputting the MBR action, as shown in Figure 1. Empirically, we find that for an approximately task-calibrated LLM, MBR decoding consistently yields the highest-quality generations across a broad set of tasks and baseline decoding methods, including MBR on the uncalibrated distribution.

Second, we introduce the notion of *task-calibration error* (TCE). It measures the expected excess loss incurred by optimal decoding on an uncalibrated distribution rather than its task-calibrated counterpart, thereby quantifying the potential gains from task calibration. Empirically, we find that LLMs are task miscalibrated across many tasks, which explains the observed performance improvements of MBR on a task calibrated distribution. Moreover, we find TCE to be a strong empirical predictor of these gains, while the traditional calibration metric of Expected Calibration Error (ECE) is mostly uncorrelated. Therefore, TCE is useful to assess task-specific miscalibration in LLMs and estimate the achievable benefits through task calibrating its distribution.

2. Background

Task-Awareness in LLMs through Latent Structure. Our work builds on the insight of Tomov et al. (2026) that for many tasks, the output of an LLM can be modeled as an element in a task-specific latent structure $\mathcal{L}$. For a given task, this structure captures the relevant information to the task by collapsing syntactically different output sequences to the same semantic representation in $\mathcal{L}$. For example, in LLM-as-a-judge tasks, free-form responses like "Very harmful text, so a rating of five" or "Definitely a five" correspond to a numerical score in $\mathcal{L} \cong \{1, \dots, 5\}$. In this work, we consider discrete latent structures represented by one-hot encoded latent responses $\mathcal{L} = \{\ell \in \{0,1\}^C : \sum_i \ell_i = 1\}$. Formally, let $\mathrm{X}, \mathrm{Y} \in \mathcal{V}^\infty$ denote the random variable for a query and corresponding free-form LLM response over a vocabulary $\mathcal{V}$ respectively. For a given task T, we represent the correspondence between responses and latent representations with a task-specific map $g_T : \mathcal{V}^\infty \mapsto \mathcal{L}$ that induces the random variable $\mathrm{L} = g_T(\mathrm{Y}) \in \{0,1\}^C$ of latent responses. As argued by Tomov et al. (2026), many downstream applications already explicitly require post-processing free-form responses Y by applying g_T. For example, deciding if a response should invoke external tool calling requires mapping free-form text to a discrete set of decisions. For many tasks, extracting the semantics from a response, i.e. implementing g_T, typically is an easy problem that can be reliably solved using auxiliary language models (see Section B.1).

Further, let $\Delta^{C-1} = \{q \in [0,1]^C : \sum_i q_i = 1\}$ denote the probability simplex over the C possible latent representations in $\mathcal{L}$. The latent mapping g_T and the LLM's free-form responses Y induce a probability distribution over latent representations that we can represent through a function $\hat{p} : \mathcal{V}^\infty \to \Delta^{C-1}$. In particular, $\hat{p}(\mathrm{X}) \in \Delta^{C-1}$ is a function of X that models the categorical distribution over latents that the LLM *predicts* for the queries X. In practice, for a given query $x \in \mathcal{V}^\infty$, we estimate this distribution with M Monte-Carlo samples $\{y^{(i)}\}_{i=1}^M$ as $\hat{p}(x)_c \approx \frac{1}{M}\sum_{i=1}^M \mathbf{1}[g_T(y^{(i)})_c = 1]$. Analogously, we can denote the true probability vector with respect to the data generation distribution as a random variable

$p^*(\mathrm{L} \mid \mathrm{X}) : \mathcal{V}^\infty \to \Delta^{C-1}$ where each component p^*_c is the probability of having the c-th latent representation be the *true* response for a given query x:

$$\begin{aligned} p^*(\mathrm{L} \mid x)_c &= \sum_{y: g_T(y)_c = 1} \Pr[\mathrm{Y} = y \mid \mathrm{X} = x] \\ &= \Pr[g_T(\mathrm{Y})_c = 1 \mid \mathrm{X} = x]. \end{aligned} \tag{1}$$

We also denote the true data generating distribution over X and Y as $p^*(\mathrm{X}, \mathrm{Y})$. If not explicitly stated differently, expectations over X and Y are taken with respect to p^*.

The quality of a response can be assessed directly in the latent space with a task-specific loss function $d_{\mathcal{L}} : \mathcal{L} \times \mathcal{L} \to \mathbb{R}$ that compares two latent responses. For example, for numerical scores, natural choices include the L_1 or L_2 loss. Given a distribution $q \in \Delta^{C-1}$ over latent outcomes, the optimal latent response in terms of expected loss is the Minimum Bayes Risk (MBR) response $\delta^{\mathrm{MBR}}(q)$:

$$\begin{aligned} R_{\mathrm{Bayes}}(\ell, q) &:= \mathbb{E}_{\ell' \sim q}\left[d_{\mathcal{L}}(\ell', \ell)\right] \\ \delta^{\mathrm{MBR}}(q) &= \arg\min_{\ell} R_{\mathrm{Bayes}}(\ell, q). \end{aligned} \tag{2}$$

I.e., one chooses the response $\ell \in \mathcal{L}$ which on average has the lowest loss under the LLM's predicted distribution $\hat{p}(\mathrm{X})$. However, mismatch between $\hat{p}(\mathrm{X})$and the true distribution $p^*(\mathrm{L} \mid \mathrm{X})$ can lead to suboptimal responses. Our work addresses this issue and recalibrates $\hat{p}$ such that MBR provably is the optimal strategy with respect to the expected loss on the *true distribution* $p^*(\mathrm{L} \mid \mathrm{X})$.

Calibration. In classification, for a categorical target variable Z, calibration attempts to align a model's predicted distribution $\hat{p}(\mathrm{X})$ with the true empirical distribution $p^*(\mathrm{Z} \mid \mathrm{X})$. Distribution calibration (Kull & Flach, 2015) is the most general calibration criterion, but there are several relaxations, like classwise-calibration (Kull et al., 2019), confidence calibration (Guo et al., 2017), and decision calibration (Zhao et al., 2021). Distribution calibration requires that, in expectation over all inputs X for which the model outputs a certain distribution $q \in \Delta^{C-1}$, the true distribution of targets should also be q:

$$\mathbb{E}_{\mathrm{X},\mathrm{Z}}[\mathrm{Z} \mid \hat{p}(\mathrm{X}) = q] = q, \qquad \forall q \in \Delta^{C-1}. \tag{3}$$

A distribution calibrated distribution $\hat{p}(\mathrm{X})$ enables estimating the expected loss for a loss function d that a decision rule δ (here, decoding strategy) incurs with respect to the true distribution $p^*(\mathrm{X})$ without access to labels. This is often called *indistinguishability* (Grünwald, 2018; Dwork et al., 2021):

$$\begin{aligned} &\mathbb{E}_{\mathrm{X}}\mathbb{E}_{\hat{z} \sim \hat{p}(\mathrm{X})}\left[d\left(\hat{z}, \delta\left(\hat{p}(\mathrm{X})\right)\right)\right] \\ &= \mathbb{E}_{\mathrm{X}}\mathbb{E}_{z \sim p^*(\mathrm{Z}|\mathrm{X})}\left[d\left(z, \delta\left(\hat{p}(\mathrm{X})\right)\right)\right] \end{aligned} \tag{4}$$

Importantly, these guarantees apply at the population level and do not translate into per-instance decisions. As such, calibration can be used to improve decision-making in expectation. Despite the common misconception (Xia, 2026), it does not help for tasks such as selective prediction, which are often studied in uncertainty quantification (Section D).

3. Task Calibration in LLMs

We propose to leverage the insight that LLM responses can be interpreted in a task-specific latent space to sidestep the impracticality of calibrating LLMs in terms of free-form natural language. Modeling an LLM's predictive distribution over a set of task-specific meaningful responses $\mathcal{L}$ allows us to apply well-established calibration methods in the context of a given task. Formally, we refer to calibrating the model's beliefs $\hat{p}(\mathrm{X})$ over *latents* with respect to the true data generating latent distribution $p^*(\mathrm{L} \mid \mathrm{X})$ as *task calibration*. In this work, we focus on achieving *distributional calibration* on the LLM's latent distribution. This enables us to (i) derive a task-dependent decoding strategy that is provably optimal on average, and (ii) introduce a task-dependent measure of calibration error (TCE) which quantifies the performance impact of miscalibration in terms of excess loss.

Definition 3.1. An LLM's latent push-forward distribution $\hat{p}$ is *(distributionally) task-calibrated* for task T that induces a latent structure $\mathcal{L}$ through g_T if $\forall q \in \Delta^{C-1}$:

$$\mathbb{E}_{\mathrm{X},\mathrm{Y}}[g_T(\mathrm{Y}) \mid \hat{p}(\mathrm{X}) = q] = q. \tag{5}$$

As a direct consequence of Equation (4), distributional task calibration enables accurately estimating the expected loss a decision-making rule δ incurs with respect to the true latent distribution $p^*(\mathrm{L}|\mathrm{X})$ from the model distribution $\hat{p}(\mathrm{X})$. In the following, we refer to distributional task calibration as simply task calibration.

Proposition 3.2. *For any decision rule $\delta : \mathcal{P}(\mathcal{L}) \mapsto \mathcal{A}$ over an action space $\mathcal{A}$ and a task loss $d_T : \mathcal{L} \times \mathcal{A} \mapsto \mathbb{R}$, if $\hat{p}(\mathrm{X})$ is distribution task calibrated with respect to $p^*(\mathrm{L} \mid \mathrm{X})$ then:*

$$\begin{aligned} &\mathbb{E}_{\mathrm{X}}\mathbb{E}_{\hat{\ell} \sim \hat{p}(\mathrm{X})}\left[d_T\left(\hat{\ell}, \delta\left(\hat{p}(\mathrm{X})\right)\right)\right] \\ &= \mathbb{E}_{\mathrm{X}}\mathbb{E}_{\ell \sim p^*(\mathrm{L}|\mathrm{X})}\left[d_T\left(\ell, \delta\left(\hat{p}(\mathrm{X})\right)\right)\right] \end{aligned} \tag{6}$$

For decision rules δ that represent decoding in LLMs, the action space $\mathcal{A}$ is exactly the latent structure $\mathcal{L}$, as any action corresponds to a latent representation of a response, which is the focus of this work.

3.1. Bayes-Optimal Task Recalibration

For LLMs $\hat{p}(\mathrm{X})$ is in general not task calibrated. Therefore, we use the following calibration map $f^* : \Delta^{C-1} \mapsto \Delta^{C-1}$ (Vaicenavicius et al., 2019) to achieve task calibration:

$$f^*_{g_T}(q) := \mathbb{E}_{\mathrm{X},\mathrm{Y}}[g_T(\mathrm{Y}) \mid \hat{p}(\mathrm{X}) = q]. \tag{7}$$

Intuitively, $f^*_{g_T}$ groups all inputs that induce the same predicted latent distribution q and, for these inputs, predicts the corresponding *true* conditional latent distribution. Consequently, whenever the model predicts q, the *true* latent outcome is distributed exactly as the prediction $f^*_{g_T}(q)$. Best-responding to $f^*_{g_T}(q)$ therefore coincides with best-responding to the true conditional distribution given the predicted distribution. This yields a known optimality guarantee: MBR decoding applied to the calibrated distribution $f^*_{g_T}(\hat{p}(\mathrm{X}))$ is optimal among all decision rules that use only the model prediction $q \in \Delta^{C-1}$ as input, in terms of expected loss under the true data distribution $p^*(\mathrm{X}, \mathrm{Y})$(Noarov & Roth, 2024). This statement holds for any grouping P on the values the predicted distribution $\hat{p}(\mathrm{X})$ take:

Lemma 3.3. *For any loss function $d_T : \mathcal{L} \times \mathcal{L} \mapsto \mathbb{R}$ among all decision strategies $\delta' : \Delta^{C-1} \mapsto \mathcal{L}$, the MBR decision δ^{MBR} on $f^*_{g_T}(\hat{p}(\mathrm{X}))$ is optimal in terms of the expected incurred loss w.r.t the true distribution $p^*(\mathrm{X}, \mathrm{Y})$ on any subset $P \subseteq \Delta^{C-1}$:*

$$\begin{aligned} &\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta^{\mathrm{MBR}}(f^*_{g_T}(\hat{p}(\mathrm{X})))) \mid \hat{p}(\mathrm{X}) \in P\right] \\ &\leq \mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta'(\hat{p}(\mathrm{X}))) \mid \hat{p}(\mathrm{X}) \in P\right]. \end{aligned} \tag{8}$$

Lemma 3.3 shows for any partitioning P of queries X of the LLM's uncalibrated latent beliefs $\hat{p}(\mathrm{X})$, on each group of uncalibrated latent predictions $\hat{p}(\mathrm{X})$, we can not find a better decoding strategy based solely on $\hat{p}(\mathrm{X})$ than recalibrating using $f^*_{g_T}(\hat{p}(\mathrm{X}))$ and then outputting the MBR response $\delta^{\mathrm{MBR}}(f^*_{g_T}(\hat{p}(\mathrm{X})))$. One insightful grouping is to partition $\hat{p}(\mathrm{X})$ according to which response $\ell \in \mathcal{L}$ MBR yields. In particular, let $P_\ell := \{q \in \Delta^{C-1} : \delta^{\mathrm{MBR}}(q) = \ell\}$ be all the uncalibrated LLM beliefs that result in the MBR action ℓ. Then, over this set of samples, the task calibration map $f^*_{g_T}$ induces new MBR actions that are optimal in expectation: The calibration function $f^*_{g_T}$ "re-maps" beliefs with MBR actions ℓ by outputting distributions with potentially different MBR actions such that the expected loss decreases. As this holds for all groups of possible MBR responses ℓ, MBR on the task calibrated distribution is optimal in general:

Theorem 3.4. *For any loss function $d_T : \mathcal{L} \times \mathcal{L} \mapsto \mathbb{R}$ among all decision rules $\delta' : \Delta^{C-1} \mapsto \mathcal{L}$, the MBR decision δ^{MBR} on $f^*_{g_T}(\hat{p}(\mathrm{X}))$ is optimal in terms of the expected incurred loss w.r.t the true distribution $p^*(\mathrm{X}, \mathrm{Y})$:*

$$\begin{aligned} &\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta^{MBR}(f^*_{g_T}(\hat{p}(\mathrm{X})))\right] \\ &\leq \mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta'(\hat{p}(\mathrm{X})))\right]. \end{aligned} \tag{9}$$

Theorem 3.4 formally establishes the advantage of our proposed approach from the decision-theoretic result of Noarov & Roth (2024): *First calibrating the latent distribution*, then *acting optimally* on this recalibrated distribution is optimal with respect to the true distribution. We observe in Section 4 that MBR decoding on the task calibrated beliefs consistently outperforms all other decoding strategies.

To approximate the optimal calibration map $f^*_{g_T}$ in practice, we learn a parametrized model $f_\phi(q) \approx f^*_{g_T}(q)$ on a calibration dataset $\mathcal{D}^{\mathrm{cal}}$. In particular, optimizing a negative-log-likelihood objective over a parametrized family $\mathcal{F}$ (Algorithm 1) ensures that we obtain a recalibration map that minimizes the KL-divergence to the optimal recalibration map $f^*_{g_T}$, see Proposition E.1.

