
Linguistic Calibration of Long-Form Generations

Neil Band¹ Xuechen Li¹ Tengyu Ma¹ Tatsunori Hashimoto¹

Abstract

Language models (LMs) may lead their users to make suboptimal downstream decisions when they confidently hallucinate. This issue can be mitigated by having the LM verbally convey the probability that its claims are correct, but existing models cannot produce long-form text with calibrated confidence statements. Through the lens of decision-making, we define linguistic calibration for long-form generations: an LM is linguistically calibrated if its generations enable its users to make calibrated probabilistic predictions. This definition enables a training framework where a supervised finetuning step bootstraps an LM to emit long-form generations with confidence statements such as “I estimate a 30% chance of...” or “I am certain that...”, followed by a reinforcement learning step which rewards generations that enable a user to provide calibrated answers to related questions. We linguistically calibrate Llama 2 7B and find in automated and human evaluations of long-form generations that it is significantly more calibrated than strong finetuned factuality baselines with comparable accuracy. These findings generalize under significant domain shifts to scientific and biomedical questions and to an entirely held-out person biography generation task. Our results demonstrate that long-form generations may be calibrated end-to-end by constructing an objective in the space of the predictions that users make in downstream decision-making.

1 Introduction

The claims made by language models (LMs) are increasingly used to inform real-world decisions, *e.g.*, what to order at a restaurant, what information to provide someone else about a topic, or which code completion to accept. However, LMs have knowledge gaps which manifest as hallucinations

¹Department of Computer Science, Stanford University. Correspondence to: Neil Band <nband@cs.stanford.edu>.

(Ji et al., 2023; Huang et al., 2023). Currently, when an LM lacks knowledge about a topic, it will do one of two things: hallucinate incorrect claims with complete confidence, or, in the case of a few strong closed-source models (OpenAI et al., 2023; Anthropic, 2023), abstain from making claims.

Confident hallucinations are especially harmful. They decrease users’ trust in the errant LM and broadly make LMs unsuitable for settings where factuality is paramount such as medicine (Thirunavukarasu et al., 2023) and law (Dahl et al., 2024). Perhaps most importantly, they lead the user to *confidently make poor decisions* (Fig. 1). However, even abstentions are suboptimal, because they provide the user with no plausible claims and their likelihoods.

Linguistic calibration (Mielke et al., 2022)—conveying confidence levels in natural language that equal the likelihood that one’s claims are correct—could mitigate the harms of hallucination. If an LM was linguistically calibrated in a manner interpretable to its users, they could make good decisions regardless of the LM’s underlying knowledge.

For example, suppose a clinical LM generates a patient’s case report, providing a diagnosis. If the LM was unsure of the correct diagnosis, it could use *numerical confidence* with its corresponding claim (“I estimate that the correct diagnosis is Condition A with 60% confidence”). Then, when the doctor faces a decision—deciding the patient’s treatment—they have access to both a salient claim and an approximate likelihood of its correctness. The manner of conveying confidence is limited only by the use of language: *e.g.*, the LM could provide *linguistic* confidence statements (“I am fairly sure that the correct diagnosis is A”) or many mutually exclusive claims (“I estimate that the correct diagnosis is A with 60% confidence, though B or C are also possibilities.”). However, both classic calibration methods like temperature scaling (Guo et al., 2017) and methods for LMs (Kadavath et al., 2022; Tian et al., 2023; Kuhn et al., 2023; Mielke et al., 2022; Lin et al., 2022; Jiang et al., 2021) are restricted to classification or short outputs and hence cannot calibrate the many claims made in each long-form LM generation (cf. §5 for related work).

We make progress on this challenge by leveraging the connection between *calibration and decision theory* (Zhao et al., 2021; Zhao & Ermon, 2021). LMs fit cleanly within the framework of decision-making. Users query LMs, learn

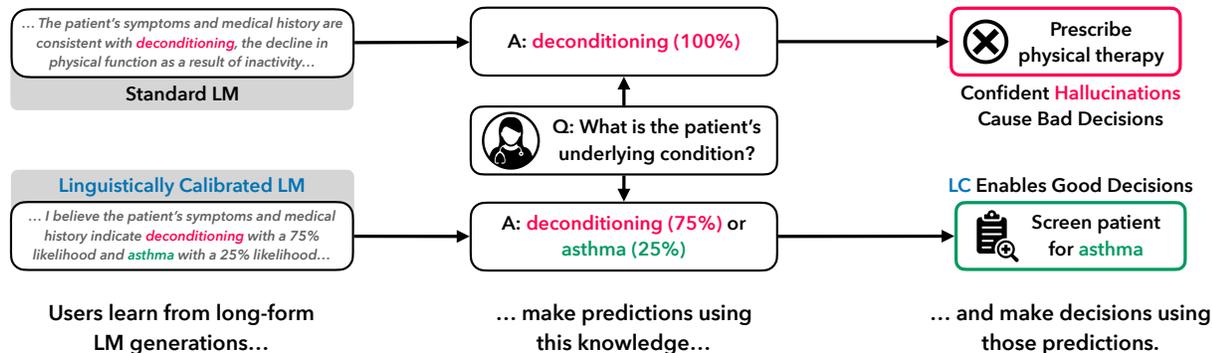


Figure 1: **Illustrative example of linguistic calibration.** We define linguistic calibration of long-form generations (LC) as calibrating an LM’s generations in a way that leads to calibrated downstream user forecasts. We apply LC to train an LM that emits calibrated statements of confidence in natural language, enabling better downstream decisions. **Left:** users read long-form generations (e.g., a doctor reading an LM-generated clinical note). **Middle:** to decide the patient’s treatment, the doctor first forecasts the patient’s underlying condition. **Upper Right:** when standard LMs lack knowledge, they hallucinate confidently, leading to a suboptimal decision (treating the wrong condition). **Lower Right:** even if the base LM cannot be confidently correct, linguistic calibration encourages the LM to spread probability over plausible claims, enabling a better decision.

from their generations, and later encounter decision-making tasks with associated questions (“What is the patient’s condition?”). They forecast an answer based on what they learned, and finally make a decision (which treatment to provide) using their forecast for which they receive some reward.

Because linguistic calibration improves downstream user decisions, we might hope to calibrate an LM by directly optimizing real-world downstream rewards. While intuitively appealing, this process is challenging; it entails tracing the knowledge learnt by users through to their decisions, and further propagating the associated rewards back to the LM.

Our contributions. We propose a definition of linguistic calibration for long-form generations (LC) which sidesteps the difficulty of tracing real-world rewards and enables training an LM to emit calibrated confidence statements in long-form generations. Our contributions are as follows:

- We define an LM emitting long-form generations to be **linguistically calibrated** if it enables users to produce *calibrated forecasts* relevant to their decision-making tasks, which in turn enable optimal decision-making.
- We instantiate this definition in a **training objective and framework** that calibrates long-form generations through a decision-theoretic lens. Our training framework first bootstraps an LM to express confidence statements with supervised finetuning. Then, it optimizes our objective using reinforcement learning (RL), rewarding the LM policy for generating text that enables calibrated forecasts on related questions.
- We linguistically calibrate Llama 2 7B using our training framework and find it significantly improves calibration versus strong baselines finetuned for factuality while matching their accuracy, in human and API-based

LLM evaluation of long-form generations. We also show that linguistic calibration has zero-shot transfer beyond the training task. Specifically, an LM calibrated using a single question-answering dataset is also calibrated on out-of-domain scientific question-answering datasets, and on an entirely held-out task of biography generation.

Instead of working in the space of text, our decision-based approach constructs an objective in the space of the predictions that users make in the process of decision-making. This makes the standard calibration machinery of proper scoring rules (Gneiting & Raftery, 2007) tractable as an objective for end-to-end calibration of long-form generations.

2 Setup

Our goal is to formulate a tractable objective that enables the end-to-end linguistic calibration of long-form LM generations. To begin, we define what it means for an LM emitting long-form generations to be linguistically calibrated.

2.1 Linguistic Calibration of Long-Form Generations

Our definition is motivated by the process of users learning from LM generations and later using their knowledge to answer questions. This **LM-assisted user forecasting** allows us to turn the problem of calibrating long-form text into the problem of calibrating probabilistic forecasts.

LM-assisted user forecasting. First, the user prompts an LM π with an open-ended query q (e.g., “Generate a clinical note for this patient...”). Next, the LM π generates a long-form context $z \sim \pi(z | q)$, e.g., the clinical note.

At some point in the future, the user encounters a question x associated with the query q , such as $x =$ “What is the

correct diagnosis of the patient?” Let $y \in \mathcal{Y}$ be the answer to the question x , and $p(y | x)$ be the ground-truth conditional distribution over answers. We note that given x , the answer y depends on neither the query q nor the generation z , *i.e.*, we have $Y \perp (Q, Z) | X$. Altogether, we have joint distribution $p(q, x, y, z) := p(q, x, y)\pi(z | q)$.

Since $p(y | x)$ is unknown to the user, they form a probabilistic forecast over possible answers y as a distribution in $\Delta^{|\mathcal{Y}|}$. In our example, the doctor can better forecast diagnoses by first reading the clinical note. In other words, the long-form generation z is salient to the question x , so the user conditions on it to make their forecast; they apply their **reader** function $f: \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$ to form a **forecast** $f(x, z) \in \Delta^{|\mathcal{Y}|}$. For example, we might have $f(x, z) = [0.2, 0.8]^\top$ placing 20% mass on Condition A and 80% on B. We denote the forecasted probability of answer y as $f(x, z)_y$.

Defining linguistic calibration for long-form generations.

LM-assisted user forecasting motivates a definition of linguistic calibration in terms of the calibration of a downstream reader; an LM is **linguistically calibrated** with respect to a user if its long-form generations z enable that user to make calibrated forecasts on question-answer pairs (x, y) using their reader function f . This notion of linguistic calibration is a property of an LM, and is defined with respect to a particular user and notion of classifier calibration.

Definition 2.1 (Linguistic Calibration of Long-Form Generations). An LM π emitting long-form generations $z \sim \pi(z | q)$ is **linguistically ϕ -calibrated** (for $\phi \in \{\text{distribution, classwise, confidence}\}$) with respect to a reader $f: \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$ if and only if f is ϕ -calibrated on the joint distribution $p(x, y, z)$.

Prior work established linguistic calibration for single-claim utterances (Mielke et al., 2022). Our definition generalizes linguistic calibration to long-form generations, which we will show has both theoretical and empirical benefits. Our focus is on calibrating long-form text, so future references to **linguistic calibration**, **linguistically calibrated**, *etc.* refer to Defn. 2.1. §5 discusses related work.

Defn. 2.1 reduces the problem of calibrating long-form text to calibrating over a smaller answer space \mathcal{Y} . Each notion of linguistic ϕ -calibration is equivalent to a traditional notion of classifier calibration ϕ applied to the reader f . We next provide examples in our notation. Readers familiar with classifier calibration could skip to §2.2.

Examples of linguistic ϕ -calibration. **Distribution calibration** is one such notion of classifier calibration (Bröcker, 2009; Kull & Flach, 2015). Intuitively, a classifier $g: \mathcal{X}' \rightarrow \Delta^{|\mathcal{Y}|}$ is distribution calibrated if its forecast $g(x') \in \Delta^{|\mathcal{Y}|}$ is faithful to the predictive uncertainty (Minderer et al., 2021): among all inputs $x' \in \mathcal{X}'$ for which the forecasted probability of class y is 0.4 (*i.e.*, $g(x')_y = 0.4$), 40% of those inputs should have label y . Taking $\mathbf{1}_y \in \{0, 1\}^{|\mathcal{Y}|}$ as the one-

hot vector for y , g is distribution calibrated if $\forall p \in \Delta^{|\mathcal{Y}|}$, $\mathbb{E}[\mathbf{1}_y | g(x') = p] = p$. Now taking our reader $f := g$ as the classifier and both the question and LM generation as the input $(x, z) := x'$, we say that an LM π is **linguistically distribution calibrated** with respect to a reader f if $\forall p \in \Delta^{|\mathcal{Y}|}$, $\mathbb{E}[\mathbf{1}_y | f(x, z) = p] = p$. Table 4 (Appendix C) provides more examples of this correspondence. These include the weaker notion of **confidence calibration** (Guo et al., 2017): among the examples with top forecasted probability equal to β , the accuracy over those examples is also β .

2.2 From Calibration to Optimal Decisions

LM-assisted user forecasting models how users learn about topics by reading LM generations and forecast answers to related questions. We now describe how they use their forecasts to inform downstream decisions. This decision-based perspective leads to our objective for linguistic calibration.

LM-assisted user decision-making. The user will make a decision by selecting an action $a \in \mathcal{A}$ (which treatment to provide the patient) according to their forecast $f(x, z) \in \Delta^{|\mathcal{Y}|}$ and loss function $L: \mathcal{A} \times \mathcal{Y} \rightarrow \mathbb{R}$. For example, the user may make a Bayes-optimal decision by choosing the action minimizing their loss under the forecast distribution: $a^* = \arg \min_a \mathbb{E}_{\hat{y} \sim f(x, z)}[L(a, \hat{y})]$. Based on the realized answer $y \sim p(y | x)$, the user suffers a loss of $L(a^*, y)$.

Ideally, one would follow this LM-assisted user decision-making process and directly train an LM to emit generations z which minimize the user’s downstream loss $L(a^*, y)$. However, it is difficult to obtain real-world rewards, and moreover to obtain a real-world distribution over LM queries and related question-answer pairs $p(q, x, y)$. We overcome these challenges by *optimizing for linguistic calibration* over a representative surrogate distribution. In other words, we encourage the LM to emit generations z that enable the user to provide calibrated answers to questions x .

In the rest of this section, we will introduce our objective for linguistic calibration and demonstrate its optimality properties. We discuss our surrogate distribution in §3.1.

Linguistic calibration implies informed decision-making.

Why is optimizing for linguistic calibration a reasonable proxy for directly optimizing downstream losses $L(a^*, y)$? Zhao et al. (2021) prove that calibrated forecasts $f(x, z)$ enable a user to make informed decisions. Specifically, a calibrated forecast enables the user to estimate the expected loss of a decision-making strategy as if they had access to the ground-truth distribution $p(y | x)$. This **accurate loss estimation** implies that Bayes-optimal decision-making outperforms other strategies. Altogether, because linguistic calibration implies calibrated forecasts (cf. Defn. 2.1), it will provide optimal decision-making guarantees.

Guarantees for weaker notions of calibration. Each

notion of linguistic ϕ -calibration (cf. Table 4) in fact corresponds to a different decision-making guarantee. Even if our optimization process cannot achieve linguistic distribution calibration, achieving linguistic ϕ -calibration with respect to a weaker notion ϕ such as confidence calibration will provide a (weaker) decision-making guarantee. Appendix C provides a proof of this correspondence (Theorem C.4).

Lastly, we note the connection between linguistic calibration and optimal decision-making motivates our evaluation metric of **reader expected calibration error** (cf. §4.1).

2.3 Training Objective for Linguistic Calibration

Because linguistic calibration is equivalent to calibration of downstream probabilistic forecasts, we can optimize for it with the standard calibration machinery of **scoring rules**. We use a **strictly proper scoring rule** as our objective, with the guarantee that it is maximized if and only if user forecasts $f(x, z)$ equal the ground-truth distribution $p(y|x)$.

Proper scoring rules. Scoring rules (Gneiting & Raftery, 2007; Savage, 1971) measure a forecast’s quality. Specifically, a scoring rule R scores a forecast $\hat{p} \in \Delta^{|\mathcal{Y}|}$ and outcome $y \in \mathcal{Y}$, i.e., $R : \Delta^{|\mathcal{Y}|} \times \mathcal{Y} \rightarrow \mathbb{R}$. **Proper** scoring rules have the desirable property that the true distribution of y , denoted by p , is a maximizer. Formally, R is proper if $\forall p, \hat{p} \in \Delta^{|\mathcal{Y}|}, \mathbb{E}_{y \sim p}[R(p, y)] \geq \mathbb{E}_{y \sim p}[R(\hat{p}, y)]$. R is **strictly proper** if the unique maximizer is the true probability p . Strictly proper scoring rules are ubiquitous in ML, e.g., the (negative) log loss or Brier score (Brier, 1950), and are natural objectives for calibration methods such as temperature (Guo et al., 2017) or Platt scaling (Platt et al., 1999).

Objective. We choose the negative log loss as our strictly proper scoring rule $R_{\text{LC}}(p, y) := \log p_y$, and therefore our objective function is $R_{\text{LC}}(f(x, z), y) := \log f(x, z)_y$. By strict propriety, if our objective is maximized, then the user’s forecast $f(x, z)$ equals the ground-truth conditional $p(y|x)$ and we obtain linguistic distribution calibration.

Proper scoring rules do not guarantee that weaker notions of calibration hold near their optima. However, it is well-established empirically that recalibrating classifiers by optimizing proper scoring rules approximately obtains weaker notions of calibration such as confidence calibration (Guo et al., 2017; Ovadia et al., 2019; Minderer et al., 2021).

3 Method

In this section, we describe our training framework for linguistically calibrating an LM. We first construct a surrogate distribution to approximate the distribution of question-answer pairs that users encounter during real-world decision-making. Then, we apply a two-step training framework (Fig. 2). First, we obtain an LM with some ability to express confidences in a long-form generation. Second, we

use it as an RL policy and optimize our proper scoring rule end-to-end, with supervision from the surrogate distribution.

3.1 Synthetic Supervision for Long-Form Calibration

In our setup (§2.1), LM-assisted user forecasting involves a tuple (q, z, x, y) , where q are user-written queries to the LM, $z \sim \pi(z|q)$ are long-form generations sampled from the LM, and (x, y) is a related question-answer pair.

Our training framework will closely follow this process and therefore requires access to a dataset of tuples (q, z, x, y) . We can synthetically generate this dataset in a manner agnostic to the downstream task, using arbitrary question-answer pairs (x, y) . In this work, we make a particular choice to use (x, y) pairs from off-the-shelf question-answering datasets.

Specifically, we first sample a question-answer pair $(x, y) \sim p(x, y)$ from a question-answering dataset, which is implicitly associated with some actual decision-making tasks. Next, we need an LM query q such that $z \sim \pi(z|q)$ is a long-form generation *salient to* (x, y) . We obtain one by converting the question x into an open-ended query $(q := \text{“Write a paragraph about } \{x\}\text{”})$ using an API-based LLM. Altogether, this gives us a tuple (q, z, x, y) where $(q, x, y) \sim p(q, x, y)$ and $z \sim \pi(z|q)$.

Next, we describe our two-step training framework.

3.2 Summary Distillation

Summary distillation (Fig. 2 Upper) bootstraps a base LM π_{Base} to have some ability to express its confidence in long-form natural language generations. We follow a simple approach inspired by Self-Consistency (Wang et al., 2023), which obtains calibrated LM confidences for short answer questions by computing a statistic of many output samples. Summary distillation generalizes this idea to longer generations, and then finetunes on our equivalent of the statistics.

First, we provide the base LM with an open-ended query q and sample many long-form responses: $\{z_i\}_{i=1}^M \sim \pi_{\text{Base}}(z|q)$. To obtain statements of confidence that are faithful to the base model’s internal confidence levels, we prompt an API-based LLM to summarize these samples into a single consensus paragraph s with statements of confidence based on the frequency of claims: $s \sim \text{Summarize}(s|z_1, \dots, z_M)$. For example, we would expect the summary shown in Fig. 2 (Upper) if 90% of the M samples answer the question with **Alt-J** and 5% with **Blur**. We perform frequency-based summarization at the *claim level*, meaning that each summary paragraph s contains multiple claims with various confidence levels and styles (e.g., numerical and linguistic).