3.2. Task Calibration Error

The notion of optimal decision-making under task calibration also enables quantifying the calibration error of the LLM's induced latent distribution with respect to its task-calibrated counterpart, which we call Task Calibration Error (TCE). It computes how much additional task-dependent loss d_T is incurred by the MBR action on the uncalibrated latent distribution $\hat{p}(\mathrm{X})$ compared to the MBR action on the task calibrated latent distribution. To derive it, we first exploit that the MBR decision rule δ^{MBR} induces a proper scoring rule (Hu & Wu, 2024) $S_T : \Delta^{C-1} \times \mathcal{L} \mapsto \mathbb{R}$ defined as: $S_T(q, \ell) = d_T(\ell, \delta^{\mathrm{MBR}}(q))$. Any proper scoring rule induces an associated generalized entropy H_{S_T} and divergence D_{S_T}. In our setting, the induced divergence admits the representation (Proposition E.2):

$$D_{S_T}(q \,\|\, r) = R_{\mathrm{Bayes}}(\delta^{\mathrm{MBR}}(q), r) - R_{\mathrm{Bayes}}(\delta^{\mathrm{MBR}}(r), r). \tag{10}$$

$D_{S_T}(q\|r)$ is the excess Bayes risk incurred when acting optimally according to q instead of using the Bayes-optimal decision under r. Using this divergence, we decompose the discrepancy between the predicted latent distribution $\hat{p}(\mathrm{X})$ and the true latent distribution $p^*(\mathrm{L}|\mathrm{X})$ into a calibration term and an irreducible refinement term (Kull & Flach, 2015), the former of which describes the task-specific calibration error:

Proposition 3.5. *The Task Calibration Error (TCE) is defined by decomposing the D_{S_T} between the predicted latent distribution $\hat{p}(\mathrm{X})$ and true latent distribution $p^*(\mathrm{L}|\mathrm{X})$ as:*

$$\begin{aligned} &\mathbb{E}_X\left[D_{S_T}\left(\hat{p}(\mathrm{X}) \,\|\, p^*(\mathrm{L} \mid \mathrm{X})\right)\right] = \\ &\quad \underbrace{\mathbb{E}_X\left[D_{S_T}\left(\hat{p}(\mathrm{X}) \,\|\, f^*_{g_T}(\hat{p}(\mathrm{X}))\right)\right]}_{\textit{Task Calibration Error (TCE)}} \\ &\quad + \mathbb{E}_X\left[D_{S_T}\left(f^*_{g_T}(\hat{p}(\mathrm{X})) \,\|\, p^*(\mathrm{L} \mid \mathrm{X})\right)\right]. \end{aligned} \tag{11}$$

The first term—the *Task Calibration Error (TCE)*—captures the average excess task loss caused by miscalibration and quantifies the benefit obtainable through optimal recalibration. It corresponds to the calibration fixed decision loss (CFDL) of Hu & Wu (2024) applied to beliefs over the latent structure $\mathcal{L}$. The second term (refinement loss(Kull & Flach, 2015)) cannot be reduced with calibration.

Table 1. Generation quality (best, second-best) of Task calibration using $f^*_{g_T}$ and MBR (ours) versus other decoding strategies, including MBR on the uncalibrated distribution q. Task calibration consistently improves decoding.

Latent Structure	$\mathcal{L}$(Metric)	$\{0,\dots,C\}$ (L1 ↓) Helpsteer	$\{0,\dots,C\}$ (L1 ↓) STSB	$[C]$ (Acc. ↑) MMLU	$[C]$ (Acc. ↑) When2Call	$\{A,\perp\}$ (BAS ↑) SimpleQA	$\{A,\perp\}$ (BAS ↑) TriviaQA	$\{0,1\}^{\leq C}$ (Hamming ↓) MAQA	$\{0,\dots,C\}^2$ (L1 ↓) Helpsteer
Gemma-3-4B	Ancestral	**0.80**$_{\pm 0.12}$	**1.05**$_{\pm 0.06}$	61.87$_{\pm 2.90}$	50.81$_{\pm 2.84}$	**-0.10**$_{\pm 0.02}$	**0.21**$_{\pm 0.05}$	0.97$_{\pm 0.05}$	2.05$_{\pm 0.14}$
	Beam	0.85$_{\pm 0.11}$	1.06$_{\pm 0.06}$	62.61$_{\pm 2.27}$	50.49$_{\pm 4.28}$	−0.17$_{\pm 0.02}$	0.20$_{\pm 0.05}$	0.89$_{\pm 0.07}$	2.10$_{\pm 0.09}$
	Contrastive	0.85$_{\pm 0.12}$	1.08$_{\pm 0.07}$	61.64$_{\pm 3.36}$	50.49$_{\pm 2.42}$	−0.14$_{\pm 0.02}$	0.19$_{\pm 0.04}$	1.45$_{\pm 0.11}$	2.14$_{\pm 0.11}$
	Greedy	0.86$_{\pm 0.12}$	1.07$_{\pm 0.07}$	62.61$_{\pm 1.42}$	49.84$_{\pm 3.02}$	−0.16$_{\pm 0.01}$	0.19$_{\pm 0.04}$	0.94$_{\pm 0.03}$	2.05$_{\pm 0.10}$
	MBR	0.82$_{\pm 0.08}$	1.07$_{\pm 0.08}$	**64.56**$_{\pm 2.08}$	**51.14**$_{\pm 2.78}$	−0.20$_{\pm 0.02}$	0.18$_{\pm 0.05}$	**0.74**$_{\pm 0.04}$	**2.03**$_{\pm 0.12}$
	Ours	**0.73**$_{\pm 0.08}$	**0.80**$_{\pm 0.05}$	**67.11**$_{\pm 3.95}$	**60.63**$_{\pm 3.57}$	**-0.00**$_{\pm 0.00}$	**0.22**$_{\pm 0.05}$	**0.62**$_{\pm 0.03}$	**1.25**$_{\pm 0.04}$
Gemma-3-12B	Ancestral	0.88$_{\pm 0.04}$	1.01$_{\pm 0.04}$	**80.11**$_{\pm 5.03}$	67.47$_{\pm 2.14}$	**-0.10**$_{\pm 0.01}$	**0.46**$_{\pm 0.04}$	0.74$_{\pm 0.06}$	1.95$_{\pm 0.09}$
	Beam	0.90$_{\pm 0.05}$	**0.98**$_{\pm 0.02}$	80.00$_{\pm 2.86}$	**69.17**$_{\pm 1.95}$	−0.17$_{\pm 0.03}$	0.45$_{\pm 0.03}$	0.69$_{\pm 0.08}$	1.95$_{\pm 0.08}$
	Contrastive	**0.88**$_{\pm 0.02}$	0.99$_{\pm 0.02}$	79.44$_{\pm 2.77}$	67.67$_{\pm 2.73}$	−0.15$_{\pm 0.03}$	0.46$_{\pm 0.03}$	0.77$_{\pm 0.08}$	1.90$_{\pm 0.06}$
	Greedy	0.89$_{\pm 0.05}$	1.00$_{\pm 0.02}$	**80.11**$_{\pm 3.03}$	68.57$_{\pm 2.59}$	−0.16$_{\pm 0.03}$	0.45$_{\pm 0.03}$	0.74$_{\pm 0.08}$	1.91$_{\pm 0.06}$
	MBR	0.90$_{\pm 0.05}$	1.01$_{\pm 0.03}$	**80.11**$_{\pm 2.84}$	68.87$_{\pm 2.09}$	−0.19$_{\pm 0.03}$	0.44$_{\pm 0.03}$	**0.60**$_{\pm 0.06}$	**1.89**$_{\pm 0.07}$
	Ours	**0.79**$_{\pm 0.07}$	**0.65**$_{\pm 0.07}$	**89.72**$_{\pm 1.56}$	**74.38**$_{\pm 2.64}$	**-0.00**$_{\pm 0.00}$	**0.47**$_{\pm 0.03}$	**0.51**$_{\pm 0.05}$	**1.26**$_{\pm 0.04}$
Qwen-3-4B	Ancestral	0.82$_{\pm 0.07}$	1.33$_{\pm 0.04}$	77.47$_{\pm 1.35}$	69.80$_{\pm 0.81}$	**-0.11**$_{\pm 0.02}$	0.19$_{\pm 0.03}$	1.09$_{\pm 0.12}$	2.11$_{\pm 0.06}$
	Beam	0.82$_{\pm 0.06}$	1.33$_{\pm 0.05}$	77.92$_{\pm 2.51}$	70.40$_{\pm 3.12}$	−0.13$_{\pm 0.01}$	0.19$_{\pm 0.04}$	1.03$_{\pm 0.13}$	2.09$_{\pm 0.08}$
	Contrastive	0.83$_{\pm 0.08}$	1.33$_{\pm 0.06}$	76.68$_{\pm 0.90}$	72.10$_{\pm 1.62}$	−0.11$_{\pm 0.01}$	**0.19**$_{\pm 0.04}$	1.20$_{\pm 0.09}$	**2.03**$_{\pm 0.06}$
	Greedy	0.83$_{\pm 0.07}$	1.34$_{\pm 0.05}$	77.24$_{\pm 1.59}$	71.30$_{\pm 1.63}$	−0.14$_{\pm 0.01}$	0.18$_{\pm 0.03}$	1.03$_{\pm 0.11}$	2.14$_{\pm 0.05}$
	MBR	**0.82**$_{\pm 0.07}$	**1.33**$_{\pm 0.04}$	**77.92**$_{\pm 1.79}$	**73.00**$_{\pm 1.70}$	−0.18$_{\pm 0.01}$	0.16$_{\pm 0.04}$	**0.87**$_{\pm 0.08}$	2.06$_{\pm 0.05}$
	Ours	**0.76**$_{\pm 0.04}$	**0.68**$_{\pm 0.05}$	**83.07**$_{\pm 0.93}$	**74.10**$_{\pm 2.75}$	**-0.00**$_{\pm 0.00}$	**0.21**$_{\pm 0.03}$	**0.72**$_{\pm 0.05}$	**1.27**$_{\pm 0.06}$
Qwen3-30B-A3B	Ancestral	0.80$_{\pm 0.06}$	1.19$_{\pm 0.12}$	77.91$_{\pm 3.26}$	74.49$_{\pm 3.72}$	**-0.01**$_{\pm 0.01}$	**0.47**$_{\pm 0.04}$	0.64$_{\pm 0.04}$	2.16$_{\pm 0.09}$
	Beam	**0.78**$_{\pm 0.06}$	1.19$_{\pm 0.10}$	**80.27**$_{\pm 2.46}$	74.19$_{\pm 3.56}$	−0.03$_{\pm 0.03}$	**0.47**$_{\pm 0.05}$	0.62$_{\pm 0.03}$	2.16$_{\pm 0.08}$
	Contrastive	0.81$_{\pm 0.07}$	**1.17**$_{\pm 0.10}$	80.27$_{\pm 2.64}$	74.39$_{\pm 2.21}$	−0.03$_{\pm 0.02}$	0.46$_{\pm 0.04}$	0.74$_{\pm 0.08}$	2.13$_{\pm 0.07}$
	Greedy	0.79$_{\pm 0.07}$	1.19$_{\pm 0.11}$	77.02$_{\pm 2.33}$	**74.80**$_{\pm 1.61}$	−0.04$_{\pm 0.02}$	0.46$_{\pm 0.05}$	0.64$_{\pm 0.06}$	2.15$_{\pm 0.12}$
	MBR	0.78$_{\pm 0.07}$	1.18$_{\pm 0.12}$	78.48$_{\pm 1.92}$	74.59$_{\pm 3.52}$	−0.07$_{\pm 0.02}$	0.46$_{\pm 0.05}$	**0.54**$_{\pm 0.03}$	**2.11**$_{\pm 0.12}$
	Ours	**0.72**$_{\pm 0.05}$	**0.67**$_{\pm 0.05}$	**88.34**$_{\pm 2.11}$	**78.63**$_{\pm 2.50}$	**0.03**$_{\pm 0.02}$	0.46$_{\pm 0.04}$	**0.49**$_{\pm 0.03}$	**1.21**$_{\pm 0.04}$

4. Experiments

We evaluate our framework along two axes: (i) We compare MBR decoding on the approximately task-calibrated distribution $f_\phi(\hat{p}(\mathrm{X}))$ to other decoding strategies, demonstrating that our strategy "calibrate first, then act optimally", consistently improves performance. (ii) We use Task Calibration Error (TCE) to assess miscalibration in LLMs for specific tasks and verify that, unlike task-agnostic metrics like ECE, TCE can predict the performance gains that can be realized from calibration.

Tasks and Datasets. We consider different latent structures $\mathcal{L}$ and task-specific losses d_T (see Table 6). (i) **Ordinal Regression**: We evaluate numerical LLM-as-a-judge tasks as ordinal prediction problems. Responses represent elements from a discrete set of reals $\mathcal{L} \cong \{0, 1, \dots, C\}$. We evaluate on Helpsteer (criterion "correctness") (Wang et al., 2023; Dong et al., 2023) and STSB (Cer et al., 2017), using the L_1-loss as d_T. (ii) **Classification**: We also consider single-label classification over a discrete set of C answers with the Exact-Match loss. Here, we use When2Call (Ross et al., 2025), where the LLM classifies if it should call a tool, request more information, or abstain. We also evaluate on the multiple-choice QA benchmark MMLU (Hendrycks et al., 2021). (iii) **Answer-Or-Abstain**: Furthermore, we consider "Answer-Or-Abstain" tasks on TriviaQA (Joshi et al., 2017) and SimpleQA-Verified (Haas et al., 2026). Here, the latent space is binary $\mathcal{L} \cong \{A, \perp\}$, and we use the asymmetric BAS loss of Wu et al. (2026) which depends a user-specified risk tolerance t (see Section C.1). (iv) **Open Set-Prediction**: Moreover, we evaluate Multi-Answer QA using MAQA (Yang et al., 2025). Here, the latent structure is the space of binary vectors $\mathcal{L} \cong \{0, 1\}^{\leq C}$. Each entry indicates if a candidate answer should be included in the response. The support set of possible answers varies across questions, so we must decompose this task and the associated Hamming loss into binary decisions of whether to include an answer. Consequently, we task calibrate the LLM's Bernoulli beliefs over if answers should be included in the response. (v) **Vector-Valued Ordinal Regression**: Lastly, we study vector-valued predictions in $\mathcal{L} \cong \{0, \dots, C\}^2$ and again use the Helpsteer dataset (Wang et al., 2023; Dong et al., 2023). Now, the LLM predicts two criteria of a text simultaneously ("correctness", "helpfulness") and the loss is L_1 distance.

Models. We evaluate the instruction-tuned models Gemma 3 4B, Gemma 3 12B (Gemma Team, 2025), Qwen 3 4B, and Qwen 3 30B A3B (Qwen Team, 2025). All models are run with default decoding parameters (Section B). We sample $M = 20$ times per prompt (Section F) to approximate the push-forward distribution $\hat{p}(\mathrm{X})$.

Calibration. Using Proposition E.1, we approximate the optimal calibration map $f^*_{g_T}$ by optimizing the NLL over a family of functions $\mathcal{F}$. We use Dirichlet calibration (Kull et al., 2019), a generalization of Platt scaling (Platt et al., 1999), which fits a linear model to the logarithm of the predicted probabilities $f_{\boldsymbol{W},\boldsymbol{b}}(\hat{p}(\mathrm{X})) = \mathrm{softmax}\left(\boldsymbol{W} \log \hat{p}(\mathrm{X}) + \boldsymbol{b}\right)$ with $\boldsymbol{W} \in \mathbb{R}^{C \times C}, \boldsymbol{b} \in \mathbb{R}^C$. An overview of MBR decoding using task calibration is given in Algorithm 2. We train and evaluate calibrators

using 5-fold cross-validation, and report the mean performance and standard deviations. We compute the TCE on the whole dataset according to Proposition 3.5, by approximating $f^*_{g_T}$ using ground-truth data and discretizing the simplex into 4^{C-1} equal-volume bins (see Section B.2). This is a multidimensional extension of binning-based ECE estimation. In Section A.3.1, we additionally ablate using a Dirichlet KDE (Popordanoska et al., 2022).

4.1. Recalibration Improves Decoding

In Table 1, we find that across all tasks and datasets, our strategy consistently achieves the best performance. Theorem 3.4 only guarantees optimality among decoding that relies solely on on $\hat{p}(\mathrm{X})$, a criterion that only applies to the baseline of MBR decoding on the uncalibrated distribution. However, both Tomov et al. (2026) and our experiments validate that this uncalibrated MBR baseline performs competitively among other decoding strategies. Therefore, improving on uncalibrated MBR decoding also improves upon more general decoding strategies as well.

Ablations. We also ablate different methods to calibrate $\hat{p}(\mathrm{X})$ before MBR decoding: (i) Scalar temperature scaling (Guo et al., 2017) to the logarithm of the predictor $\log \hat{p}(\mathrm{X})$, and (ii) the decision calibration algorithm of Zhao et al. (2021). (iii) We also compare against directly learning to predict the correct latent action $\ell \in \mathcal{L}$ from the uncalibrated distribution using a policy $\pi_\phi : \Delta^{C-1} \mapsto \mathcal{L}$. This policy is fitted with a classification loss like NLL on the calibration data, similar to the calibration objective used to learn f_ϕ (Proposition E.1). Consequently, this baseline is realized exactly by re-interpreting the calibration map f_ϕ as a classifier $\pi_\phi(\hat{p}(\mathrm{X})) = \arg\max_c f_\phi(\hat{p}(\mathrm{X}))_c$ instead of outputting the MBR response. The policy baseline π_ϕ is realized by outputting the ℓ assigned the maximum probability (MP) by f_ϕ. Note that by Theorem 3.4, this baseline can not outperform MBR on the task calibrated latent distribution. However, comparing to this baseline shows how much performance gain is achieved by fully task calibrating the latent distribution $\hat{p}(\mathrm{X})$, and how much comes from learning the correct predictions directly from a calibration set using a NLL loss. For some task losses d_T (Exact Match & Hamming), the MP action of f_ϕ coincides with the MBR action and, here, our framework can be interpreted as learning a classifier on the latent belief $\hat{p}(\mathrm{X})$. For all other losses, performance improvements over this baseline π_ϕ stem directly from fully calibrating the latent beliefs.