Finally, to distill these extracted confidences back into the base model, we finetune π_{Base} on the dataset of open-ended query and summary pairs $\{(q^{(i)}, s^{(i)})\}_{i=1}^N$ to obtain the su-

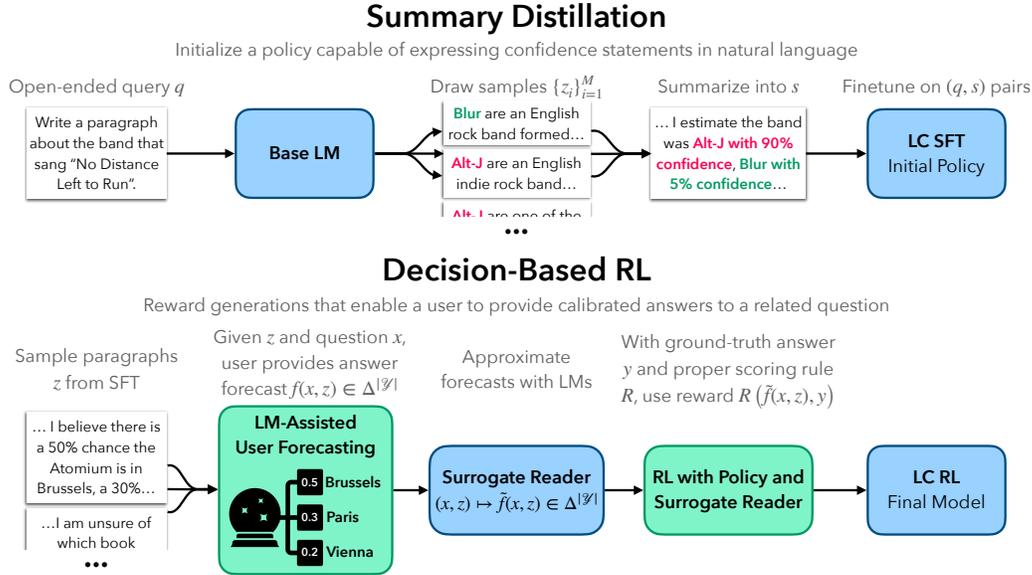


Figure 2: Our training framework for linguistic calibration of long-form generations (LC) calibrates the long-form generations of an LM by calibrating downstream user forecasts. It involves two steps: **summary distillation (Upper)** and **decision-based RL (Lower)**. Datasets are in white, LMs in blue, and steps involving user or surrogate forecasts are in green.

pervised finetuned (SFT) model π_{SFT} . π_{SFT} serves as a strong initial policy for the second RL-based step.

3.3 Decision-Based RL

Decision-based RL (Fig. 2 Lower) linguistically calibrates a policy π_{RL} (initialized at π_{SFT}) by finetuning it to emit long-form generations z that improve the calibration of the user forecast $f(x, z)$.

RL objective. We use our log likelihood proper scoring rule R_{LC} (cf. §2.3) as our reward function. Our RL objective optimizes R_{LC} over our semi-synthetic distribution:

$$\max_{\pi_{\text{RL}}} \mathbb{E}_{(q,x,y) \sim p(q,x,y), z \sim \pi_{\text{RL}}(z|q)} [R_{\text{LC}}(f(x, z), y)]. \quad (1)$$

3.4 Implementation

We next describe our instantiation of decision-based RL, which we used in our experiments. However, we note that the notion of linguistic calibration defined in §2 is agnostic to these design decisions. See Appendix D for further implementation details, including pseudocode (Algorithm 1).

Surrogate reader. For our training framework to be as scalable as possible, we would ideally avoid using a human or LLM-simulated reader f in the RL loop. We find that we can train a neural net surrogate reader $\tilde{f} : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$ which produces reasonable forecasts, because “reading” is *not a fundamentally hard task*. For example, if z provides a clear list of possible answers to the question x and associated percentage likelihoods, reading is a simple extractive task.

Using the surrogate, we optimize approximate reward $R_{\text{LC}}(\tilde{f}(x, z), y) \approx R_{\text{LC}}(f(x, z), y)$. In our evaluation, we

will test if our LM calibrated on this approximate reward generalizes to produce long-form generations z which improve simulated LLM and human forecasts $f(x, z)$.

We cannot simply train a neural net to directly predict a softmax output $f(x, z) \in \Delta^{|\mathcal{Y}|}$, because \mathcal{Y} is the vast space of all answers expressible in a finite-length string. Instead, we decompose forecasting into two operations:

1. **ExtractAnswers:** $(x, z) \mapsto \text{ExtractAnswers}(x, z) \subset \mathcal{Y}$ extracts all possible answers to the question x from the paragraph z . We implement this by finetuning a pretrained LM (RedPajama 3B, together.ai (2023)).
2. **ForecastProbs:** $(x, z, y') \mapsto [0, 1]$ assigns a probability to an answer y' to question x based on the paragraph z . We finetune π_{SFT} with a cross-entropy loss.

We define the surrogate reader’s forecast $\tilde{f}(x, z)$ as a categorical distribution with probability $\text{ForecastProbs}(x, z, y')$ on each answer $y' \in \text{ExtractAnswers}(x, z)$, and probability 0 on all others. In this particular construction, we are not guaranteed that the surrogate forecast $\tilde{f}(x, z)$ will be normalized, but in practice we find that adding a regularization term is sufficient to enforce normalization:

$$R_{\text{LC}}(\tilde{f}(x, z), y) = \log \tilde{f}(x, z)_y - \lambda \left| 1 - \sum_{y' \in \mathcal{Y}} \tilde{f}(x, z)_{y'} \right|,$$

where $\lambda > 1$ restores strict propriety (cf. D.2 for proof). Lastly, we use a standard KL penalty from π_{SFT} to mitigate over-optimization of the surrogate reader (Ouyang et al., 2022), giving us the objective (with KL coefficient β)

$$\max_{\pi_{\text{RL}}} \mathbb{E}_{(q,x,y) \sim p(q,x,y), z \sim \pi_{\text{RL}}(z|q)} \left[R_{\text{LC}}(\tilde{f}(x, z), y) - \beta \log \frac{\pi_{\text{RL}}(z|q)}{\pi_{\text{SFT}}(z|q)} \right].$$

4 Experiments

This section empirically validates our training and evaluation framework for linguistic calibration¹, demonstrating that it fulfills the following three goals:

(1) LC provides better calibration with comparable or better accuracy. We show that our linguistically calibrated LM π_{LC} emits long-form generations z which improve the calibration of user forecasts with accuracy comparable to or better than strong baselines finetuned for factuality with RL.

(2) LC is computationally tractable. We show that π_{LC} —which avoids obtaining many costly human forecasts by training with cheap surrogates—improves the calibration of *human* forecasts at test time. We also develop an automated framework to evaluate linguistic calibration with simulated forecasts and validate its agreement with crowdworkers.

(3) LC generalizes well out-of-distribution. We demonstrate that the improvement in forecast calibration due to adopting LM π_{LC} generalizes under significant domain shift to scientific question-answer distributions $p_{\text{OOD}}(x, y)$. We also evaluate π_{LC} on an entirely held-out task of person biography generation without any re-training, finding that π_{LC} produces calibrated claims throughout the long-form generation z according to a fine-grained simulated evaluation.

4.1 Setup

We use our training framework to linguistically calibrate Llama 2 7B, sourcing question-answer (QA) pairs (x, y) (cf. §3.1) from TriviaQA (Joshi et al., 2017). We emphasize that our LMs produce *long-form generations* z on the question’s topic, unlike prior works which calibrate models that predict a class distribution or short answer (cf. §5). Further details on the training framework can be found in Appendix D.

Question-answering evaluation framework. Following our generative process during training (§3.1), we use off-the-shelf QA datasets as a proxy for questions encountered during real-world decision-making, and evaluate the linguistic calibration of generations z through the performance of downstream forecasts. Specifically, for a held-out QA pair (x, y) , we convert x into an open-ended query q , sample a long-form generation $z \sim \pi(z | q)$ from various LMs π , and evaluate the calibration and accuracy of forecast $f(x, z)$.

Naturally, this framework depends on which users are providing forecasts and how, *i.e.*, the choice of reader f . We are interested in the case where users strongly rely on the knowledge of the LM. Therefore, we include instructions to the user (either simulated or human) to ignore their background knowledge about the correct answer when providing a forecast (cf. Appendix E for further evaluation details).

¹We release code at github.com/tatsu-lab/linguistic_calibration.

Reader expected calibration error. We measure the calibration of the reader f over the joint distribution $p(x, y, z)$ with the standard **expected calibration error (ECE)** metric (Guo et al., 2017). Intuitively, **reader ECE** is a proxy for decision-making performance through the equivalence of confidence calibration and optimal decision-making (cf. Theorem C.4 in Appendix C). For a formal statement of reader ECE and implementation details, see Appendix E.

Baselines. In our main evaluations, we compare LC RL (π_{LC}) with both *non-confidence* and *confidence* baselines, all derived from Llama 2 7B. We provide a strong data-matched comparison to LC by finetuning directly for factuality using RL. This baseline is similar to Tian et al. (2024), but instead of using self-supervised or automated factuality scores as the RL reward, we use correctness determined with ground-truth question-answer pairs from TriviaQA. In-context learning (ICL) baselines use TriviaQA examples from a prompt development split, and SFT/RL baselines use the same splits as π_{SFT} and π_{LC} . Each example is a (q, x, y) tuple, where q is an open-ended query obtained from question x (cf. §3.1).

We include the following non-confidence baselines:

- **ICL.** We randomly sample 8 open-ended queries, generate long-form responses with GPT-4, manually fact-check those responses using Wikipedia, and use fact-checked (query, response) pairs as Llama 2 7B ICL examples.
- **Claude Distill.** We generate long-form responses with Claude 2 over all queries in the SFT split, and finetune Llama 2 7B on these (query, response) pairs.
- **Factuality SFT.** We use the above ICL baseline to generate long-form responses over all queries in the SFT split, and finetune Llama 2 7B on these (query, response) pairs. We found Factuality SFT to outperform Claude Distill on a TriviaQA validation split, so we use it as the starting point for the following baseline, Factuality RL.
- **Factuality RL.** To provide a strong RL baseline, we train a reward model to score the correctness of long-form outputs. Our approach to obtain this baseline is analogous to the decision-based RL algorithm (Algorithm 1), except instead of training a surrogate reader, we train a single reward model that, given a generation z and QA pair (x, y) , predicts a binary indicator whether z provides the correct answer to x . This serves as the RL reward. We use Factuality SFT as the initial policy for PPO.

When training our confidence methods, we use the ICL baseline above to generate the responses which are summarized in summary distillation. Our confidence baselines include the **LC SFT** model (π_{SFT}) and the following baseline:

- **Summary ICL.** We use the summary distillation algorithm (§3.2) on 8 queries sampled from the prompt development split to produce 8 Claude 2 summaries $\{s^{(i)}\}_{i=1}^8$, which we use in (query, summary) ICL examples.

Other baselines including GPT-4. Appendix B includes results for several other methods: Llama 2 Chat (which underperformed Factuality SFT), the oracle baseline of directly evaluating summaries s , and GPT-4-based methods including 0-Shot, ICL 8-Shot, asking for confidence statements 0-shot, and Summary ICL 8-Shot. Unsurprisingly, GPT-4-based methods are far more factual than all Llama 2 7B-based methods. However, we find that LC RL has reader ECE comparable to GPT-4 baselines (cf. Figs. 7, 9, 11, 13, 15), despite significantly worse factuality. This demonstrates that even small LLMs with relatively weak factuality can be well-calibrated with the right objective.

4.2 Linguistic Calibration using QA Datasets

To begin, we evaluate our methods using held-out (x, y) pairs from the TriviaQA, Jeopardy (Kaggle, 2020), SciQ (Welbl et al., 2017), and BioASQ Task B (Krithara et al., 2023) question-answering datasets. The latter two datasets are a significant domain shift; SciQ consists of crowd-sourced physics, chemistry, and biology exam questions, and BioASQ Task B consists of biomedical questions annotated by experts (cf. Appendix E for dataset details).

These QA evaluations validate two of our three goals: we find that LC improves calibration with comparable or better accuracy, and that our training and evaluation framework are computationally tractable. Our strong results on the Jeopardy, SciQ, and BioASQ datasets also partially validate our final goal of strong out-of-distribution generalization. We report TriviaQA and SciQ results here, and refer the reader to Appendix B for results with all datasets and baselines.

Better ECE with comparable or better accuracy in long-form generation. Our main result in Fig. 3 is that LC RL has significantly better ECE than non-confidence baselines, including Factuality RL, while matching or exceeding their accuracy. This result holds in-distribution on TriviaQA with both simulated (Fig. 3a) and human (Fig. 3b) readers, and out-of-distribution (OOD) on SciQ (Fig. 3c), Jeopardy (Fig. 9), and BioASQ (Fig. 13) with a simulated reader, demonstrating that LC generalizes under domain shift.

Our results also support the effectiveness of decision-based RL. LC RL significantly improves over LC SFT in both ECE and accuracy, with a greater absolute improvement in ECE/accuracy than Factuality SFT to Factuality RL. This supports our claim that optimizing proper scoring rules of downstream forecasts is an effective way to induce calibration in long-form generations.

Reliability diagrams demonstrate meaningful confidences. A natural question is whether the confidences learned by LC models are meaningful. For example, if all of a model’s confidences collapsed to its average accuracy, it would obtain perfect ECE despite having confidences that are useless for tasks such as conveying the likelihood

of hallucination. For a more fine-grained assessment of our model’s calibration, we use **reliability diagrams** (DeGroot & Fienberg, 1983; Niculescu-Mizil & Caruana, 2005), which visualize the average confidence and accuracy of each ECE bin. The plot of the pathological model above would be a single point. A perfectly calibrated model with meaningful confidences would have an identity plot $y = x$. In Fig. 4, we observe that LC model confidences are both meaningful, covering a wide range of confidence values, and consistently close to the identity across confidence values. This validates that LC is effective in linguistically conveying the likelihood of hallucination in a long-form generation.

4.3 Zero-Shot Transfer to Biography Generation

The QA evaluation validated two of our three goals: (1) LC RL pareto-dominates baselines on the accuracy-ECE frontier. Its significant improvement over LC SFT validates the effectiveness of decision-based RL. (2) We demonstrated the computational efficiency of our training and evaluation framework, because LC RL is trained with cheap surrogates but performs well in evaluations with human readers, and our simulated readers have high agreement with human readers (see Appendix E.3 for full forecast agreement statistics). Lastly, the QA evaluation partially validates (3) OOD generalization: LC RL performs well on the Jeopardy, SciQ, and BioASQ datasets with simulated readers.

To conclusively validate this final goal, we evaluate LC on a significant distribution shift in the task. Our models were trained to perform long-form generation about trivia-style questions, and we now test their ability to write factual biographies on people sampled from Wikipedia. Specifically, we source 500 people from the unlabeled split of FactScore (Min et al., 2023) and use prompt “Write a paragraph bio about {person}”. We emphasize that *our models were not trained on biography generation*.

FactScore-based metric. We also use a more fine-grained simulated evaluation than the QA tasks, testing the accuracy and calibration of generated biographies z at the per-claim level. Specifically, we split generations into a list of claims, filter out subjective claims, and then compute accuracy and ECE over all claims pooled across biographies, following FactScore (Min et al., 2023). We use Claude 2 for splitting and filtering, and an identical fact checking pipeline to Min et al. other than using Claude 2 instead of ChatGPT for fact-checking conditioned on retrieved Wikipedia context paragraphs. To compute ECE, we need to assign confidence values to each claim. For numerical uncertainties such as percentages, this is a simple extractive task which API-based LLMs perform well. For linguistic uncertainties, we provide Claude 2 with a short list of mappings between linguistic phrases and consensus probabilities collected in a human study (Wallsten, 1990), and allow the LLM to generalize from this to assign probabilities for other linguistic phrases

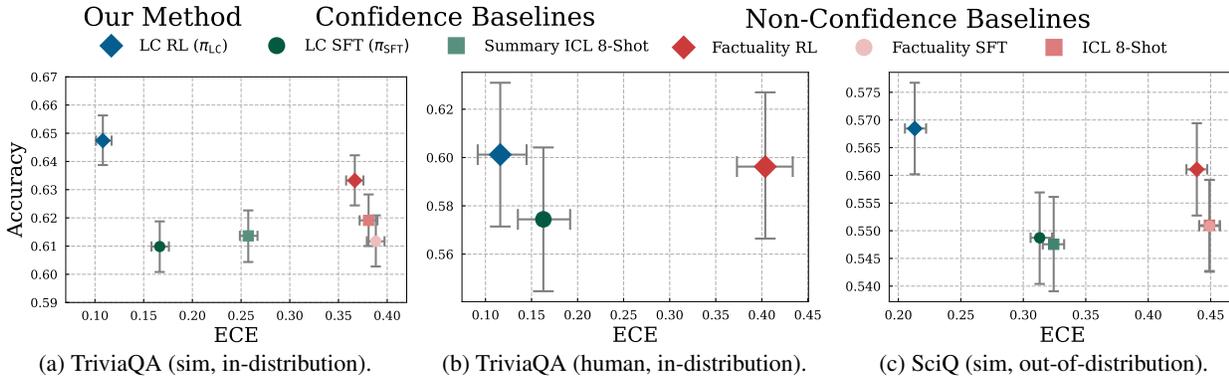


Figure 3: **Accuracy-ECE Frontier for Question-Answering** (upper left is better), simulated (sim) and human readers. LC RL Pareto-dominates Factuality RL and SFT, with significantly better ECE while matching or exceeding their accuracy.

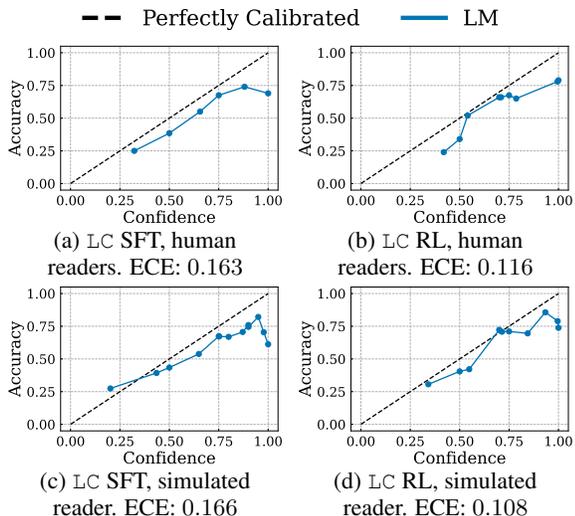


Figure 4: **TriviaQA Reliability Diagrams**. LC models display a wide range of confidences and good calibration in their long-form generations, with LC RL improving calibration further. Human and simulated results closely match.

not present in the mapping (cf. Appendix E.2 for details).

LC generalizes to biography generation and claim-level evaluation. Both LC methods demonstrate significant ECE and accuracy gains compared to non-confidence baselines, generalizing well to an entirely held-out task (Fig. 5). Because we here compute ECE at the claim level and the LC methods obtain reasonable ECEs and reliability diagrams (Fig. 6), we confirm that they incorporate calibrated confidence statements *throughout their long-form generations*. Additionally, decision-based RL significantly improves accuracy and ECE over LC SFT even under significant task shift, further validating our linguistic calibration objective.