In Table 2 (full results in Table 3), we observe that: (a) Dirichlet calibration is consistently the best strategy and MBR decoding on it most faithfully recovers the desired optimal action on $f^*_{g_T}$. (b) For losses where the MP and MBR action do not coincide, MBR decoding outperforms the classifier on the latents (MP action). This confirms that there is substantial gain not only in correcting the LLM's latent beliefs by fitting a classifier π_ϕ but instead task calibrating the LLM's belief distribution.

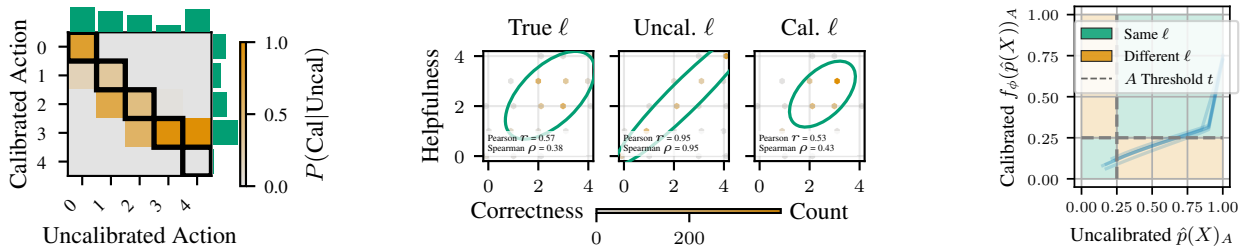
Table 2. Ablation on calibration methods for Qwen3-30B-A3B, including the MAP action on the Dirichlet calibrated predictor. Dirichlet calibration and MBR perform the best in most tasks.

$\mathcal{L}$(Metric)	$\{0,\dots,C\}$ (L1 ↓)		$\{A,\perp\}$ (BAS ↑)		$\{0,\dots,C\}^2$ (L1 ↓)
	Helpsteer	STSB	SimpleQA	TriviaQA	Helpsteer
Uncalibrated	$0.78_{\pm 0.07}$	$1.18_{\pm 0.12}$	$-0.07_{\pm 0.02}$	$0.46_{\pm 0.05}$	$2.11_{\pm 0.12}$
Temperature	$0.77_{\pm 0.07}$	$1.18_{\pm 0.12}$	$-0.10_{\pm 0.03}$	$0.46_{\pm 0.05}$	$2.04_{\pm 0.09}$
Decision	$\mathbf{0.77}_{\pm 0.04}$	$0.70_{\pm 0.08}$	$\mathbf{0.03}_{\pm 0.02}$	$\mathbf{0.47}_{\pm 0.05}$	$\mathbf{1.23}_{\pm 0.02}$
Policy π_ϕ	$0.78_{\pm 0.07}$	$\mathbf{0.70}_{\pm 0.06}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.47}_{\pm 0.04}$	$1.37_{\pm 0.04}$
Dirichlet	$\mathbf{0.72}_{\pm 0.05}$	$\mathbf{0.67}_{\pm 0.05}$	$\mathbf{0.03}_{\pm 0.02}$	$0.46_{\pm 0.04}$	$\mathbf{1.21}_{\pm 0.04}$

What the Calibration Maps Learn. We now study how the calibration map $f_\phi(\hat{p}(\mathrm{X}))$ transforms the original latent probabilities $\hat{p}(\mathrm{X})$. For binary action spaces like the Answer-or-Abstain latent structure $\mathcal{L} \cong \{A, \perp\}$, the map can be fully described by how it maps the probability of one of the actions, for example A ("answer"). In Figure 2c, we observe that for this action space, the learned map f_ϕ consistently lowers the predicted probability of the action A ("answer") to mitigate overconfidence in the uncalibrated LLM's belief distribution. Effectively, the recalibration map will only output probabilities this action $f_\phi(\hat{p}(\mathrm{X}))_A$ above the threshold in the BAS-metric ($t = 0.25$) if the uncalibrated distribution assigns a significantly higher probability ($\hat{p}(\mathrm{X})_A > 0.7$).

For multi-criterion ordinal regression (Helpsteer) ($\mathcal{L} \cong \{0,\dots,4\}^2$), i.e. rating the correctness of a model response, we study how MBR responses change between the uncalibrated and calibrated beliefs in Figure 2a. We find a pattern akin to an "S-Curve": for low ratings (0–2), the model tends to be underconfident and calibration shifts MBR responses toward higher scores. In contrast, high ratings (3–4) are either left unchanged or shifted downward. Interestingly, the learned calibration map effectively no longer yields the latent response $\ell = 4$. However, by Lemma 3.3, among all queries for which the uncalibrated LLM's MBR response is $\ell = 4$, i.e. $P_4 = \{\hat{p}(\mathrm{X}) \ : \delta^{\mathrm{MBR}}(\hat{p}(\mathrm{X})) = 4\}$, calibration reduces the average loss. In our experiments, predicting $\ell = 3$ whenever the model's uncalibrated belief yields an MBR action $\ell = 4$ has better performance on average. Also, no other uncalibrated MBR action is shifted to a calibrated MBR response of $\ell = 4$. This implies that there is no signal in the LLM's uncalibrated beliefs $\hat{p}(\mathrm{X})$ from which it is optimal to predict $\ell = 4$. This is not a flaw of task calibration but reveals a severe shortcoming in the LLM: It is unable to reliably predict $\ell = 4$. Hence, the optimal calibration strategy in terms of the expected loss is to avoid predicting $\ell = 4$ altogether.

Similarly, we study how task calibration can help to mitigate systematic biases (Gallegos et al., 2024; Shi et al., 2025) in an LLM's distribution. In particular, we consider the multi-criterion ordinal regression task on Helpsteer, where



(a) ■ Absolute Counts & ■ $\mathbb{P}(\text{Cal}|\text{Uncal})$. Change from uncalibrated to calibrated action for Helpsteer. Low rating are upshifted while higher rating remain unchanged or are decreased.

(b) Answer correlations for vector-valued Helpsteer predictions ("helpfulness" and "correctness"). Ellipses indicate the 2-s.d. covariance region. The uncalibrated model shows heavy correlation between the criteria, whereas task calibration recovers the weaker correlation structure observed in the ground truth.

(c) Learned calibration map f_ϕ for $\mathcal{L} = \{A, \perp\}$ for the probability of action $\ell = A$. While the uncalibrated LLM predicts $\ell = A$("Answer") if $\hat{p}(\mathrm{X})_A \geq t$, the calibrated one is more conservative.

the LLM rates a model's response on both helpfulness and correctness simultaneously. In Figure 2b, we find that the uncalibrated $\hat{p}(\mathrm{X})$ heavily correlates ratings for the two criteria, meaning the LLMs is very likely to assign identical values to both. However, in the true data, the correlation is significantly weaker. The task calibration map $f_\phi(\hat{p}(\mathrm{X}))$ more accurately recovers the true correlation structure and therefore improves the LLM's reliability by correcting this correlation bias using the calibration data.

4.2. Task-Specific Calibration Error

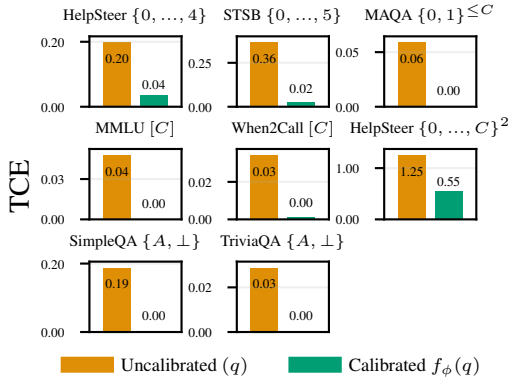
We evaluate the quality of our recalibration map f_ϕ through the Task Calibration Error (TCE). We compare the TCE of the the original model predictions $\hat{p}(\mathrm{X})$ with their recalibrated counterparts $f_\phi(\hat{p}(\mathrm{X}))$. Figure 3a shows that the LLMs are often substantially task miscalibrated, which indicates possible performance improvements through task calibration. Importantly, Dirichlet recalibration drastically reduces TCE – often to zero –, demonstrating that the approximately recalibrated predictor $\delta^{\mathrm{MBR}}(f_\phi(\hat{p}(\mathrm{X})))$ recovers most of the theoretically attainable improvement. It also verifies that a simple affine (Dirichlet) recalibration suffices to find MBR actions that are optimal according to Theorem 3.4, consistent with the strong empirical decoding performance seen in Table 1.

We also verify the practical usefulness of TCE by investigating how well it predicts performance gains achieved by recalibration and compare it to the Expected Calibration Error (ECE) (DeGroot & Fienberg, 1983; Guo et al., 2017) computed from the LLM's latent distribution $\hat{p}(\mathrm{X})$. As shown in Figure 3b, in contrast to ECE, TCE exhibits a strong correlation with the realized loss reduction obtained through recalibration. Notably, as TCE is defined directly in terms of the task-specific loss, it permits a task-dependent interpretation, and the concrete values are meaningful. In contrast, ECE quantifies an average difference in confidence that cannot be directly tied to the task's loss. We defer full results to Section A.3.

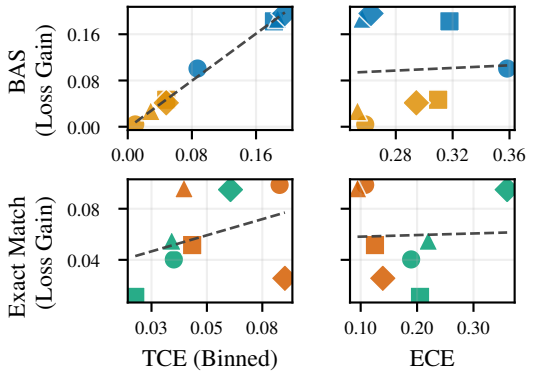
5. Related Work

Calibration. Classical calibration notions such as confidence, class-wise, and distributional calibration typically concern classification (Kull & Flach, 2015; Guo et al., 2017; Kull et al., 2019). Similar to decision calibration (Zhao et al., 2021), our work is motivated by a given decision-making problem which, in our case, is implied by the task at hand. Our task calibration differs from decision calibration as the latter ensures optimality of MBR decoding only among other MBR decision rules. Our proposed criterion of distributionally calibrating the LLM's latent beliefs makes MBR decoding optimal among *all* decision rules derived from the latent beliefs. Furthermore, our Task Calibration Error (TCE) can be seen as applying the decision-specific calibration error of Hu & Wu (2024) to an LLM's latent beliefs. In our work, we connect concepts from the existing literature on calibration for classification with LLM's by interpreting their output distribution in a task-dependent latent space, and improve decision making such as decoding.

Calibration in LLMs. Recent work has shown that LLMs are often miscalibrated (Kadavath et al., 2022; Plaut et al., 2025). Consequently, methods emerged that recalibrate token probabilities or verbalized confidence scores (Xie et al., 2024; Li et al., 2025b). Some recent work also moves beyond token-level probabilities: Nakkiran et al. (2025) show that language models can exhibit confidence calibration in semantic answer spaces. This semantic answer grouping can be seen as a special case of our proposed *task* confidence calibration by instantiating the latent map g_T with a semantic answer grouping. For long-form generation, some work (Huang et al., 2024; Zhang et al., 2025; Band et al., 2024) examines confidence calibration of atomic statements extracted from responses. From the perspective of our framework, this corresponds to confidence task calibration using a latent mapping g_T that extracts these atomic statements. Overall, our notion of distributional task calibration is applicable to any latent structure and therefore generalizes many existing notions of calibration in LLMs that can be recov-



(a) Task Calibration Error (TCE) for uncalibrated q and Dirichlet-calibrated $f_\phi(q)$ model on Gemma 3-12B. Across all tasks, the model is severely task miscalibrated, which is mitigated by our approximate recalibration method.



(b) Task Calibration Error (TCE) correlates with downstream gains, whereas ECE does not. Datasets: ● SimpleQA, ● TriviaQA, ● When2Call, ● MMLU. Models: ● Qwen3-30B-A3B, ■ Qwen3-4B, ▲ Gemma-3-12B, ◆ Gemma-3-4B

ered for a specific g_T. Moreover, only few works connect calibration to decision-making in LLMs: Wu et al. (2026) study Answer-versus-Abstain tasks under asymmetric costs, while Band et al. (2024) use calibration to improve classification. Our framework can be applied to any task-specific loss function and enables optimal decisions through MBR decoding in any given task.

Decoding in LLMs. There are numerous approaches to obtaining responses from the predictive distribution of an LLM (Shi et al., 2024). Well-established paradigms include deterministic strategies like Greedy token selection, Beam Search (Freitag & Al-Onaizan, 2017), or Contrastive Decoding (Li et al., 2023), as well as sampling-based non-deterministic approaches like Ancestral sampling or top-k sampling (Fan et al., 2018). MBR decoding originates in Machine Translation where risk is typically a learned similarity score (Eikema & Aziz, 2022; Müller & Sennrich, 2021). Lukasik et al. (2024) utilize MBR for LLMs in regression problems and Tomov et al. (2026) generalize this approach to arbitrary latent structures $\mathcal{L}$. They empirically observe the latent structure to be beneifical for MBR decoding. We build on this framework of interpreting the LLM's output distribution in a task-aware latent space $\mathcal{L}$ and both theoretically and empirically show that calibrating this latent distribution enables MBR as an optimal task-specific decoding strategy with strong empirical performance.

6. Discussion

Limitations. Our optimality guarantee holds for decoding strategies that are based solely on the LLM's latent distribution $\hat{p}(\mathrm{X})$. If additional information is available, such as internal representations or alternative queries, there may be decoding strategies that perform better. Nonetheless, we empirically show that our framework outperforms decoding strategies beyond our formal framework in practice. Moreover, the performance improvement is given *in expectation*: task calibration can improve *average* downstream decision quality, but does not imply per-instance performance gain. Furthermore, task calibration requires a discrete latent support that is shared across samples. While this is natural for many tasks (e.g., ratings or fixed candidate sets), it may not hold in more open-ended settings. However, our work is useful toward this direction as well, as open-ended tasks can often be meaningfully reduced to a simpler structure. For example, we demonstrate that *Answer-versus-Abstain* settings can be effectively task calibrated. Further, we show how more complex problems, such as set prediction over open-ended support, can be decomposed into simpler decision-making problems that are again tractable to task calibrate. Lastly, like the work of Tomov et al. (2026) we build upon, our framework leverages semantics induced by a given task. Therefore, task calibration improves reliability and decoding performance for a given problem, but not for all tasks simultaneously with a single recalibration map. Our method should be seen as a widely applicable tool to enhance LLMs when they are deployed in a downstream application that admits a task-dependent latent structure.

Future Work Our results highlight the potential of the proposed *calibrate first, then act optimally* strategy to systematically improve downstream performance. The framework is generic and applies to any setting that admits a discrete latent representation with shared support across outputs, or that can be decomposed into such a structure. One particularly promising direction is LLM-as-a-judge systems (Gu et al., 2025), which frequently rely on rubric-based evaluations and thus naturally fit into our framework. Because judges often produce noisy or biased scores, task calibration could substantially improve their reliability (Lee et al., 2026). More broadly, we see potential for task calibration to facilitate more reliable model decisions across a variety of tasks and application domains.

Impact Statement

In this work, we examine how calibration can improve the generation performance of Large Language Models. While any research may be misused, our primary goal is to improve the reliability of these models to support their safe deployment in critical domains. We believe the benefits will outweigh the potential risks.

Acknowledgments

We want to thank Franz Rieger, Leon Hetzel, and Arthur Kosmala for reviewing the paper. The research presented has been performed in the frame of the RADELN project funded by TUM Georg Nemetschek Institute Artificial Intelligence for the Built World (GNI). It is further supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy with funds from the Hightech Agenda Bayern. Additionally, this paper has been supported by the DAAD programme Konrad Zuse Schools of Excellence in Artificial Intelligence, sponsored by the German Federal Ministry of Education and Research. Lastly, Rajeev Verma is a part of UvA-Bosch Delta Lab at the University of Amsterdam funded by the Bosch Center for Artificial Intelligence.

References

Band, N., Li, X., Ma, T., and Hashimoto, T. Linguistic calibration of long-form generations, 2024. URL https://arxiv.org/abs/2404.00474.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pp. 1–14, 2017.

Chhikara, P. Mind the confidence gap: Overconfidence, calibration, and distractor effects in large language models. *arXiv preprint arXiv:2502.11028*, 2025.