Lastly, we note the surprising finding that LC SFT improves in accuracy compared to Factuality RL. We attribute this to the tendency of LC models to generate a higher proportion of less “precise” claims which are still objective and correct

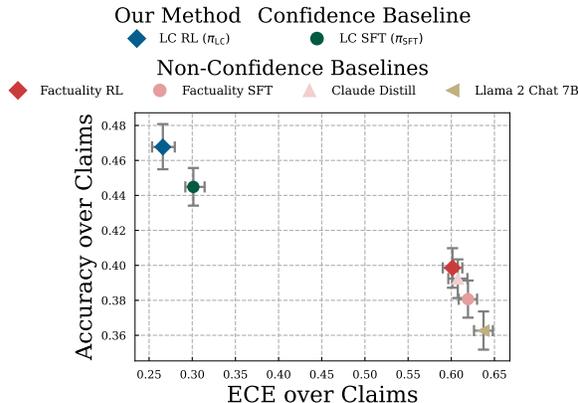


Figure 5: **Accuracy-ECE Frontier for Person Biography Generation**, simulated reader (upper left is better). Accuracy and ECE are computed at the claim level, using a finetuned retrieval model and API-based LLM for fact checking, following Min et al. (2023). LC RL significantly outperforms non-confidence methods in accuracy and ECE.

and therefore count towards the accuracy metric. Qualitative examples on all datasets are provided in Appendix B.13.

Altogether, our strong results in a challenging distribution shift setting validate our final goal.

5 Related Work

Calibration. The calibration of probabilistic forecasts is an extensively studied topic (Brier, 1950; Savage, 1971; Murphy, 1973; Dawid, 1984; Zdrozny & Elkan, 2001; Hebert-Johnson et al., 2018; Kull & Flach, 2015). In particular, isotonic regression (Niculescu-Mizil & Caruana, 2005), Platt scaling (Platt et al., 1999), temperature scaling (Guo et al., 2017), and histogram binning (Kumar et al., 2019) are effective approaches for improving the calibration of probability estimates. Other methods improve calibration through ensembling (Lakshminarayanan et al., 2017) and

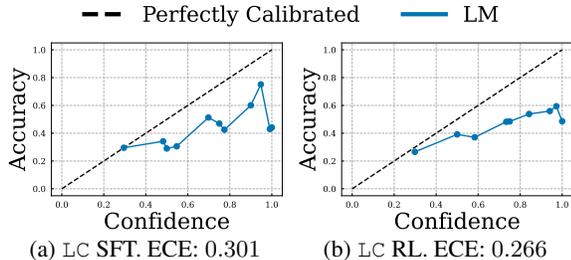


Figure 6: **Person Biography Generation Reliability Diagrams** for LC SFT (Left) and LC RL (Right), simulated reader. LC methods demonstrate non-trivial calibration in an entirely out-of-distribution task which requires calibrated statements of confidence throughout the paragraph.

Bayesian model averaging using an approximate model posterior (Blundell et al., 2015; Band et al., 2021; Malinin et al., 2021; Nado et al., 2022; Tran et al., 2022).

Recent works studied the probabilistic calibration of LMs, showing that LMs can be well-calibrated (Kadavath et al., 2022). LMs that went through RL from human feedback (RLHF) tend to be ill-calibrated on multiple-choice questions (OpenAI et al., 2023), but temperature scaling fixes the issue. Chain-of-thought prompting leads to ill-calibration (Bai et al., 2022). The present focus is the linguistic calibration of LMs that produce *long-form text*, a setting to which established recalibration methods do not apply. This is because long-form generations may contain many claims, each with confidence expressed in language.

LMs producing uncertainty. The literature has studied methods to let LMs produce confidence scores. Works have evaluated LM uncertainty estimation on multiple choice (Ye et al., 2024) and studied how sampling variance can estimate uncertainty (Malinin & Gales, 2021; Kuhn et al., 2023; Wang et al., 2022). Our sampling-based Summary ICL and Direct Summary Eval baselines (Appendix B) avoid logit access, extending the black-box confidence elicitation framework in Xiong et al. (2024) to long-form text.

Other works analyze how LMs express linguistic uncertainty (Mielke et al., 2022; Zhou et al., 2023). Considering single-claim utterances, Mielke et al. (2022) define an LM to be linguistically calibrated if it emits a verbalized confidence statement matching the likelihood of its response’s correctness. However, the long-form, multi-claim generations that users encounter in practice have neither a single closed-form confidence nor a correctness; each generation contains information that answers many possible downstream questions. Defn. 2.1 recovers the notion of linguistic calibration in Mielke et al. when specialized to single-claim outputs z .²

²Let $q := x$, and suppose the LM generates a single-claim utterance $z \sim \pi(z | q)$ with a confidence statement and answer from \mathcal{Y} , e.g., $z = \text{“I’m not sure, but my guess is Los Angeles.”}$ Then Mielke et al. (2022) considers *linguistic confidence calibration*.

A related line of work enables LMs to *directly* express uncertainty, focusing on short utterances (Mielke et al., 2022; Xiong et al., 2024), arithmetic problems (Lin et al., 2022), and QA (Jiang et al., 2021; Tian et al., 2023; Shrivastava et al., 2023). All these prior works consider settings where the set of responses is a small closed set and the notion of calibration is well-defined. Other works finetune LMs to abstain (Cheng et al., 2024) or output templated uncertainty phrases (Yang et al., 2023) on QA tasks. Lastly, concurrent work (Huang et al., 2024) evaluates methods such as self-consistency (Wang et al., 2022) and supervised finetuning in calibrating long-form generations. To the best of our knowledge, our method is the first to simultaneously provide calibrated *text-based* statements of confidence, which are important for interpretability (Mielke et al., 2022), while working in the setting of *long-form, multi-claim* generations. We overcome the challenge of defining calibration in this setting by drawing connections between decision-making and uncertainty quantification, enabling us to build a *single end-to-end objective* that calibrates long-form generations.

We discuss additional related work in calibration, decision-making, and LM finetuning in Appendix A.

6 Discussion

Limitations and future work. Our linguistically calibrated LM generalizes well from surrogate to human forecasts. However, many of the confidence statements it emits are fairly unambiguous, e.g., percentages. Therefore, future work could investigate how closely LM and human interpretations of ambiguous linguistic confidence statements match, which could enable training LMs with linguistic confidence statements that are tailored to user populations. Additionally, we use off-the-shelf question-answering datasets as a proxy for questions encountered during real-world decision-making. To improve LC’s generalization to decision-making in-the-wild, future work could curate a more representative QA dataset. Lastly, we work in a white-box setting where finetuning is possible; our training framework could not calibrate API-based LLMs with access only to completions.

Conclusions. We defined linguistic calibration of long-form generations: calibrating the long-form generations of an LM in a way that leads to calibrated probabilistic forecasts by its downstream users. By constructing an objective in the space of these forecasts, we were able to apply the standard calibration machinery of proper scoring rules for end-to-end linguistic calibration. Instantiating this objective in a training framework and linguistically calibrating Llama 2 7B enables it to emit calibrated confidence statements, significantly improving the calibration of downstream human and simulated forecasts while matching or exceeding strong RL-tuned baselines in accuracy.

Acknowledgements

We thank Michael Y. Li, Yann Dubois, Yu Sun, Zitong Yang, and members of the Tatsu Lab, Ma Lab, Stanford Machine Learning Group, and Stanford NLP Group for their helpful feedback. This work was supported by Open Philanthropy, IBM, NSF IIS 2211780, the Stanford HAI–Google Cloud Credits Program, and the Anthropic Researcher Access Program. NB acknowledges funding from an NSF Graduate Research Fellowship and a Quad Fellowship. XL acknowledges funding from a Stanford Graduate Fellowship and a Meta PhD Fellowship.

Impact Statement

The ideas and methods introduced in this work generally improve the linguistic calibration of LMs to benefit users in downstream decision-making. Thus, we believe this work will likely positively impact applied areas of AI where an LM copilot is adopted.

On the other hand, while we have demonstrated improvements in calibration with our methods, we acknowledge that our LMs’ expressions of confidence can still differ from the true likelihoods of its claims being correct. It is thus important to note that even linguistically calibrated LMs should still be treated as tools to aid decision-making instead of replacements for human judgment and expertise.

References

- Akyürek, A. F., Akyürek, E., Choshen, L., Wijaya, D., and Andreas, J. Deductive closure training of language models for coherence, accuracy, and updatability, 2024.
- Anthropic. Model card and evaluations for claude models, 2023.
- Askill, A., Bai, Y., Chen, A., Drain, D., Ganguli, D., Henighan, T., Jones, A., Joseph, N., Mann, B., Das-Sarma, N., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Kernion, J., Ndousse, K., Olsson, C., Amodei, D., Brown, T., Clark, J., McCandlish, S., Olah, C., and Kaplan, J. A general language assistant as a laboratory for alignment, 2021.
- Bai, Y., Kadavath, S., Kundu, S., Askill, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., et al. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*, 2022.
- Band, N., Rudner, T. G. J., Feng, Q., Filos, A., Nado, Z., Dusenberry, M. W., Jerfel, G., Tran, D., and Gal, Y. Benchmarking bayesian deep learning on diabetic retinopathy detection tasks. In *NeurIPS Datasets and Benchmarks Track*, 2021.
- Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. Weight Uncertainty in Neural Networks. In Bach, F. and Blei, D. (eds.), *PMLR*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1613–1622, Lille, France, 07–09 Jul 2015. PMLR.
- Boyd, S. and Vandenberghe, L. Convex optimization. *Cambridge University Press*, 2004.
- Brier, G. W. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950. doi: 10.1175/1520-0493(1950)078<0001:VOFEIT>2.0.CO;2. URL https://journals.ametsoc.org/view/journals/mwre/78/1/1520-0493_1950_078_0001_vofeit_2_0_co_2.xml.
- Bröcker, J. Reliability, sufficiency, and the decomposition of proper scores. *Quarterly Journal of the Royal Meteorological Society*, 135(643):1512–1519, 2009. doi: <https://doi.org/10.1002/qj.456>. URL <https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/qj.456>.
- Cheng, Q., Sun, T., Liu, X., Zhang, W., Yin, Z., Li, S., Li, L., He, Z., Chen, K., and Qiu, X. Can ai assistants know what they don’t know?, 2024.
- Cover, T. M. and Thomas, J. A. *Elements of Information Theory*. Wiley, New York, 1991.
- Cresswell, J. C., Sui, Y., Kumar, B., and Vouitsis, N. Conformal prediction sets improve human decision making, 2024.
- Cui, G., Yuan, L., Ding, N., Yao, G., Zhu, W., Ni, Y., Xie, G., Liu, Z., and Sun, M. Ultrafeedback: Boosting language models with high-quality feedback, 2023.
- Dahl, M., Magesh, V., Suzgun, M., and Ho, D. E. Large legal fictions: Profiling legal hallucinations in large language models, 2024.
- Dao, T. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.
- Dao, T., Fu, D. Y., Ermon, S., Rudra, A., and Ré, C. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022.
- Dawid, A. P. Present position and potential developments: Some personal views statistical theory the prequential approach. *Journal of the Royal Statistical Society: Series A (General)*, 147(2):278–290, 1984.
- DeGroot, M. H. and Fienberg, S. E. The comparison and evaluation of forecasters, 1983.
- Dettmers, T., Lewis, M., Shleifer, S., and Zettlemoyer, L. 8-bit optimizers via block-wise quantization, 2022.

- Dubois, Y., Li, X., Taori, R., Zhang, T., Gulrajani, I., Ba, J., Guestrin, C., Liang, P., and Hashimoto, T. AlpacaFarm: A simulation framework for methods that learn from human feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL <https://openreview.net/forum?id=4hturzLcKX>.
- Evans, O., Cotton-Barratt, O., Finnveden, L., Bales, A., Balwit, A., Wills, P., Righetti, L., and Saunders, W. Truthful ai: Developing and governing ai that does not lie, 2021.
- Foster, D. P. and Vohra, R. Regret in the on-line decision problem. *Games and Economic Behavior*, 29(1):7–35, 1999. ISSN 0899-8256. doi: <https://doi.org/10.1006/game.1999.0740>. URL <https://www.sciencedirect.com/science/article/pii/S0899825699907406>.
- Foster, D. P. and Vohra, R. V. Asymptotic calibration. *Biometrika*, 85(2):379–390, 1998. ISSN 00063444. URL <http://www.jstor.org/stable/2337364>.
- Gneiting, T. and Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007. doi: [10.1198/016214506000001437](https://doi.org/10.1198/016214506000001437). URL <https://doi.org/10.1198/016214506000001437>.
- Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1321–1330. PMLR, 06–11 Aug 2017. URL <https://proceedings.mlr.press/v70/guo17a.html>.
- Hebert-Johnson, U., Kim, M., Reingold, O., and Rothblum, G. Multicalibration: Calibration for the (Computationally-identifiable) masses. In Dy, J. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1939–1948. PMLR, 10–15 Jul 2018. URL <https://proceedings.mlr.press/v80/hebert-johnson18a.html>.
- Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., and Liu, T. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions, 2023.
- Huang, Y., Liu, Y., Thirukovalluru, R., Cohan, A., and Dhingra, B. Calibrating long-form generations from large language models, 2024.
- Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12), mar 2023. ISSN 0360-0300. doi: [10.1145/3571730](https://doi.org/10.1145/3571730). URL <https://doi.org/10.1145/3571730>.
- Jiang, Z., Araki, J., Ding, H., and Neubig, G. How Can We Know When Language Models Know? On the Calibration of Language Models for Question Answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 09 2021. ISSN 2307-387X. doi: [10.1162/tacl.a.00407](https://doi.org/10.1162/tacl.a.00407). URL <https://doi.org/10.1162/tacl.a.00407>.
- Joshi, M., Choi, E., Weld, D. S., and Zettlemoyer, L. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017. Association for Computational Linguistics.
- 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.
- Kaggle. 200,000+ jeopardy! questions, 2020. URL <https://www.kaggle.com/datasets/tunguz/200000-jeopardy-questions/data>.
- Krithara, A., Nentidis, A., Bougiatiotis, K., and Paliouras, G. Bioasq-qa: A manually curated corpus for biomedical question answering. *Scientific Data*, 10(1):170, Mar 2023. ISSN 2052-4463. doi: [10.1038/s41597-023-02068-4](https://doi.org/10.1038/s41597-023-02068-4). URL <https://doi.org/10.1038/s41597-023-02068-4>.
- Kuhn, L., Gal, Y., and Farquhar, S. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation, 2023.
- Kull, M. and Flach, P. Novel decompositions of proper scoring rules for classification: Score adjustment as precursor to calibration. In Appice, A., Rodrigues, P. P., Santos Costa, V., Soares, C., Gama, J., and Jorge, A. (eds.), *Machine Learning and Knowledge Discovery in Databases*, pp. 68–85, Cham, 2015. Springer International Publishing. ISBN 978-3-319-23528-8.
- 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. In Wallach, H., Larochelle,

- H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/8ca01ea920679a0fe3728441494041b9-Paper.pdf.
- Kumar, A., Liang, P. S., and Ma, T. Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, 32, 2019.
- Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pp. 6402–6413, 2017.
- Lhoest, Q., Villanova del Moral, A., Jernite, Y., Thakur, A., von Platen, P., Patil, S., Chaumond, J., Drame, M., Plu, J., Tunstall, L., Davison, J., Šaško, M., Chhablani, G., Malik, B., Brandeis, S., Le Scao, T., Sanh, V., Xu, C., Patry, N., McMillan-Major, A., Schmid, P., Gugger, S., Delangue, C., Matussière, T., Debut, L., Bekman, S., Cistac, P., Goehringer, T., Mustar, V., Lagunas, F., Rush, A., and Wolf, T. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 175–184, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL <https://aclanthology.org/2021.emnlp-demo.21>.
- Liang, P., Bommasani, R., Lee, T., Tsipras, D., Soylu, D., Yasunaga, M., Zhang, Y., Narayanan, D., Wu, Y., Kumar, A., Newman, B., Yuan, B., Yan, B., Zhang, C., Cosgrove, C., Manning, C. D., Ré, C., Acosta-Navas, D., Hudson, D. A., Zelikman, E., Durmus, E., Ladhak, F., Rong, F., Ren, H., Yao, H., Wang, J., Santhanam, K., Orr, L., Zheng, L., Yuksekgonul, M., Suzgun, M., Kim, N., Guha, N., Chatterji, N., Khattab, O., Henderson, P., Huang, Q., Chi, R., Xie, S. M., Santurkar, S., Ganguli, S., Hashimoto, T., Icard, T., Zhang, T., Chaudhary, V., Wang, W., Li, X., Mai, Y., Zhang, Y., and Koreeda, Y. Holistic evaluation of language models, 2023.
- Lin, S., Hilton, J., and Evans, O. Teaching models to express their uncertainty in words, 2022.
- Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.
- Malinin, A. and Gales, M. Uncertainty estimation in autoregressive structured prediction. In *International Conference on Learning Representations*, 2021. URL <https://openreview.net/forum?id=jN5y-zb5Q7m>.
- Malinin, A., Band, N., Gal, Y., Gales, M., Ganshin, A., Chesnokov, G., Noskov, A., Ploskonosov, A., Prokhorenkova, L., Provilkov, I., Raina, V., Raina, V., Roginskiy, D., Shmatova, M., Tigas, P., and Yangel, B. Shifts: A Dataset of Real Distributional Shift Across Multiple Large-Scale Tasks. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.
- Mielke, S. J., Szlam, A., Dinan, E., and Boureau, Y.-L. Reducing conversational agents’ overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872, 2022. doi: 10.1162/tacl.a.00494. URL <https://aclanthology.org/2022.tacl-1.50>.
- Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W.-t., Koh, P., Iyyer, M., Zettlemoyer, L., and Hajishirzi, H. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.741. URL <https://aclanthology.org/2023.emnlp-main.741>.
- Minderer, M., Djolonga, J., Romijnders, R., Hubis, F., Zhai, X., Houlsby, N., Tran, D., and Lucic, M. Revisiting the calibration of modern neural networks. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 15682–15694. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/8420d359404024567b5aefda1231af24-Paper.pdf.
- Murphy, A. H. A new vector partition of the probability score. *Journal of Applied Meteorology and Climatology*, 12(4):595–600, 1973.
- Nado, Z., Band, N., Collier, M., Djolonga, J., Dusenberry, M. W., Farquhar, S., Feng, Q., Filos, A., Havasi, M., Jenatton, R., Jerfel, G., Liu, J., Mariet, Z., Nixon, J., Padhy, S., Ren, J., Rudner, T. G. J., Sbahi, F., Wen, Y., Wenzel, F., Murphy, K., Sculley, D., Lakshminarayanan, B., Snoek, J., Gal, Y., and Tran, D. Uncertainty baselines: Benchmarks for uncertainty & robustness in deep learning, 2022.