DeGroot, M. H. and Fienberg, S. E. The comparison and evaluation of forecasters. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 32(1-2):12–22, 1983.

Dong, Y., Wang, Z., Sreedhar, M. N., Wu, X., and Kuchaiev, O. Steerlm: Attribute conditioned sft as an (user-steerable) alternative to rlhf, 2023.

Dwork, C., Kim, M. P., Reingold, O., Rothblum, G. N., and Yona, G. Outcome indistinguishability. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, 2021.

Eikema, B. and Aziz, W. Sampling-based approximations to minimum Bayes risk decoding for neural machine translation. In Goldberg, Y., Kozareva, Z., and Zhang, Y. (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 10978–10993, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.754. URL https://aclanthology.org/2022.emnlp-main.754/.

Fan, A., Lewis, M., and Dauphin, Y. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 889–898, 2018.

Freitag, M. and Al-Onaizan, Y. Beam search strategies for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pp. 56–60, 2017.

Gallegos, I. O., Rossi, R. A., Barrow, J., Tanjim, M. M., Kim, S., Dernoncourt, F., Yu, T., Zhang, R., and Ahmed, N. K. Bias and fairness in large language models: A survey. *Computational linguistics*, 50(3):1097–1179, 2024.

Gemma Team. Gemma 3 technical report, 2025. URL https://arxiv.org/abs/2503.19786.

Grünwald, P. Safe probability. *Journal of Statistical Planning and Inference*, 2018.

Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K., Wang, Y., Gao, W., Ni, L., and Guo, J. A survey on llm-as-a-judge, 2025. URL https://arxiv.org/abs/2411.15594.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International conference on machine learning*, pp. 1321–1330. PMLR, 2017.

Haas, L., Yona, G., D'Antonio, G., Goldshtein, S., and Das, D. Simpleqa verified: A reliable factuality benchmark to measure parametric knowledge, 2026. URL https://arxiv.org/abs/2509.07968.

Hashimoto, W., Kamigaito, H., and Watanabe, T. Decoding uncertainty: The impact of decoding strategies for uncertainty estimation in large language models. In *Findings of the Association for Computational Linguistics*, 2025.

He, P., Liu, X., Gao, J., and Chen, W. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Hu, L. and Wu, Y. Calibration error for decision making, 2024. URL https://arxiv.org/abs/2404.13503.

Huang, Y., Liu, Y., Thirukovalluru, R., Cohan, A., and Dhingra, B. Calibrating long-form generations from large language models, 2024. URL https://arxiv.org/abs/2402.06544.

Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, 2017. URL https://arxiv.org/abs/1705.03551.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., Johnston, S., El-Showk, S., Jones, A., Elhage, N., Hume, T., Chen, A., Bai, Y., Bowman, S., Fort, S., Ganguli, D., Hernandez, D., Jacobson, J., Kernion, J., Kravec, S., Lovitt, L., Ndousse, K., Olsson, C., Ringer, S., Amodei, D., Brown, T., Clark, J., Joseph, N., Mann, B., McCandlish, S., Olah, C., and Kaplan, J. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, 2023. URL https://arxiv.org/abs/2302.09664.

Kull, M. and Flach, P. Novel decompositions of proper scoring rules for classification: Score adjustment as precursor to calibration. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 68–85. Springer, 2015.

Kull, M., Perello Nieto, M., Kängsepp, M., Silva Filho, T., Song, H., and Flach, P. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration. *Advances in neural information processing systems*, 32, 2019.

Lee, C., Zeng, T., Jeong, J., yong Sohn, J., and Lee, K. How to correctly report llm-as-a-judge evaluations, 2026. URL https://arxiv.org/abs/2511.21140.

Li, C., Zhou, H., Glavaš, G., Korhonen, A., and Vulić, I. Large language models are miscalibrated in-context learners. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 11575–11596, 2025a.

Li, X. L., Holtzman, A., Fried, D., Liang, P., Eisner, J., Hashimoto, T. B., Zettlemoyer, L., and Lewis, M. Contrastive decoding: Open-ended text generation as optimization. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, 2023.

Li, Y., Xiong, M., Wu, J., and Hooi, B. Conftuner: Training large language models to express their confidence verbally, 2025b. URL https://arxiv.org/abs/2508.18847.

Lukasik, M., Narasimhan, H., Menon, A. K., Yu, F., and Kumar, S. Regression-aware inference with llms, 2024. URL https://arxiv.org/abs/2403.04182.

Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., and Lucic, M. Revisiting the calibration of modern neural networks. *Advances in neural information processing systems*, 34:15682–15694, 2021.

Müller, M. and Sennrich, R. Understanding the properties of minimum Bayes risk decoding in neural machine translation. In Zong, C., Xia, F., Li, W., and Navigli, R. (eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 259–272, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.22. URL https://aclanthology.org/2021.acl-long.22/.

Nakkiran, P., Bradley, A., Goliński, A., Ndiaye, E., Kirchhof, M., and Williamson, S. Trained on tokens, calibrated on concepts: The emergence of semantic calibration in llms, 2025. URL https://arxiv.org/abs/2511.04869.

Nixon, J., Dusenberry, M. W., Zhang, L., Jerfel, G., and Tran, D. Measuring calibration in deep learning. In *CVPR workshops*, volume 2, 2019.

Noarov, G. and Roth, A. Calibration for decision making: A principled approach to trustworthy ml, 2024. URL https://www.let-all.com/blog/2024/03/13/calibration-for-decision-making-a-principled-appr

Pavlovic, M. Understanding model calibration - a gentle introduction and visual exploration of calibration and the expected calibration error (ece). In *ICLR Blogposts 2025*, 2025. URL https://iclr-blogposts.github.io/2025/blog/calibration/. https://iclr-blogposts.github.io/2025/blog/calibration/.

Platt, J. et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

Plaut, B., Khanh, N. X., and Trinh, T. Probabilities of chat llms are miscalibrated but still predict correctness on multiple-choice q&a, 2025. URL https://arxiv.org/abs/2402.13213.

Popordanoska, T., Sayer, R., and Blaschko, M. B. A consistent and differentiable lp canonical calibration error estimator, 2022. URL https://arxiv.org/abs/2210.07810.

Qwen Team. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Ross, H., Mahabaleshwarkar, A. S., and Suhara, Y. When2Call: When (not) to call tools. In Chiruzzo, L., Ritter, A., and Wang, L. (eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 3391–3409, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics. ISBN 979-8-89176-189-6. doi: 10.18653/v1/2025.naacl-long.174. URL https://aclanthology.org/2025.naacl-long.174/.

Shi, C., Yang, H., Cai, D., Zhang, Z., Wang, Y., Yang, Y., and Lam, W. A thorough examination of decoding methods in the era of llms. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 8601–8629, 2024.

Shi, L., Ma, C., Liang, W., Diao, X., Ma, W., and Vosoughi, S. Judging the judges: A systematic study of position bias in LLM-as-a-judge. In Inui, K., Sakti, S., Wang, H., Wong, D. F., Bhattacharyya, P., Banerjee, B., Ekbal, A., Chakraborty, T., and Singh, D. P. (eds.), *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, pp. 292–314, Mumbai, India, December 2025. The Asian Federation of Natural Language Processing and The Association for Computational Linguistics. ISBN 979-8-89176-298-5. doi: 10.18653/v1/2025.ijcnlp-long.18. URL https://aclanthology.org/2025.ijcnlp-long.18/.

Therneau, T. M. and Atkinson, E. Concordance. Vignette of the `survival` R package, December 2024. URL https://cran.r-project.org/web/packages/survival/vignettes/concordance.pdf. Accessed: 2025-08-29.

Tomov, T., Fuchsgruber, D., and Günnemann, S. Task-awareness improves llm generations and uncertainty, 2026. URL https://arxiv.org/abs/2601.21500.

Vaicenavicius, J., Widmann, D., Andersson, C., Lindsten, F., Roll, J., and Schön, T. Evaluating model calibration in classification. In *International conference on artificial intelligence and statistics*, 2019.

Wang, Z., Dong, Y., Zeng, J., Adams, V., Sreedhar, M. N., Egert, D., Delalleau, O., Scowcroft, J. P., Kant, N., Swope, A., and Kuchaiev, O. Helpsteer: Multi-attribute helpfulness dataset for steerlm, 2023.

Wu, S., Gustafsson, F. K., Phillips, E., Gao, B., Thakur, A., and Clifton, D. A. Bas: A decision-theoretic approach to evaluating large language model confidence, 2026. URL https://arxiv.org/abs/2604.03216.

Xia, G. What (and what not) are calibrated probabilities actually useful for? In *ICLR Blogposts 2026*, 2026. URL https://iclr-blogposts.github.io/2026/blog/2026/useful-calibrated-uncertainties/. https://iclr-blogposts.github.io/2026/blog/2026/useful-calibrated-uncertainties/.

Xie, J., Chen, A. S., Lee, Y., Mitchell, E., and Finn, C. Calibrating language models with adaptive temperature scaling. In Al-Onaizan, Y., Bansal, M., and Chen, Y.-N. (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 18128–18138, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.1007. URL https://aclanthology.org/2024.emnlp-main.1007/.

Yang, Y., Yoo, H., and Lee, H. Maqa: Evaluating uncertainty quantification in llms regarding data uncertainty, 2025. URL https://arxiv.org/abs/2408.06816.

Zhang, C., Yang, R., Zhang, Z., Huang, X., Yang, S., Yu, D., and Collier, N. Atomic calibration of llms in long-form generations, 2025. URL https://arxiv.org/abs/2410.13246.

Zhao, S., Kim, M. P., Sahoo, R., Ma, T., and Ermon, S. Calibrating predictions to decisions: A novel approach to multi-class calibration, 2021. URL https://arxiv.org/abs/2107.05719.

A. Additional Experiments

A.1. Ablation on Calibration Methods

Table 3. Ablation on calibration methods on all models, including the MAP action on the Dirichlet calibrated predictor. Dirichlet calibration and MBR perform the best in most tasks. For Hamming and Exact Match losses, the MBR action and the MP action (classifier π_ϕ) are the same.

Latent Structure	$\mathcal{L}$(Metric)	$\{0,\dots,C\}$ (L1 ↓)		$[C]$ (Acc. ↑)		$\{A,\perp\}$ (BAS ↑)		$\{0,1\}^{\leq C}$ (Hamming ↓)	$\{0,\dots,C\}^2$ (L1 ↓)
		Helpsteer	STSB	MMLU	When2Call	SimpleQA	TriviaQA	MAQA	Helpsteer
Gemma-3-4B	Uncalibrated	$0.82_{\pm 0.08}$	$1.07_{\pm 0.08}$	$64.56_{\pm 2.08}$	$51.14_{\pm 2.78}$	$\mathbf{-0.20}_{\pm 0.02}$	$0.18_{\pm 0.05}$	$0.74_{\pm 0.04}$	$2.03_{\pm 0.12}$
	Temperature	$0.80_{\pm 0.10}$	$1.07_{\pm 0.08}$	$64.56_{\pm 2.08}$	$51.14_{\pm 2.78}$	$-0.25_{\pm 0.02}$	$0.17_{\pm 0.05}$	$0.74_{\pm 0.04}$	$1.98_{\pm 0.14}$
	Decision	$\mathbf{0.70}_{\pm 0.10}$	$0.84_{\pm 0.04}$	$\mathbf{67.36}_{\pm 1.17}$	$\mathbf{56.10}_{\pm 3.40}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.24}_{\pm 0.05}$	$\mathbf{0.66}_{\pm 0.04}$	$\mathbf{1.24}_{\pm 0.07}$
	Policy π_ϕ	$0.82_{\pm 0.11}$	$\mathbf{0.82}_{\pm 0.07}$	$\mathbf{67.11}_{\pm 3.95}$	$\mathbf{60.63}_{\pm 3.57}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.23}_{\pm 0.03}$	$\mathbf{0.62}_{\pm 0.03}$	$1.42_{\pm 0.04}$
	Dirichlet	$\mathbf{0.73}_{\pm 0.08}$	$\mathbf{0.80}_{\pm 0.05}$	$\mathbf{67.11}_{\pm 3.95}$	$\mathbf{60.63}_{\pm 3.57}$	$\mathbf{-0.00}_{\pm 0.00}$	$0.22_{\pm 0.05}$	$\mathbf{0.62}_{\pm 0.03}$	$\mathbf{1.25}_{\pm 0.04}$
Gemma-3-12B	Uncalibrated	$0.90_{\pm 0.05}$	$1.01_{\pm 0.03}$	$80.11_{\pm 2.84}$	$68.87_{\pm 2.09}$	$\mathbf{-0.19}_{\pm 0.03}$	$0.44_{\pm 0.03}$	$0.60_{\pm 0.06}$	$1.89_{\pm 0.07}$
	Temperature	$0.88_{\pm 0.05}$	$1.01_{\pm 0.02}$	$80.11_{\pm 2.84}$	$68.87_{\pm 2.09}$	$-0.25_{\pm 0.03}$	$0.44_{\pm 0.03}$	$0.60_{\pm 0.06}$	$1.79_{\pm 0.05}$
	Decision	$\mathbf{0.83}_{\pm 0.04}$	$0.79_{\pm 0.05}$	$\mathbf{81.34}_{\pm 2.97}$	$\mathbf{71.77}_{\pm 2.72}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.47}_{\pm 0.03}$	$\mathbf{0.56}_{\pm 0.06}$	$\mathbf{1.25}_{\pm 0.06}$
	Policy π_ϕ	$0.92_{\pm 0.02}$	$\mathbf{0.69}_{\pm 0.05}$	$\mathbf{89.72}_{\pm 1.56}$	$\mathbf{74.38}_{\pm 2.64}$	$\mathbf{-0.00}_{\pm 0.00}$	$0.47_{\pm 0.02}$	$\mathbf{0.51}_{\pm 0.05}$	$1.36_{\pm 0.07}$
	Dirichlet	$\mathbf{0.79}_{\pm 0.07}$	$\mathbf{0.65}_{\pm 0.07}$	$\mathbf{89.72}_{\pm 1.56}$	$\mathbf{74.38}_{\pm 2.64}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.47}_{\pm 0.03}$	$\mathbf{0.51}_{\pm 0.05}$	$\mathbf{1.26}_{\pm 0.04}$
Qwen-3-4B	Uncalibrated	$0.82_{\pm 0.07}$	$1.33_{\pm 0.04}$	$\mathbf{77.92}_{\pm 1.79}$	$\mathbf{73.00}_{\pm 1.70}$	$\mathbf{-0.18}_{\pm 0.01}$	$0.16_{\pm 0.04}$	$0.87_{\pm 0.08}$	$2.06_{\pm 0.05}$
	Temperature	$0.79_{\pm 0.08}$	$1.33_{\pm 0.04}$	$\mathbf{77.92}_{\pm 1.79}$	$\mathbf{73.00}_{\pm 1.70}$	$-0.22_{\pm 0.01}$	$0.15_{\pm 0.04}$	$0.87_{\pm 0.08}$	$1.97_{\pm 0.06}$
	Decision	$\mathbf{0.78}_{\pm 0.06}$	$0.73_{\pm 0.05}$	$77.58_{\pm 2.41}$	$71.80_{\pm 3.85}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.21}_{\pm 0.03}$	$\mathbf{0.75}_{\pm 0.07}$	$\mathbf{1.29}_{\pm 0.04}$
	Policy π_ϕ	$0.84_{\pm 0.06}$	$\mathbf{0.71}_{\pm 0.02}$	$\mathbf{83.07}_{\pm 0.93}$	$\mathbf{74.10}_{\pm 2.75}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.21}_{\pm 0.03}$	$\mathbf{0.72}_{\pm 0.05}$	$1.40_{\pm 0.09}$
	Dirichlet	$\mathbf{0.76}_{\pm 0.04}$	$\mathbf{0.68}_{\pm 0.05}$	$\mathbf{83.07}_{\pm 0.93}$	$\mathbf{74.10}_{\pm 2.75}$	$\mathbf{-0.00}_{\pm 0.00}$	$0.21_{\pm 0.03}$	$\mathbf{0.72}_{\pm 0.05}$	$\mathbf{1.27}_{\pm 0.06}$
Qwen3-30B-A3B	Uncalibrated	$0.78_{\pm 0.07}$	$1.18_{\pm 0.12}$	$78.48_{\pm 1.92}$	$74.59_{\pm 3.52}$	$-0.07_{\pm 0.02}$	$0.46_{\pm 0.05}$	$0.54_{\pm 0.03}$	$2.11_{\pm 0.12}$
	Temperature	$0.77_{\pm 0.07}$	$1.18_{\pm 0.12}$	$78.48_{\pm 1.92}$	$74.59_{\pm 3.52}$	$-0.10_{\pm 0.03}$	$0.46_{\pm 0.05}$	$0.54_{\pm 0.03}$	$2.04_{\pm 0.09}$
	Decision	$\mathbf{0.77}_{\pm 0.04}$	$0.70_{\pm 0.08}$	$\mathbf{81.39}_{\pm 2.42}$	$\mathbf{75.60}_{\pm 2.94}$	$\mathbf{0.03}_{\pm 0.02}$	$\mathbf{0.47}_{\pm 0.05}$	$\mathbf{0.51}_{\pm 0.03}$	$\mathbf{1.23}_{\pm 0.02}$
	Policy π_ϕ	$0.78_{\pm 0.07}$	$\mathbf{0.70}_{\pm 0.06}$	$\mathbf{88.34}_{\pm 2.11}$	$\mathbf{78.63}_{\pm 2.50}$	$\mathbf{-0.00}_{\pm 0.00}$	$\mathbf{0.47}_{\pm 0.04}$	$\mathbf{0.49}_{\pm 0.03}$	$1.37_{\pm 0.04}$
	Dirichlet	$\mathbf{0.72}_{\pm 0.05}$	$\mathbf{0.67}_{\pm 0.05}$	$\mathbf{88.34}_{\pm 2.11}$	$\mathbf{78.63}_{\pm 2.50}$	$\mathbf{0.03}_{\pm 0.02}$	$0.46_{\pm 0.04}$	$\mathbf{0.49}_{\pm 0.03}$	$\mathbf{1.21}_{\pm 0.04}$

In Table 3, we compare our Dirichlet task calibration map to different calibration strategies. We consistently observe it to yield the highest quality results, indicating that the MBR actions on the recalibrated LLM beliefs are optimal.