- Niculescu-Mizil, A. and Caruana, R. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd International Conference on Machine Learning, ICML '05*, pp. 625–632, New York, NY, USA, 2005. Association for Computing Machinery. ISBN 1595931805. doi: 10.1145/1102351.1102430. URL <https://doi.org/10.1145/1102351.1102430>.
- OpenAI, Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., Avila, R., Babuschkin, I., Balaji, S., Balcom, V., Baltescu, P., Bao, H., Bavarian, M., Belgum, J., Bello, I., Berdine, J., Bernadett-Shapiro, G., Berner, C., Bogdonoff, L., Boiko, O., Boyd, M., Brakman, A.-L., Brockman, G., Brooks, T., Brundage, M., Button, K., Cai, T., Campbell, R., Cann, A., Carey, B., Carlson, C., Carmichael, R., Chan, B., Chang, C., Chantzis, F., Chen, D., Chen, S., Chen, R., Chen, J., Chen, M., Chess, B., Cho, C., Chu, C., Chung, H. W., Cummings, D., Currier, J., Dai, Y., Decareaux, C., Degry, T., Deutsch, N., Deville, D., Dhar, A., Dohan, D., Dowling, S., Dunning, S., Ecoffet, A., Eleti, A., Eloundou, T., Farhi, D., Fedus, L., Felix, N., Fishman, S. P., Forte, J., Fulford, I., Gao, L., Georges, E., Gibson, C., Goel, V., Gogineni, T., Goh, G., Gontijo-Lopes, R., Gordon, J., Grafstein, M., Gray, S., Greene, R., Gross, J., Gu, S. S., Guo, Y., Hallacy, C., Han, J., Harris, J., He, Y., Heaton, M., Heidecke, J., Hesse, C., Hickey, A., Hickey, W., Hoeschele, P., Houghton, B., Hsu, K., Hu, S., Hu, X., Huizinga, J., Jain, S., Jain, S., Jang, J., Jiang, A., Jiang, R., Jin, H., Jin, D., Jomoto, S., Jonn, B., Jun, H., Kaftan, T., Łukasz Kaiser, Kamali, A., Kanitscheider, I., Keskar, N. S., Khan, T., Kilpatrick, L., Kim, J. W., Kim, C., Kim, Y., Kirchner, H., Kiros, J., Knight, M., Kokotajlo, D., Łukasz Kondraciuk, Kondrich, A., Konstantinidis, A., Kopic, K., Krueger, G., Kuo, V., Lampe, M., Lan, I., Lee, T., Leike, J., Leung, J., Levy, D., Li, C. M., Lim, R., Lin, M., Lin, S., Litwin, M., Lopez, T., Lowe, R., Lue, P., Makanju, A., Malfacini, K., Manning, S., Markov, T., Markovski, Y., Martin, B., Mayer, K., Mayne, A., McGrew, B., McKinney, S. M., McLeavey, C., McMillan, P., McNeil, J., Medina, D., Mehta, A., Menick, J., Metz, L., Mishchenko, A., Mishkin, P., Monaco, V., Morikawa, E., Mossing, D., Mu, T., Murati, M., Murk, O., Mély, D., Nair, A., Nakano, R., Nayak, R., Neelakantan, A., Ngo, R., Noh, H., Ouyang, L., O’Keefe, C., Pachocki, J., Paino, A., Palermo, J., Pantuliano, A., Parascandolo, G., Parish, J., Parparita, E., Passos, A., Pavlov, M., Peng, A., Perelman, A., de Avila Belbute Peres, F., Petrov, M., de Oliveira Pinto, H. P., Michael, Pokornyy, Pokrass, M., Pong, V., Powell, T., Power, A., Power, B., Proehl, E., Puri, R., Radford, A., Rae, J., Ramesh, A., Raymond, C., Real, F., Rimbach, K., Ross, C., Rotsted, B., Roussez, H., Ryder, N., Saltarelli, M., Sanders, T., Santurkar, S., Sastry, G., Schmidt, H., Schnurr, D., Schulman, J., Selman, D., Sheppard, K., Sherbakov, T., Shieh, J., Shoker, S., Shyam, P., Sidor, S., Sigler, E., Simens, M., Sitkin, J., Slama, K., Sohl, I., Sokolowsky, B., Song, Y., Staudacher, N., Such, F. P., Summers, N., Sutskever, I., Tang, J., Tezak, N., Thompson, M., Tillet, P., Tootoonchian, A., Tseng, E., Tuggle, P., Turley, N., Tworek, J., Uribe, J. F. C., Vallone, A., Vijayvergiya, A., Voss, C., Wainwright, C., Wang, J. J., Wang, A., Wang, B., Ward, J., Wei, J., Weinmann, C., Welihinda, A., Welinder, P., Weng, J., Weng, L., Wiethoff, M., Willner, D., Winter, C., Wolrich, S., Wong, H., Workman, L., Wu, S., Wu, J., Wu, M., Xiao, K., Xu, T., Yoo, S., Yu, K., Yuan, Q., Zaremba, W., Zellers, R., Zhang, C., Zhang, M., Zhao, S., Zheng, T., Zhuang, J., Zhuk, W., and Zoph, B. Gpt-4 technical report, 2023.
- Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., and Lowe, R. Training language models to follow instructions with human feedback, 2022. URL <https://arxiv.org/abs/2203.02155>.
- Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., and Snoek, J. Can You Trust Your Model’s Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift. In *Advances in Neural Information Processing Systems* 32. 2019.
- Park, P. S., Goldstein, S., O’Gara, A., Chen, M., and Hendrycks, D. Ai deception: A survey of examples, risks, and potential solutions, 2023.
- 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.
- Savage, L. J. Elicitation of personal probabilities and expectations. *Journal of the American Statistical Association*, 66(336):783–801, 1971. ISSN 01621459. URL <http://www.jstor.org/stable/2284229>.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O. Proximal policy optimization algorithms, 2017.
- Shrivastava, V., Liang, P., and Kumar, A. Llamas know what gpts don’t show: Surrogate models for confidence estimation, 2023.
- Thirunavukarasu, A. J., Ting, D. S. J., Elangovan, K., Gutierrez, L., Tan, T. F., and Ting, D. S. W. Large language models in medicine. *Nature Medicine*, 29(8):1930–1940, Aug 2023. ISSN 1546-170X. doi:

- 10.1038/s41591-023-02448-8. URL <https://doi.org/10.1038/s41591-023-02448-8>.
- Tian, K., Mitchell, E., Zhou, A., Sharma, A., Rafailov, R., Yao, H., Finn, C., and Manning, C. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.330. URL <https://aclanthology.org/2023.emnlp-main.330>.
- Tian, K., Mitchell, E., Yao, H., Manning, C. D., and Finn, C. Fine-tuning language models for factuality. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=WPZ2yPag4K>.
- together.ai. Releasing 3b and 7b redpajama-incite family of models including base, instruction-tuned & chat models, 2023. URL <https://www.together.ai/blog/redpajama-models-v1>.
- Tran, D., Kirsch, A., Lakshminarayanan, B., Hu, H., Phan, D., Sculley, D., Snoek, J., Liu, J. Z., Ren, J., van Amersfoort, J., Han, K., Buchanan, E. K., Murphy, K. P., Collier, M., Dusenberry, M. W., Band, N., Thain, N., Jenatton, R., Rudner, T. G. J., Gal, Y., Nado, Z., Mariet, Z. E., Wang, Z., and Ghahramani, Z. Plex: Towards reliability using pretrained large model extensions. In *ICML 2022 Workshop on Pre-training*, 2022.
- Wallsten, T. *Measuring Vague Uncertainties and Understanding Their Use in Decision Making*, pp. 377–398. Measuring Vague Uncertainties and Understanding Their Use in Decision Making, 01 1990. ISBN 978-90-481-5785-3. doi: 10.1007/978-94-015-7873-8_15.
- Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models, 2022. URL <https://arxiv.org/abs/2203.11171>.
- Wang, X., Wei, J., Schuurmans, D., Le, Q. V., Chi, E. H., Narang, S., Chowdhery, A., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=1PL1NIMMrw>.
- Welbl, J., Liu, N. F., and Gardner, M. Crowdsourcing multiple choice science questions, 2017.
- Whiting, M. E., Hugh, G., and Bernstein, M. S. Fair work: Crowd work minimum wage with one line of code. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 7(1):197–206, Oct. 2019. doi: 10.1609/hcomp.v7i1.5283. URL <https://ojs.aaai.org/index.php/HCOMP/article/view/5283>.
- Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL <https://www.aclweb.org/anthology/2020.emnlp-demos.6>.
- Xiong, M., Hu, Z., Lu, X., LI, Y., Fu, J., He, J., and Hooi, B. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=gjeQKFxFpZ>.
- Yang, Y., Chern, E., Qiu, X., Neubig, G., and Liu, P. Alignment for honesty, 2023.
- Ye, F., Yang, M., Pang, J., Wang, L., Wong, D. F., Yilmaz, E., Shi, S., and Tu, Z. Benchmarking llms via uncertainty quantification, 2024.
- Zadrozny, B. and Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *Icml*, volume 1, pp. 609–616, 2001.
- Zhao, S. and Ermon, S. Right decisions from wrong predictions: A mechanism design alternative to individual calibration. In Banerjee, A. and Fukumizu, K. (eds.), *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2683–2691. PMLR, 13–15 Apr 2021. URL <https://proceedings.mlr.press/v130/zhao21a.html>.
- Zhao, S., Kim, M., Sahoo, R., Ma, T., and Ermon, S. Calibrating predictions to decisions: A novel approach to multi-class calibration. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 22313–22324. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/bbc92a647199b832ec90d7cf57074e9e-Paper.pdf.

Zhao, Y., Gu, A., Varma, R., Luo, L., Huang, C.-C., Xu, M., Wright, L., Shojanazeri, H., Ott, M., Shleifer, S., Desmaison, A., Balioglu, C., Damania, P., Nguyen, B., Chauhan, G., Hao, Y., Mathews, A., and Li, S. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023.

Zhou, K., Jurafsky, D., and Hashimoto, T. Navigating the grey area: How expressions of uncertainty and overconfidence affect language models. In Bouamor, H., Pino, J., and Bali, K. (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5506–5524, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.335. URL <https://aclanthology.org/2023.emnlp-main.335>.

Supplementary Material

Table of Contents

Appendix A: Additional Related Work

Appendix B: Additional Results

Appendix C: Benefits of Linguistic Calibration for Decision-making

Appendix D: Training Framework

Appendix E: Evaluation Framework

A Additional Related Work

Calibration and decision-making. The particular decision-making framework we adopted was originally used to convey the confidence of individual predictions to decision-makers (Zhao & Ermon, 2021), and later used to draw an equivalence between different notions of calibration and optimal decision-making (Zhao et al., 2021) (cf. §2.2 and Appendix C). In seminal work, Foster & Vohra (1998; 1999) showed that the existence of certain no regret schemes in an online decision-making setting imply the existence of calibrated probabilistic forecasts. More recently, Cresswell et al. (2024) explore using conformal prediction sets in classification tasks to improve human decision-making, but do not consider calibrating long-form LM generations.