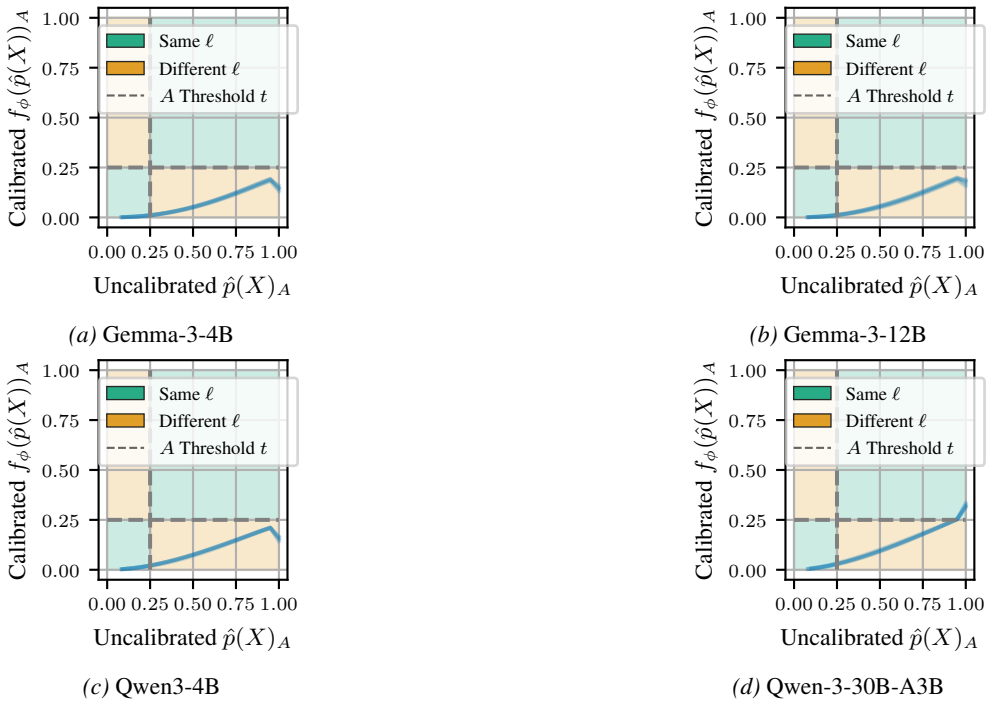
We also outperform directly learning a classifier on the latent belief distribution $\hat{p}(\mathrm{X})$ which is equivalent to selecting the Maximum Probability (MP) action on the Dirichlet calibration map f_ϕ. For Exact Match and Hamming loss, the MP and the MBR responses are identical, and we consequently also achieve the same performance. In this case, our framework can indeed by interpreted as learning a classifier on the latent beliefs. Similarly, temperature scaling does not affect the MP or MBR action and, therefore, it yields the same performance as not calibrating for these losses. For the other loss functions, we indeed find that calibration and MBR decoding consistently outperforms the learned classifier. This shows that the power of our method comes not from utilizing the labels in the calibration set to correct the LLM's predictions but instead from calibrating its latent belief distribution.

A.2. Calibration Maps

Figures 4 and 5 show how the learned recalibration maps act on the binary latent space $\{A, \perp\}$: In Figure 4, we see that for all LLMs but the Qwen3-30B-A3B model, the learned calibration map counteracts the strong overconfidence in answering by collapsing all actions onto abstaining $\perp$ by predicting a probability below the threshold t. While this results in the calibrated LLM always abstaining, Table 1 confirms that this result in lower average loss. One explanation is that the three LLMs for which this behaviour occurs are all poor predictors of the true label and, therefore, the best calibration map in terms of the BAS risk is to always abstain instead of trusting the LLM's predicted distribution. In contrast, for TriviaQA, we, too, observe overconfidence of the LLMs in Figure 5 that is mitigated by our task calibration: Only uncalibrated probabilities that are close to one (i.e. very high confidence) get mapped to values above the answer threshold t.

A.3. Task Calibration Error (TCE)

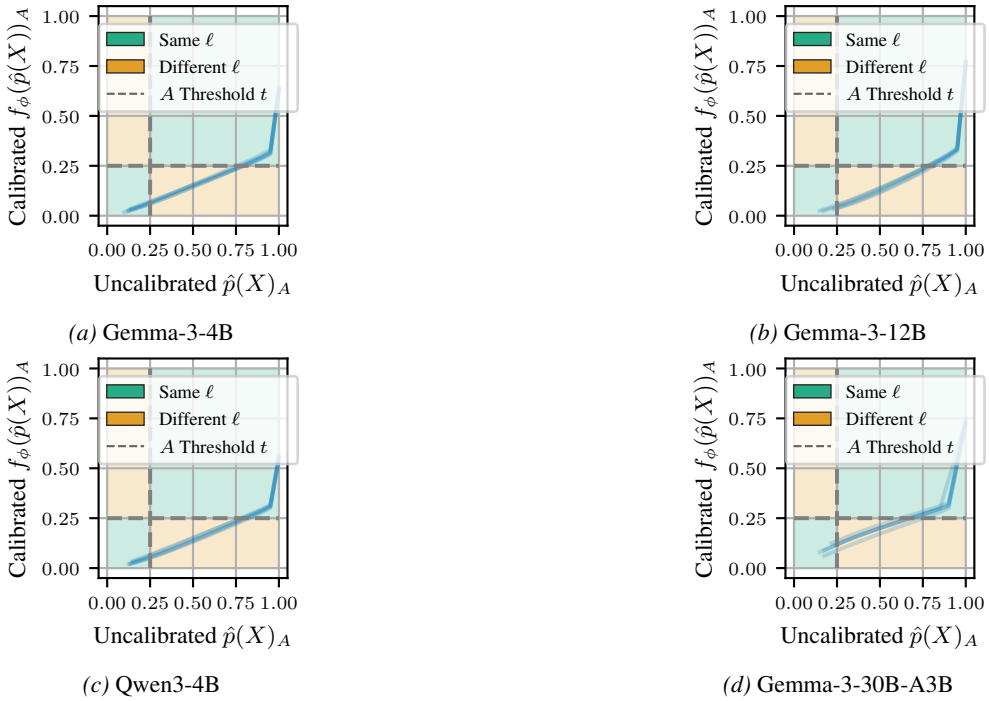
Table 4 reports TCE values estimated with the binned estimator. Across models and tasks, we observe substantial task miscalibration before recalibration. Since TCE is defined in the same units as the task loss, it admits a direct interpretation; for example, for Qwen3-30B-A3B on MMLU, a TCE of 0.083 corresponds to 8.3 percentage points of excess error.



(a) Gemma-3-4B *(b)* Gemma-3-12B

(c) Qwen3-4B *(d)* Qwen-3-30B-A3B

Figure 4. Learned calibration maps $f_\phi(\hat{p}(\mathrm{X})$ on SimpleQA.



(a) Gemma-3-4B *(b)* Gemma-3-12B

(c) Qwen3-4B *(d)* Gemma-3-30B-A3B

Figure 5. Learned calibration maps $f_\phi(\hat{p}(\mathrm{X})$ on TriviaQA.

Applying our Dirichlet calibration map $f_\phi(q)$ reduces TCE to values that are close to zero in nearly all settings, indicating that most of the achievable improvement from task calibration is realized.

Figure 6 summarizes, across all tasks and models, how TCE and ECE correlate with the loss reduction obtained through recalibration. TCE shows a strong, consistent correlation with realized gains across almost all settings, whereas ECE does

Table 4. Task Calibration Error (TCE) using binned estimator for uncalibrated q and Dirichlet-calibrated $f_\phi(q)$ model. All LLMs are severely task miscalibrated which is mitigated by our approximate recalibration method.

Model/Dataset	**Latent** $\mathcal{L}$	$\{0,...,4\}$ + L1 HelpSteer	$\{0,...,5\}$ + L1 STSB	$[C]$ + Exact Match MMLU	When2Call	$\{A,\perp\}$ + BAS SimpleQA	TriviaQA	$\{0,1\}^{\leq C}$ + Hamming MAQA	$\{0,...,C\}^2$ + L1 HelpSteer
Gemma-3-4B	q	0.282	0.323	0.085	0.061	0.196	0.048	0.103	1.216
	$f_\phi(q)$	**0.099**	**0.050**	**0.030**	**0.001**	**0.000**	**0.004**	**0.000**	**0.424**
Gemma-3-12B	q	0.197	0.364	0.040	0.034	0.187	0.029	0.059	1.249
	$f_\phi(q)$	**0.037**	**0.023**	**0.000**	**0.001**	**0.000**	**0.000**	**0.000**	**0.547**
Qwen3-4B	q	0.139	0.692	0.043	0.018	0.182	0.048	0.137	1.375
	$f_\phi(q)$	**0.039**	**0.033**	**0.006**	**0.004**	**0.000**	**0.004**	**0.000**	**0.594**
Qwen3-30B-A3B	q	0.117	0.589	0.083	0.035	0.087	0.009	0.052	1.449
	$f_\phi(q)$	**0.026**	**0.041**	**0.006**	**0.001**	**0.000**	**0.007**	**0.000**	**0.530**

not. Moreover, because TCE is defined in the same units as the task loss, it tracks performance improvements on a directly interpretable scale, while ECE values are harder to relate to concrete changes in task performance.

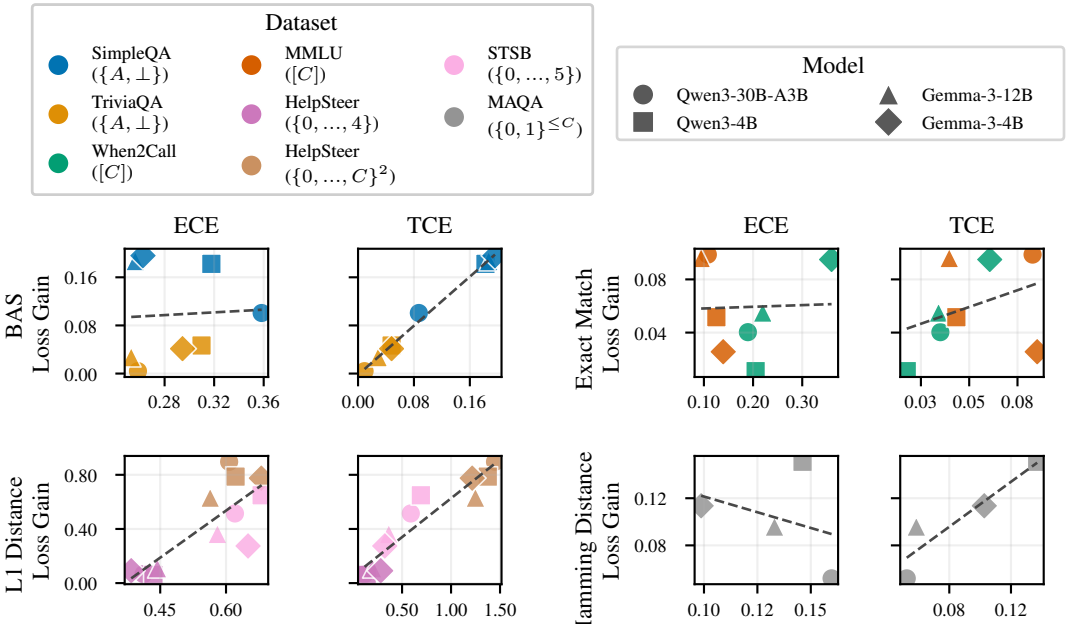


Figure 6. Task Calibration Error (TCE) and ECE for all Tasks

A.3.1. Ablation with Dirichlet Kernel Estimator

To assess the robustness of our conclusions, we also employ the estimator of Popordanoska et al. (2022) with a Dirichlet kernel of fixed bandwidth 0.01. The corresponding results are reported in Table 5. Overall, they match those obtained with the binning-based estimator: TCE is substantially reduced after calibration and is often close to zero, indicating that most of the achievable improvement is recovered. Occasionally, discrepancies arise when the conditional estimator is supported by very few samples (sometimes a single data point), reflecting the sparsity of the empirical distribution in those regions.

B. Implementation details

Here, we present a high-level algorithmic overview of the proposed task-calibrated MBR decoding method. Algorithm 1 describes how the calibration map f_ϕ is fitted on a calibration dataset while Algorithm 2 describes how to perform MBR decoding on the task calibrated latent distribution as shown in Table 1.

Computational Resources. We evaluate all LLMs with default hyperparameters from HuggingFace (top-k, top-p, temperature, etc.) to ensure a realistic practical environment. We perform inference on two types of machines: (i) Nvidia H100 GPUs, and (ii) Nvidia H200 GPUs. Calibration is done on these machines as well, even though they are computationally

Algorithm 1 Fitting task calibration map f_ϕ for a given task T

Require: Calibration dataset of queries and free-form true answers $\{x^{(i)}, y^{(i)}\}_{i=1}^N$, task-dependent latent encoder $g_T : \mathcal{V}^\infty \mapsto \mathcal{L}$, LLM : $\mathcal{V}^\infty \mapsto \mathcal{V}^\infty$
Ensure: Calibration map $f_\phi : \Delta^{C-1} \mapsto \Delta^{C-1}$
for $i \leftarrow 1$ to N **do**
  Generate true latents $\ell^{(i)} \leftarrow g_T(y^{(i)})$
  for $j \leftarrow 1$ to M **do**
    Sample LLM responses $\hat{y}^{(i,j)} \leftarrow \text{LLM}(x^{(i)})$
    Generate sampled latent $\hat{\ell}^{(i,j)} = g_T(y^{(i,j)})$
  end for
  Compute empirical distribution $\hat{p}_c^{(i)} \leftarrow \frac{1}{M}\sum_{j=1}^M \mathbf{1}[\hat{\ell}^{(i,j)} = c]$, where $\hat{p}^{(i)} \in \Delta^{C-1}$
end for
$f_\phi \leftarrow \arg\min_{f_\phi} \frac{1}{N}\sum_{i=1}^N \text{Cross-Entropy}(f_\phi(\hat{p}^{(i)}), \text{one-hot}(y^{(i)}))$

Algorithm 2 MBR decoding with task calibrated LLM f_ϕ for a given task T

Require: Task-calibration map $f_\phi : \Delta^{C-1} \mapsto \Delta^{C-1}$ from Algorithm 1, task-dependent latent encoder $g_T : \mathcal{V}^\infty \mapsto \mathcal{L}$, task-loss $d_T : \mathcal{L} \times \mathcal{L} \mapsto \mathbb{R}$, Query $x \in \mathcal{V}^\infty$, LLM : $\mathcal{V}^\infty \mapsto \mathcal{V}^\infty$
Ensure: MBR decoding output $\ell \in \mathcal{L}$
for $j \leftarrow 1$ to M **do**
  Sample LLM responses $\hat{y}^{(j)} \leftarrow \text{LLM}(x)$
  Generate sampled latents $\hat{\ell}^{(j)} = g_T(y^{(j)})$
end for
Compute empirical distribution $\hat{p}_c \leftarrow \frac{1}{M}\sum_{j=1}^M \mathbf{1}[\hat{\ell}^{(j)} = c]$, where $\hat{p} \in \Delta^{C-1}$
Compute calibrated distribution $q \leftarrow f_\phi(\hat{p})$
Output MBR action $\ell \leftarrow \delta^{\text{MBR}}(q) = \arg\min_{\ell \in \mathcal{L}} \mathbb{E}_{\ell' \sim q}\left[d_T(\ell', \ell)\right]$

Algorithm 3 Binning estimator for Task Calibration Error (TCE)

Require: Predicted latent distributions $\{p_i\}_{i=1}^n$ with $p_i \in \Delta^{C-1}$, observed true latents $\{\ell_i^*\}_{i=1}^n$, number of bins per dimension m.
Ensure: Binned estimate $\widehat{\text{TCE}}$
Partition the simplex Δ^{C-1} into m^{C-1} equally volume bins
for $i \leftarrow 1$ to n **do**
  Assign prediction p_i to its bin B_i
end for

$\widehat{\text{TCE}} \leftarrow 0$
for each occupied bin B **do**
  Collect indices $I_B \leftarrow \{i : B_i = B\}$
  Estimate the true conditional distribution in bin B $\quad \hat{q}_B \leftarrow \frac{1}{|I_B|}\sum_{i \in I_B} \ell_i^*$
  Accumulate the bin contribution $\quad \widehat{\text{TCE}} \leftarrow \widehat{\text{TCE}} + \sum_{i \in I_B} D_{S_T}(p_i \| \hat{q}_B)$
end for

$\widehat{\text{TCE}} \leftarrow \frac{1}{n}\widehat{\text{TCE}}$

Table 5. Task Calibration Error (TCE) using Dirichlet kernel estimator for uncalibrated q and Dirichlet-calibrated $f_\phi(q)$ model

Model/Dataset Latent $\mathcal{L}$		$\{0,...,4\}$ + L1 HelpSteer	$\{0,...,5\}$ + L1 STSB	$[C]$ + Exact Match MMLU	When2Call	$\{A,\perp\}$ + BAS SimpleQA	TriviaQA	$\{0,1\}^{\leq C}$ + Hamming MAQA	$\{0,...,C\}^2$ + L1 HelpSteer
Gemma-3-4B	q	0.266	0.340	**0.124**	0.112	0.196	0.059	0.678	**0.212**
	$f_\phi(q)$	**0.182**	**0.072**	0.145	**0.001**	**0.000**	**0.014**	**0.000**	0.299
Gemma-3-12B	q	0.238	0.419	0.129	0.068	0.187	0.035	1.036	0.118
	$f_\phi(q)$	**0.114**	**0.048**	**0.019**	**0.003**	**0.000**	**0.002**	**0.206**	**0.115**
Qwen3-4B	q	0.253	0.719	0.094	0.047	0.182	0.052	0.806	0.143
	$f_\phi(q)$	**0.102**	**0.077**	**0.037**	**0.013**	**0.000**	**0.004**	**0.000**	**0.109**
Qwen3-30B-A3B	q	0.223	0.617	0.148	0.055	0.101	0.024	0.836	0.155
	$f_\phi(q)$	**0.061**	**0.082**	**0.031**	**0.004**	**0.001**	**0.009**	**0.072**	**0.075**

Table 6. Datasets with tasks T, latent structures $\mathcal{L}$, and task-dependent loss functions d_T.

Task	Ordinal Regression		Classification		Answer-or-Abstain		Open-Set Prediction	Vector Ordinal Regression
Dataset	Helpsteer	STSB	MMLU	When2Call	SimpleQA	TriviaQA	MAQA	Helpsteer
Latent Structure $\mathcal{L} \cong$	$\{0,\dots,4\}$	$\{1,\dots 5\}$	{(a), (b), (c), (d)}	{"call tool", "request info", "abstain"}	$\{A,\perp\}$	$\{A,\perp\}$	$\{0,1\}^{\leq C}$	$\{0,\dots,4\}^2$
Task-Dependent Loss $d_T(\ell,\ell')$	$\lvert\ell-\ell'\rvert$	$\lvert\ell-\ell'\rvert$	$\mathbf{1}[\ell\neq\ell']$	$\mathbf{1}[\ell\neq\ell']$	$d_t^{\text{BAS}}(\ell,\ell')$	$d_t^{\text{BAS}}(\ell,\ell')$	$\mathbf{1}[\ell_i\neq\ell'_i]$ for binary decisions about including answer i	$\lVert\ell-\ell'\rVert_1$

lightweight and can easily be done on CPU-only machines as well. While inference (sampling responses from the LLM) takes up to multiple days for the biggest datasets with longest prompts and largest LLMs, calibration runs finish within a few minutes on our machines.

Calibration When fitting calibrators (temperature scaling and Dirichlet calibration) using NLL loss, we transform the logarithm of the predicted latent beliefs $\hat{p}(\mathrm{X})$. We add $\epsilon = 10^{-12}$ to these beliefs to prevent taking the logarithm of zero. In particular, temperature scaling with a positive temperature τ is realized by optimizing over the logarithm of the temperature $t = \log\tau$:

$$f_t(q) = \mathrm{softmax}(\exp(t)^{-1}\log(q+\epsilon)), \qquad t\in\mathbb{R}. \tag{12}$$

Similarly, Dirichlet calibration is realized as:

$$f_{\boldsymbol{W},\boldsymbol{b}}(q) = \mathrm{softmax}(\boldsymbol{W}\log(q+\epsilon)+\boldsymbol{b}), \qquad \boldsymbol{W}\in\mathbb{R}^{C\times C}, \boldsymbol{b}\in\mathbb{R}^C. \tag{13}$$

Datasets. We detail the datasets, corresponding tasks and the associated latent structures and loss functions in Table 6. Here, we overload elements ℓ which do not represent one-hot vectors in this context but instead the true corresponding latent meaning. In practice, each one-hot representation directly corresponds to one latent interpretation, so the one-hot representation of a structure and its "true meaning" are isomorphic (denoted by $\cong$).

We also briefly describe each dataset and the associated LLM task. (i) For Helpsteer, the LLM is shown a prompt and corresponding response and should rate the response according to one or more criteria. For the ordinal regression task, we choose the criterion "correctness" and let the LLM output a natural number for a rating between 0 and 4. (ii) For STSB, the LLM is shown two sentences and it should rate their similarity on a scale on 1 to 5. We again only consider natural numbers through rounding to the closest integer. (iii) For MMLU, the LLM is shown a question and four potential answers enumerated with (a) to (d). The LLM should select the correct answer among the provided candidates. (iv) For When2Call, we present the LLM with a prompt and a set of available tools. The LLM should then assess whether to call one of the tools, request more information from user or abstain from answering alltogether. (v) For SimpleQA and (vi) TriviaQA, we present the LLM with a question and ask for the answer. Whether the LLM action is to answer A or abstain $\perp$ is determined by the relative frequency of the semantic majority answer (see Section B.1). (vii) For MAQA, we present the LLM with an ambiguous question that permits multiple answers. The LLM should list all valid answers. We construct a set of all observed answers in the generated responses and the truth and then decompose this decision task into binary decision tasks of whether to include an individual answer i in the final response. (viii) For Vector-valued Ordinal Regression, we again

use Helpsteer and show the LLM a prompt and a corresponding response. It now should rate the response in terms of two criteria simultaneously ("correctness", "helpfulness") by outputting two integer scores in $\{0, \ldots, 4\}$.

For all datasets except MAQA, we use HuggingFace as a data source: For STSB, we use mteb/stsbenchmark-sts, for MMLU, we use cais/mmlu, for When2Call, we use nvidia/When2Call, for SimpleQA, we use google/simpleqa-verified, for TriviaQA, we use mandarjoshi/trivia_qa, for Helpsteer, we use nvidia/HelpSteer. All datasets correspond to their latest available versions as of May 2026. For MAQA we use the official repository MAQA-Data.

B.1. Mapping Language Outputs to Latent Structures

Here, we detail how we realize the mappings g_T to free-form text LLM responses into the corresponding latent structures. As our core assumption is that the LLM outputs free-form text that is embedded in the latent structure, i.e. the LLM is able to reasonable address the task we prompt it with, we discard answers that can not properly parsed through any mapping g_T. These occurrences are extremely rare as the instruction-tuned LLMs studied in this work respect the instruction prompts sufficiently well. Therefore, g_T can often be implemented just with simple Regular Expressions or lightweight Natural Language Inference models. In general, for all structures $\mathcal{L}$ considered in this work, we empirically find that the mapping g_T does not lose any semantically relevant information and succesfully extracts the correct meaning from a response in almost all cases.

Ordinal Regression Models are instructed to follow a specific output format as seen in Section F. The LLM's output is post-processed by an auxiliary language model extracting the integer response (again, see Section F) and then parsed using a Regex.

Classification Models are instructed to follow a specific output format as seen in Section F. The answer is then passed using a Regex.

Answer-or-Abstain Similar to Kuhn et al. (2023), we use DeBERTa (He et al., 2021) as an entailment model to group the individual free-form text responses into semantic clusters that form the support of the domain over which the most frequent answer is selected. All elements falling into the cluster of the most frequent answer are mapped to `Answer` while all others are mapped to $\perp$.

Open Set Prediction Given an answer string s, we want to extract all *semantically distinct* answers in that string. As an example to the question "Which ocean borders the USA? " the string $s =$" Both the Atlantic and Pacific Ocean border the USA. "gets mapped to $\{\texttt{Atlantic}, \texttt{Pacific}\}$ by g_T. Concretely g_T is implemented using `gpt-5-mini` with prompts in (Section F).

B.2. Task Calibration Error

Binning Estimator for TCE We assume we have m^{C-1} equally spaced bins on the simplex. For each prediction $p_i \in \Delta^{C-1}$, we assign a bin index B_i. For each occupied bin B, let

$$I_B = \{i : B_i = B\}.$$

The empirical conditional label distribution in bin B is

$$\hat{q}_B = \frac{1}{|I_B|} \sum_{i \in I_B} \ell_i,$$

where ℓ_i is the one-hot encoding of the observed label as explained in Section 2. The binned TCE estimator is then

$$\widehat{\mathrm{TCE}} = \frac{1}{n} \sum_{B} \sum_{i \in I_B} D(p_i || \hat{q}_B).$$

The procedure is also shown in Algorithm 3.

C. Related Work Extended

C.1. BAS: A Decision-Theoretic Approach to Evaluating LLM Confidence

The BAS metric of Wu et al. (2026) can be viewed as a special case of our latent-structure decision framework. Let $\mathcal{L} = \{\mathrm{A}, \perp\}$, where A denotes `answer` and $\perp$ denotes `abstain`. We identify both the latent and action space with these two selective decisions. For a fixed threshold $t \in [0, 1)$, define the task loss

$$d_T^{\mathrm{BAS}}(a, b) = \begin{cases} 0, & b = \perp, \\ -1, & b = \mathrm{A},\ a = \mathrm{A}, \\ \frac{t}{1-t}, & b = \mathrm{A},\ a = \perp\ . \end{cases} \tag{14}$$

Importantly, the loss is asymmetric, and a corresponds to the true decision while b denotes the model prediction. For a given x, let $\hat{p}(x)_{\mathrm{A}}$ denote the predictive probability that answering is correct. The corresponding Bayes decision is

$$\delta_t^{\mathrm{MBR}}(\hat{p}(x)) = \arg\min_{b \in \{\mathrm{A}, \perp\}} \mathbb{E}_{a \sim \hat{p}(x)} \left[d_t^{\mathrm{BAS}}(a, b) \right].$$

This coincides with the BAS answer-or-abstain policy. To see this, we compare the two possible actions:

$$\mathbb{E}_{a \sim \hat{p}(x)}[d_t^{\mathrm{BAS}}(a, \perp)] = 0,$$

and

$$\mathbb{E}_{a \sim \hat{p}(x)}[d_t^{\mathrm{BAS}}(a, \mathrm{A})] = -\hat{p}(x)_{\mathrm{A}} + \frac{t}{1-t}\hat{p}(x)_{\perp}.$$

Thus, choosing $b = \mathrm{A}$ is optimal iff

$$-\hat{p}(x)_{\mathrm{A}} + \frac{t}{1-t}\hat{p}(x)_{\perp} \leq 0.$$

Using $\hat{p}(x)_{\perp} = 1 - \hat{p}(x)_{\mathrm{A}}$, this is equivalent to

$$-\hat{p}(x)_{\mathrm{A}} + \frac{t}{1-t}(1 - \hat{p}(x)_{\mathrm{A}}) \leq 0 \iff \hat{p}(x)_{\mathrm{A}} \geq t.$$

Therefore,

$$\delta_t^{\mathrm{MBR}}(\hat{p}(x)) = \begin{cases} \mathrm{A}, & \hat{p}(x)_{\mathrm{A}} \geq t, \\ \perp, & \text{otherwise}, \end{cases}$$

which exactly recovers the BAS answer-or-abstain rule. In our experiments, we set $t = 0.25$, and the implementation of the mapping g_T is detailed in Section B.1.

D. On Calibration And Uncertainty Quantification.

Model calibration enables *estimation of the average loss incurred by a decision rule*. Importantly, it does not ensure good per-sample decision-making. This is particularly relevant in the context of uncertainty quantification (UQ), a field that the literature often confounds with model calibration (Minderer et al., 2021; Nixon et al., 2019). UQ methods are typically evaluated by how well they predict model errors or out-of-distribution examples. In the context of LLMs, hallucination detection is a well-established surrogate task for evaluating UQ performance. Here, the uncertainty associated with an individual instance is thresholded to predict if a response is a confabulation. Hence, the reported metrics often depend on the relative rankings of inputs based on the predicted uncertainty. Calibrating a (latent) predictive distribution has no formal implications on either this relative ranking or the per-instance thresholding. In particular, calibration does not claim to improve the discrimination of individual predictions into correct and incorrect answers by using uncertainty. Instead, it ensures that the expected loss under a decision rule computed from the model's beliefs matches the true expected loss. In the context of our work, our calibration map f^* further ensures that MBR is the optimal decision rule with respect to some loss function. This does not imply that, for example, the maximum probability or similar statistics computed from the calibrated latent beliefs (e.g. Bayes risk as proposed for UQ by Tomov et al. (2026)) are more reliable to detect hallucinations or forecast model errors. For an in-depth discussion of calibration and its relationship to reliability and uncertainty quantification, with illustrative examples, we refer to the excellent blog post by Xia (2026).

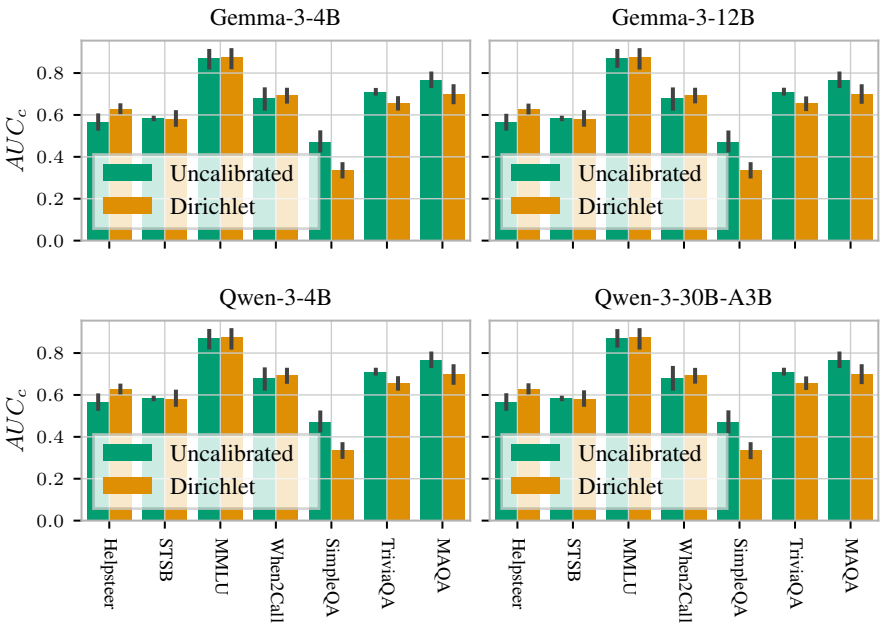


Figure 7. Uncertainty quantification (UQ) AUC_c scores for Bayes risk of Equation (2) and the task-specific loss d_T incurred by the MBR decoding action for different tasks and models. There is no clear trend for task calibration: On some datasets it improves UQ, on others, it does not. This aligns with our discussion.

Nonetheless, we showcase how task calibration affects uncertainty quantification for completeness in Figure 7. Here, we measure the concordance statistic AUC_c that assesses how well a continuous uncertainty estimator can predict a continuous target variable. The concordance AUC_c is an estimate of $\mathbb{P}(y_i > y_j \mid \hat{y}_i > \hat{y}_j)$. We follow (Therneau & Atkinson, 2024) and discard ties on y, i.e. $y_i = y_j$ and treat ties on the estimator, i.e. $y_i = y_j$ with a score of $\frac{1}{2}$. The value of the resulting score can be interpreted analogously to the traditional Area-under-the-Precision-Recall-Curve (AUC-ROC) metrics, with 0.5 corresponding to random chance and 1 to perfect ranking ability. In our case, we evaluate the Bayes risk (Equation (2) like Tomov et al. (2026) as a proxy for per-instance uncertainty and measure how well it predicts the task dependent loss d_T. We observe that on some tasks, there surprisingly is an improved capacity to estimate uncertainty from the task calibrated LLM beliefs (MMLU, When2Call, Helpsteer). On other tasks, however, the quality of the uncertainty estimate deteriorates. Overall, there is no trend that is consistent among all tasks. This aligns well with our discussion: Calibration enables improved decision making in terms of average loss. It does not improve per-instance decisions like predicting the error caused by the prediction from the Bayes risk as a proxy for uncertainty.

E. Proofs

Lemma 3.3. *For any loss function $d_T : \mathcal{L} \times \mathcal{L} \mapsto \mathbb{R}$ among all decision strategies $\delta' : \Delta^{C-1} \mapsto \mathcal{L}$, the MBR decision δ^{MBR} on $f^*_{g_T}(\hat{p}(\mathrm{X}))$ is optimal in terms of the expected incurred loss w.r.t the true distribution $p^*(\mathrm{X}, \mathrm{Y})$ on any subset $P \subseteq \Delta^{C-1}$:*

$$\begin{aligned} &\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta^{\mathrm{MBR}}(f^*_{g_T}(\hat{p}(\mathrm{X})))) \mid \hat{p}(\mathrm{X}) \in P\right] \\ &\leq \mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta'(\hat{p}(\mathrm{X}))) \mid \hat{p}(\mathrm{X}) \in P\right]. \end{aligned} \tag{8}$$

Proof. Define

$$\mathrm{S} := \hat{p}(\mathrm{X}), \qquad \mathrm{L} := g_T(Y).$$

Then S is the observable model output, and for any realization of S, $f^*_{g_T}(\mathrm{S})$ denotes the true conditional distribution of L given S.

Now for any δ':

$$\begin{aligned}
\mathbb{E}_{\mathrm{X},\mathrm{Y}}[d_T(g_T(\mathrm{Y}), \delta'(\hat{p}(\mathrm{X}))) \mid \hat{p}(\mathrm{X}) \in P] &= \mathbb{E}_{\mathrm{X},\mathrm{Y}}[d_T(\mathrm{L}, \delta'(\mathrm{S})) \mid \mathrm{S} \in P] && (15)\\
&= \mathbb{E}_{\mathrm{S}|\mathrm{S}\in P}[\mathbb{E}_{\mathrm{X},\mathrm{Y}}[d_T(\mathrm{L}, \delta'(\mathrm{S})) \mid \mathrm{S}, \mathrm{S} \in P]] && (16)\\
&= \mathbb{E}_{\mathrm{S}|\mathrm{S}\in P}\left[\mathbb{E}_{\ell\sim f^*_{g_T}(\mathrm{S})}[d_T(\ell, \delta'(\mathrm{S})) \mid \mathrm{S}, \mathrm{S} \in P]\right] && (17)\\
&\geq \mathbb{E}_{\mathrm{S}|\mathrm{S}\in P}\left[\min_{a\in\mathcal{L}} \mathbb{E}_{\ell\sim f^*_{g_T}(\mathrm{S})}[d_T(\ell, a) \mid \mathrm{S}, \mathrm{S} \in P]\right] && (18)\\
&= \mathbb{E}_{\mathrm{S}|\mathrm{S}\in P}\left[\mathbb{E}_{\ell\sim f^*_{g_T}(\mathrm{S})}\left[d_T\big(\ell, \delta^{\mathrm{MBR}}(f^*_{g_T}(\mathrm{S}))\big) \,\middle|\, \mathrm{S}, \mathrm{S} \in P\right]\right] && (19)\\
&= \mathbb{E}_{\mathrm{S}|\mathrm{S}\in P}\left[\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T\big(\mathrm{L}, \delta^{\mathrm{MBR}}(f^*_{g_T}(\mathrm{S}))\big) \mid \mathrm{S}, \mathrm{S} \in P\right]\right] && (20)\\
&= \mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T\big(\mathrm{L}, \delta^{\mathrm{MBR}}(f^*_{g_T}(\mathrm{S}))\big) \mid \mathrm{S} \in P\right]. && (21)
\end{aligned}$$

Substituting back $\mathrm{L} = g_T(\mathrm{Y})$ and $\mathrm{S} = \hat{p}(\mathrm{X})$ yields

$$\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T\big(g_T(\mathrm{Y}), \delta^{\mathrm{MBR}}(f^*_{g_T}(\hat{p}(\mathrm{X})))\big) \mid \hat{p}(\mathrm{X}) \in P\right] \leq \mathbb{E}_{\mathrm{X},\mathrm{Y}}[d_T(g_T(\mathrm{Y}), \delta'(\hat{p}(\mathrm{X}))) \mid \hat{p}(\mathrm{X}) \in P]. \tag{22}$$

In Equation (16) we apply the tower rule by conditioning on $\mathrm{S} = \hat{p}(\mathrm{X})$. In Equation (17) we use that, conditional on S, the random variable $\mathrm{L} = g_T(\mathrm{Y})$ is distributed according to $f^*_{g_T}(\mathrm{S})$ when taking the expectation over X, Y with respect to the true data generation distribution. In Equation (18) we use that, for each realization of S, the conditional risk of the action $\delta'(\mathrm{S})$ is at least the minimum achievable conditional risk over all actions $a \in \mathcal{L}$. In Equation (19) we use the definition of the Minimum Bayes Risk decision rule. Finally, Equation (20) uses again the tower rule. Hence, no other decision strategy depending only on $\hat{p}(\mathrm{X})$ can attain lower expected task loss for any condition on $\hat{p}(\mathrm{X})$. □

Theorem 3.4. *For any loss function $d_T : \mathcal{L} \times \mathcal{L} \mapsto \mathbb{R}$ among all decision rules $\delta' : \Delta^{C-1} \mapsto \mathcal{L}$, the MBR decision δ^{MBR} on $f^*_{g_T}(\hat{p}(\mathrm{X}))$ is optimal in terms of the expected incurred loss w.r.t the true distribution $p^*(\mathrm{X}, \mathrm{Y})$:*

$$\begin{aligned}
&\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta^{MBR}(f^*_{g_T}(\hat{p}(\mathrm{X})))\right]\\
&\leq\ \mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[d_T(g_T(\mathrm{Y}), \delta'(\hat{p}(\mathrm{X})))\right].
\end{aligned} \tag{9}$$

Proof. This follows directly from Lemma 3.3 by setting $P = \Delta^{C-1}$. □

Proposition E.1. *For a given parametric calibration map family $\mathcal{F}$, the map with minimal negative log-likelihood minimizes the expected KL-divergence to $f^*_{g_T}$:*

$$\arg\min_{f_\phi\in\mathcal{F}} \mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[-g_T(\mathrm{Y})^T \log f_\phi(\hat{p}(\mathrm{X}))\right] = \arg\min_{f_\phi\in\mathcal{F}} \mathbb{E}_{\mathrm{X}}\left[\mathrm{KL}\left[f^*_{g_T}(\hat{p}(\mathrm{X}) \| f_\phi(\hat{p}(\mathrm{X}))\right]\right]. \tag{23}$$

Proof.

$$\begin{aligned}
\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[-g_T(\mathrm{Y}^T)\log f_\phi(\hat{p}(\mathrm{X})\right] &= \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X},\mathrm{Y}}\left[-g_T(\mathrm{Y})^T\log f_\phi(\hat{p}(\mathrm{X})\right] \mid \hat{p}(\mathrm{X})\right] && (24)\\
&= \mathbb{E}_{\hat{p}(\mathrm{X})}\left[-\mathbb{E}_{\mathrm{X},\mathrm{Y}}[g_T(\mathrm{Y}) \mid \hat{p}(\mathrm{X})]^T\log f_\phi(\hat{p}(\mathrm{X}))\right] && (25)\\
&= \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[-(f^*_{g_T}(\hat{p}(\mathrm{X})))^T\log f_\phi(\hat{p}(\mathrm{X}))\right]\right] && (26)\\
&= \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[-(f^*_{g_T}(\hat{p}(\mathrm{X})))^T\log f_\phi(\hat{p}(\mathrm{X}))\right]\right]\\
&\quad + \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))^T\log f^*_{g_T}(\hat{p}(\mathrm{X}))\right]\right] && (27)\\
&\quad - \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))^T\log f^*_{g_T}(\hat{p}(\mathrm{X}))\right]\right]\\
&= \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[(f^*_{g_T}(\hat{p}(\mathrm{X})))^T\log\frac{f^*_{g_T}(\hat{p}(\mathrm{X}))}{f_\phi(\hat{p}(\mathrm{X}))}\right]\right] && (28)\\
&\quad - \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))^T\log f^*_{g_T}(\hat{p}(\mathrm{X}))\right]\right]\\
&= \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[\mathrm{KL}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))\|f_\phi(\hat{p}(\mathrm{X}))\right]\right]\right]\\
&\quad + \mathbb{E}_{\hat{p}(\mathrm{X})}\left[\mathbb{E}_{\mathrm{X}|\hat{p}(\mathrm{X})}\left[\mathbb{H}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))\right]\right]\right] && (29)\\
&= \mathbb{E}_{\mathrm{X}}\left[\mathrm{KL}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))\|f_\phi(\hat{p}(\mathrm{X}))\right]\right]\\
&\quad + \mathbb{E}_{\mathrm{X}}\left[\mathbb{H}\left[f^*_{g_T}(\hat{p}(\mathrm{X}))\right]\right] && (30)
\end{aligned}$$

The second expected entropy term does not depend on $f_\phi(\hat{p}(\mathrm{X}))$ and hence minimizing the cross entropy also minimizes the expected KL divergence. □

Proposition E.2. *The generalized entropy & divergence induced by $S_T(q,\ell) = d_T(\ell, \delta^{MBR}(q))$ are:*

$$H_{S_T}(q) = R_{\mathrm{Bayes}}(\delta^{MBR}(q), q), \quad D_{S_T}(q\,\|\,r) = R_{\mathrm{Bayes}}(\delta^{MBR}(q), r) - R_{\mathrm{Bayes}}(\delta^{MBR}(r), r). \tag{31}$$

Proof. We first recall the definitions of the generalized entropy and divergence induced by a scoring function S_T:

$$\begin{aligned}
H_{S_T}(q) &= \mathbb{E}_{\ell\sim q}[S_T(q,\ell)], && (32)\\
D_{S_T}(q\|r) &= \mathbb{E}_{\ell\sim r}[S_T(q,\ell)] - \mathbb{E}_{\ell\sim r}[S_T(r,\ell)]. && (33)
\end{aligned}$$

Using $S_T(q,\ell) = d_T(\ell, \delta^{\mathrm{MBR}}(q))$, we obtain

$$\begin{aligned}
H_{S_T}(q) &= \mathbb{E}_{\ell\sim q}[S_T(q,\ell)] && (34)\\
&= \mathbb{E}_{\ell\sim q}\big[d_T(\ell, \delta^{\mathrm{MBR}}(q))\big] && (35)\\
&= R_{\mathrm{Bayes}}(\delta^{\mathrm{MBR}}(q), q). && (36)
\end{aligned}$$

Likewise,

$$\begin{aligned}
D_{S_T}(q\|r) &= \mathbb{E}_{\ell\sim r}[S_T(q,\ell)] - \mathbb{E}_{\ell\sim r}[S_T(r,\ell)] && (37)\\
&= \mathbb{E}_{\ell\sim r}\big[d_T(\ell, \delta^{\mathrm{MBR}}(q))\big] - \mathbb{E}_{\ell\sim r}\big[d_T(\ell, \delta^{\mathrm{MBR}}(q))\big] && (38)\\
&= R_{\mathrm{Bayes}}(\delta^{\mathrm{MBR}}(q), r) - R_{\mathrm{Bayes}}(\delta^{\mathrm{MBR}}(r), r). && (39)
\end{aligned}$$

□

Proposition 3.5. *The Task Calibration Error (TCE) is defined by decomposing the D_{S_T} between the predicted latent distribution $\hat{p}(\mathrm{X})$ and true latent distribution $p^*(\mathrm{L}|\mathrm{X})$ as:*

$$\begin{aligned}
&\mathbb{E}_X\left[D_{S_T}\big(\hat{p}(\mathrm{X})\,\|\,p^*(\mathrm{L}\mid\mathrm{X})\big)\right] = \\
&\qquad \underbrace{\mathbb{E}_X\left[D_{S_T}\big(\hat{p}(\mathrm{X})\,\|\,f^*_{g_T}(\hat{p}(\mathrm{X}))\big)\right]}_{\textit{Task Calibration Error (TCE)}} \\
&\qquad + \ \mathbb{E}_X\left[D_{S_T}\big(f^*_{g_T}(\hat{p}(\mathrm{X}))\,\|\,p^*(\mathrm{L}\mid\mathrm{X})\big)\right].
\end{aligned} \tag{11}$$

Proof. Follows directly from the known decomposition (Kull & Flach, 2015). □

F. Prompts

Prompt 1. g_T Prompt for Set-Case

```
You extract explicit, semantically distinct atomic answers from a model answer.

Rules:
- Extract only what the answer explicitly states; no inference.
- Split coordinated lists (commas, and/or, slashes) into separate items.
- Merge synonyms/rephrasings; keep one canonical surface form, preserving key head nouns.
- Remove leading articles/hedges; remove trailing punctuation/parentheticals unless part
    of the name.
- Ignore explanations/justifications; include quantities only if the quantity itself is
    the item.
- If no explicit item appears (I don't know, etc.), output: I don't know.
- Output strictly: one item per line, no bullets or extra text.

Follow the output format in the examples.

Example 1
Q: Which oceans border the USA?
A: The Atlantic and Pacific Oceans.
Output:
Atlantic Ocean
Pacific Ocean

Example 2
Q: What are the official languages of Switzerland?
A: German, French, Italian, and Romansh are the four official languages.
Output:
German
French
Italian
Romansh



Q: {question}
A: {answer}

Task: List all explicit, distinct answers from the model answer, one per line.
```

Prompt 2. Prompt for When2Call

```
You are an expert in composing functions. You are given a question and a set of possible
    functions.
Based on the question, decide what the assistant should do.
If a function can directly answer the question, you may call it.
If the question lacks required parameters, ask a follow-up question.
If none of the functions can be used to answer the question, say so.
If you decide to invoke a function, you MUST put it in the format of [func_name1(
    params_name1=params_value1, params_name2=params_value2...), func_name2(params)].

You may briefly explain your reasoning first, but your final line must be exactly one of:
FINAL_ACTION: tool_call
FINAL_ACTION: request_for_info
FINAL_ACTION: cannot_answer

Here is a list of functions in JSON format that you can invoke.
{tools}

Question: {question}
```

Prompt 3. Prompt for MMLU

```
What is the correct answer to the following multiple choice question? You may very briefly
     explain your reasoning first but then give your final answer as one of the letters as
     "Final answer: letter".
Question: <question>
A. <choice 1>
B. <choice 2>
C. <choice 3>
D. <choice 4>
```

Prompt 4. Prompt for HelpSteer

```
Assess the following model response for {criterion_name} ({criterion_description})
on a scale from 0 to 4, where higher is better.

You may briefly explain your reasoning first, but your final line must be exactly one
number from 0 to 4 in the format: "Final answer: <number>".

Model response: {question}
```

Prompt 5. Prompt for STSB

```
Measure the sentence similarity between two sentences measured on a continous scale of 0
to 5 with 0 being unrelated and 5 being related.
Now for the following two sentences, what is the similarity score?

You may briefly explain your reasoning first, but your final line must be exactly one
number from 0 to 5 in the format: "Final answer: <number>".

Sentences: {question}
```

Prompt 6. Prompt for TriviaQA/SimpleQA/MAQA

```
{question}. Only provide the answer without explanation.
```

Prompt 7. Prompt for Helpsteer, Two Criteria

```
Assess the following model response for {criteria_names_and_descriptions} on a scale
from 0 to 4 each, where higher is better.

You may briefly explain your reasoning first, but your final line must be exactly one
number from 0 to 4 for each criterion in the format: "Final answer: {format_string}".

Model response: {question}

Score:
```

Prompt 8. Prompt for Extracting Integers from LLM Response

```
You extract the final numeric answer from a model response.

Rules:
- Read the full response and identify the final numeric value the answer commits to.
- Ignore intermediate calculations, option numbers, and numbers mentioned only in
    reasoning.
- Return exactly one bare number with no explanation or surrounding text.
- If the response does not clearly provide one final numeric answer, return UNKNOWN.

Question: {question}
Answer: {answer}

Task: Return only the final numeric answer stated in the answer.
```

G. Additional Figures

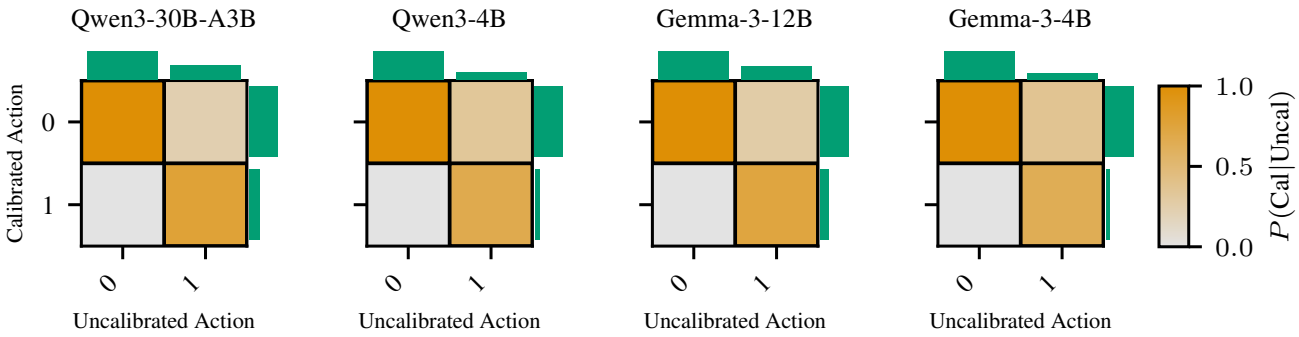
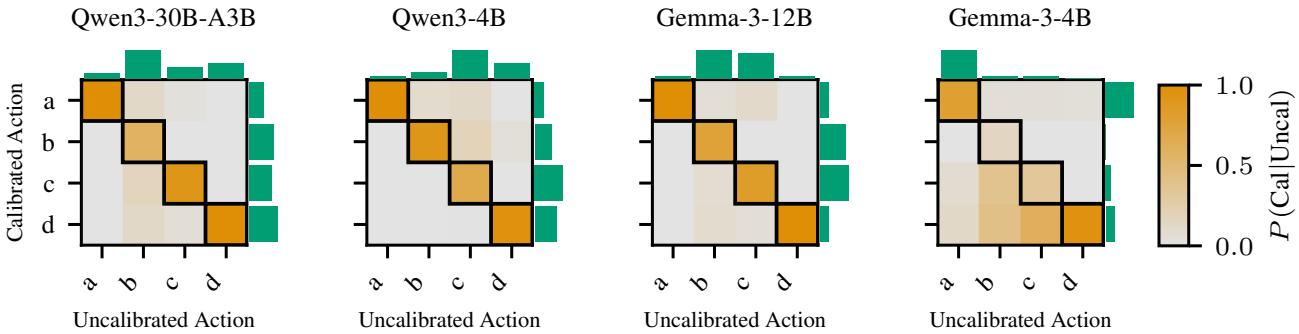
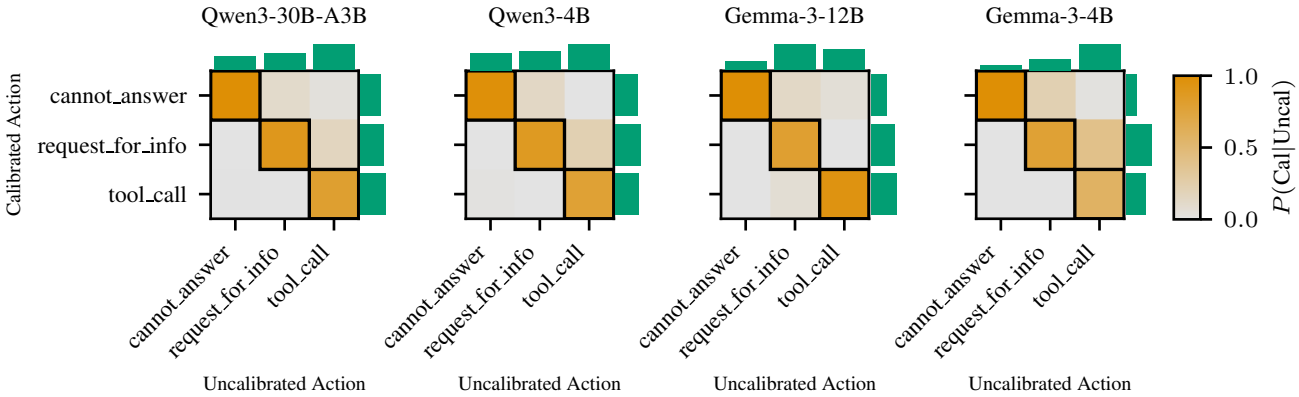


Figure 8. Action movement for MAQA ($\{0, 1\}^{\leq C}$).

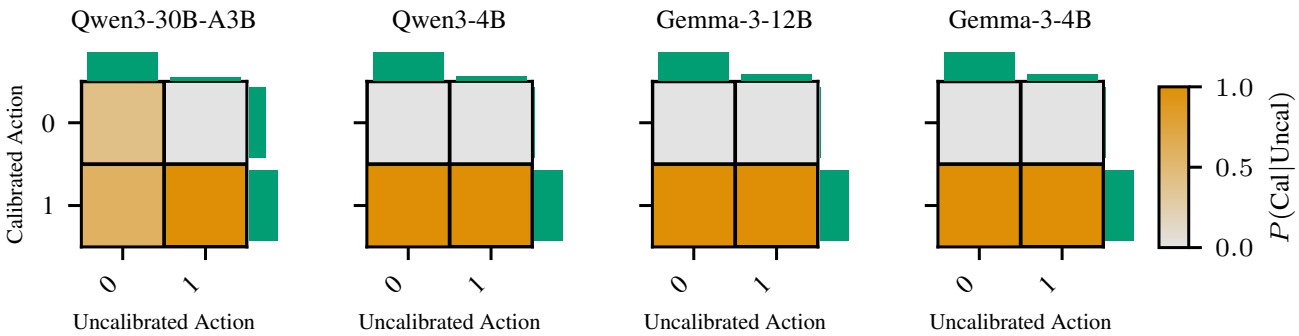


(a) MMLU ($[C]$)

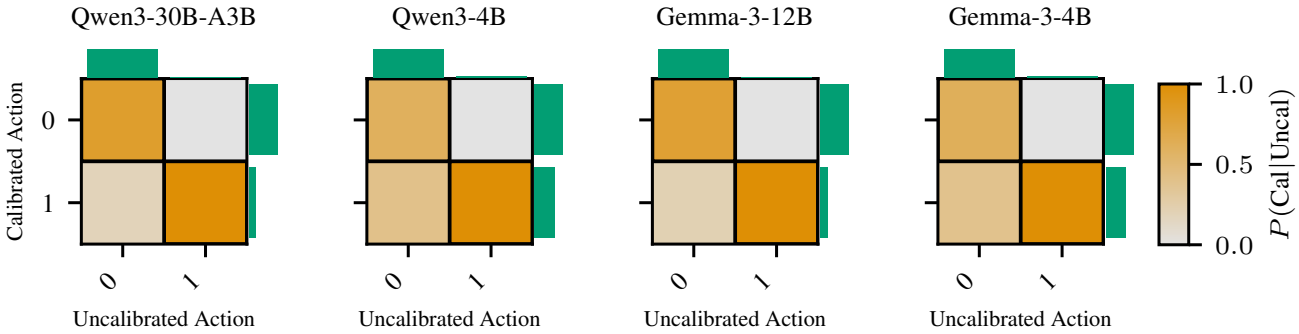


(b) When2Call ($[C]$)

Figure 9. Action movement for classification tasks.

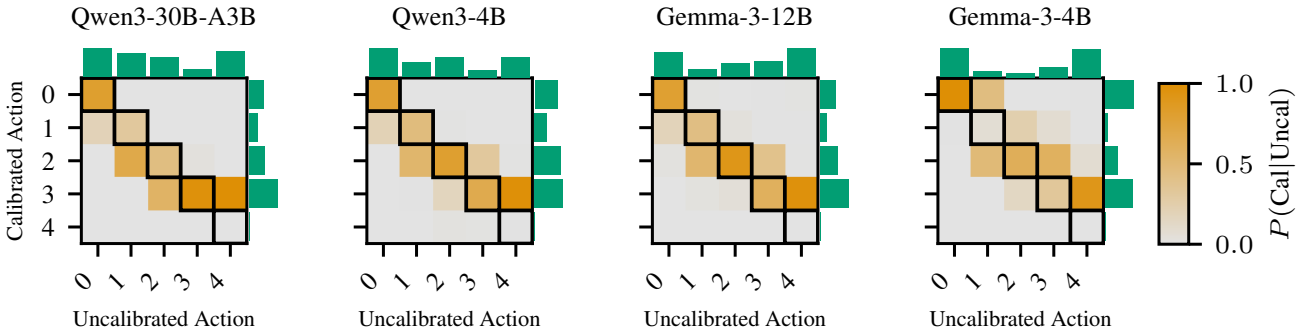


(a) SimpleQA ($\{A, \perp\}$)

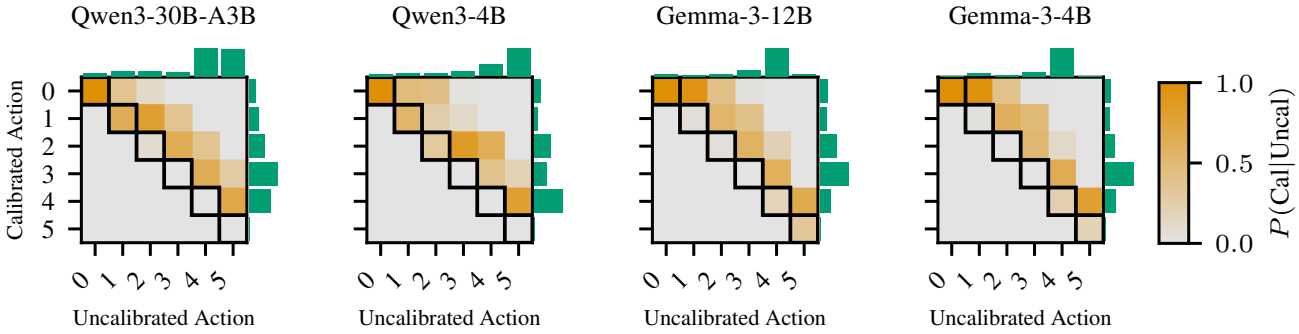


(b) TriviaQA ($\{A, \perp\}$)

Figure 10. Action movement for abstention tasks.



(a) HelpSteer ($\{0, \ldots, C\}$)



(b) STSB ($\{0, \ldots, C\}$)

Figure 11. Action movement for ordinal/interval tasks.

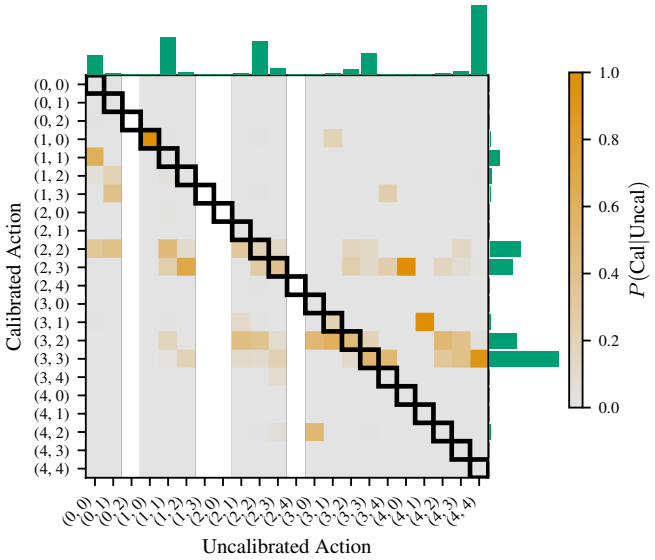


Figure 12. HelpSteer ($\{0, \ldots, C\}^2$) Qwen3-4B

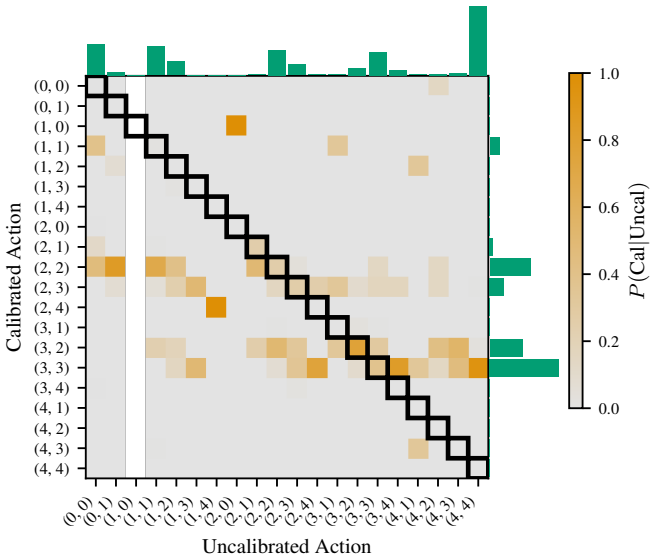


Figure 13. HelpSteer ($\{0, \ldots, C\}^2$) Qwen3-30B-A3B

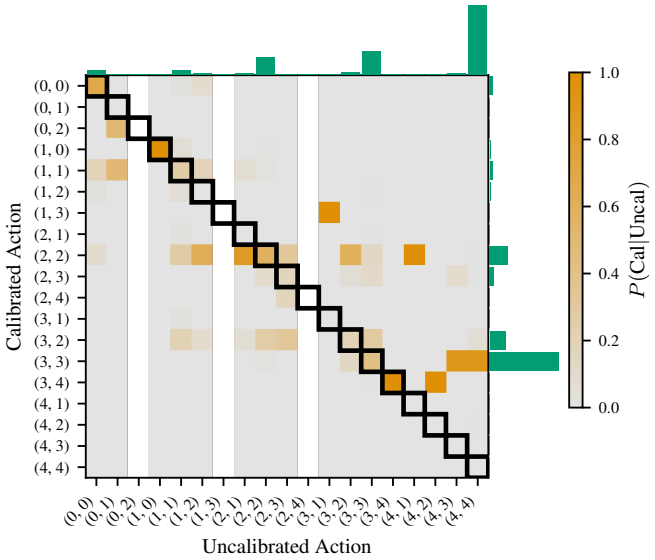


Figure 14. HelpSteer ($\{0, \ldots, C\}^2$) Gemma3-4B

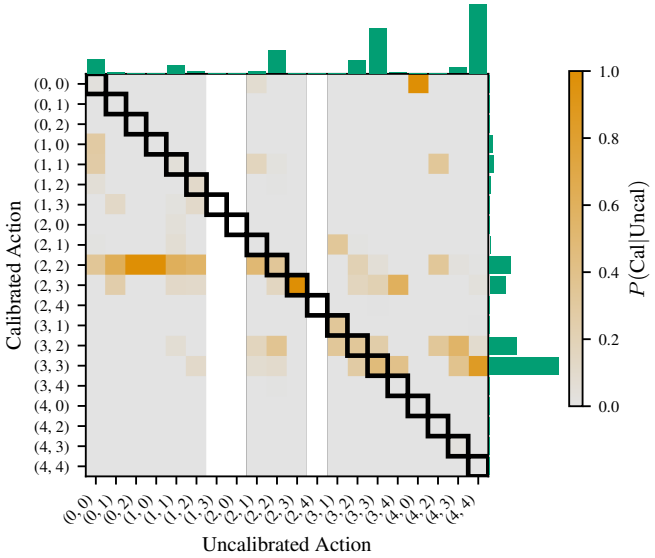


Figure 15. HelpSteer ($\{0, \ldots, C\}^2$) Gemma-3-12B