Other LM finetuning works. Improving the factuality of LMs is a complementary approach to calibration in mitigating LM hallucinations. Previous works have improved the factuality of LMs by finetuning on self-supervised or automated factuality scores (Tian et al., 2024; Akyürek et al., 2024). A related line of work uses supervised finetuning and RL to improve the honesty of LLMs (Askeel et al., 2021; Ouyang et al., 2022; Park et al., 2023; Evans et al., 2021; Cui et al., 2023), hypothesizing that the pretraining objective alone is insufficient to encourage honest responses. Because improving factuality alone can improve calibration metrics such as ECE, we include a strong baseline finetuned with RL on ground-truth factuality labels and find that our approach to linguistic calibration significantly improves ECE beyond this baseline while matching or exceeding its accuracy.

B Additional Results

B.1 Codebase

For prompts, checkpoints, and implementations of methods and datasets, please refer to our codebase:

https://github.com/tatsu-lab/linguistic_calibration

B.2 Additional Baselines

In addition to the baselines described in §4.1, we below provide full results including several other baselines. All references to GPT-4 in this paper use version `gpt-4-1106-preview`. Unless otherwise specified, all references to Claude or Claude 2 use version `claude-2.0`. Due to compute constraints, all GPT-4 baselines are evaluated on a smaller number of question-answering test samples: 1000 samples for TriviaQA, Jeopardy, and SciQ, and the entire 1515 sample test set for BioASQ Task B (cf. Appendix E for further dataset and evaluation details).

We include the following additional non-confidence baselines:

- **Llama 2 Chat.** We zero-shot prompt Llama 2 Chat to generate long-form responses to evaluation queries.
- **GPT-4 0-Shot.** We zero-shot prompt GPT-4 to generate long-form responses to evaluation queries.
- **GPT-4 ICL 8-Shot.** Analogous to the Llama 2 7B **ICL** baseline. We randomly sample 8 queries, generate long-form responses with GPT-4, manually fact-check those responses using Wikipedia, and use these fact-checked (query, response) pairs as ICL examples for GPT-4.

We also include the following additional confidence baselines:

- **GPT-4 Just Ask for Uncertainty (JAFU) 0-Shot.** We zero-shot prompt GPT-4 to generate long-form responses to evaluation queries, and include an instruction in the prompt directing GPT-4 to indicate any uncertainty in its claims using probabilities.
- **GPT-4 Summary ICL 8-Shot.** Analogous to the Llama 2 7B **Summary ICL** baseline. We sample 8 queries from the prompt development split $\{q^{(i)}\}_{i=1}^8$ (*i.e.*, questions x which have been converted to open-ended queries). For each query $q^{(i)}$, we generate 8 long-form responses from GPT-4: $\{z_j^{(i)}\}_{j=1}^8 \sim \pi_{\text{GPT-4}}(z \mid q^{(i)})$. We then summarize these responses into a single consensus response: $s^{(i)} \sim \text{Summarize}(s \mid z_1^{(i)}, \dots, z_8^{(i)})$. Finally, we use the queries and summaries as ICL examples $\{(q^{(i)}, s^{(i)})\}_{i=1}^8$.
- **Direct Summary Eval.** For a given evaluation query q , we use the Llama 2 7B **ICL** baseline to generate 8 long-form responses $\{z_i\}_{i=1}^8$, and use Claude 2 to summarize these responses into a single consensus response: $s \sim \text{Summarize}(s \mid z_1, \dots, z_8)$. Then, we directly evaluate these summaries s . **This is an oracle baseline** because it requires sampling several long-form generations from the base LM at evaluation time, followed by summarization with an API-based LLM.

B.3 TriviaQA: Full Accuracy-ECE Frontier

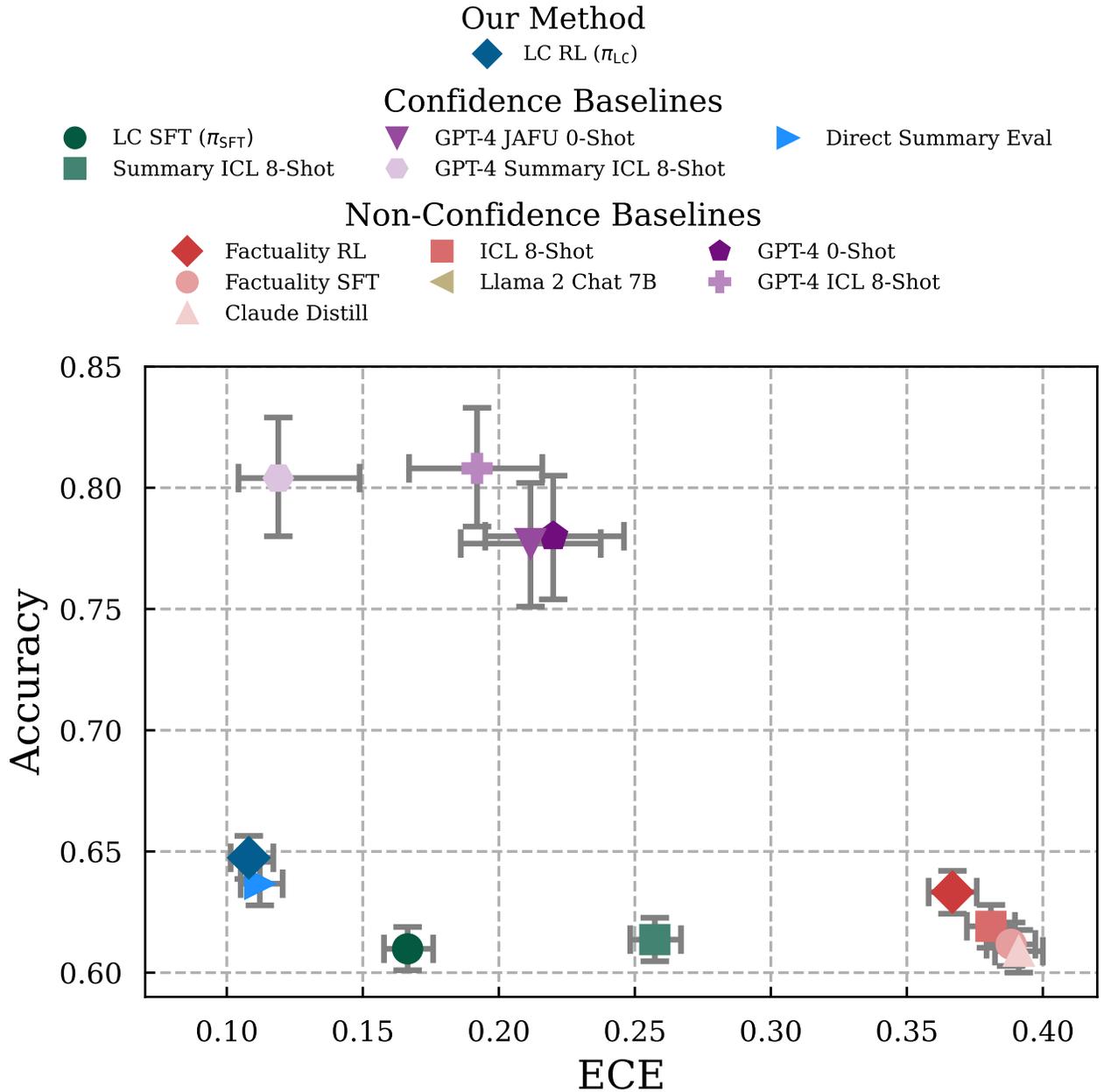
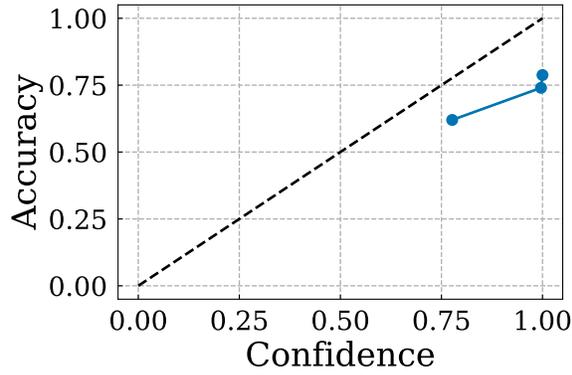
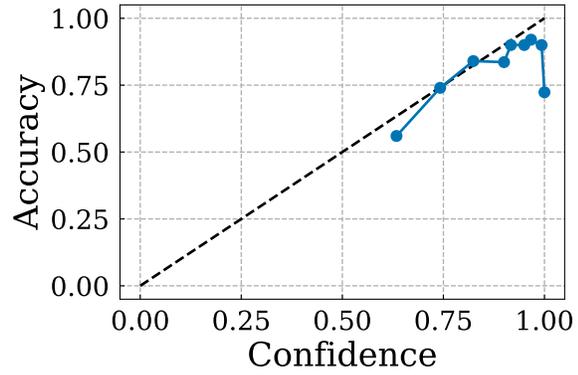


Figure 7: **Accuracy-ECE Frontier for All Methods on TriviaQA**, evaluated with simulated reader (upper left is better). GPT-4 Summary ICL works well despite its simplicity. LC RL outperforms all methods in ECE, including GPT-4 Summary ICL and Direct Summary Eval. Llama 2 Chat 7B performs significantly worse than all other methods and is not displayed for clarity (cf. Table 1 for numerical results).

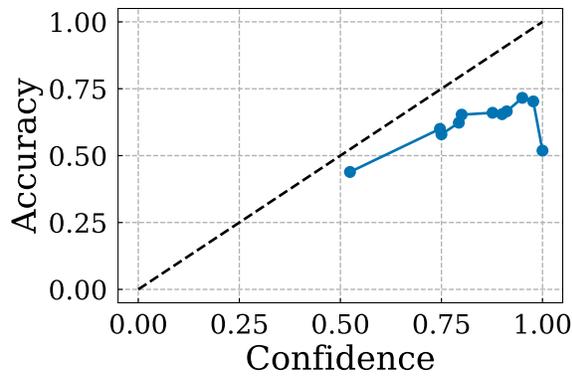
B.4 TriviaQA: Additional Reliability Diagrams



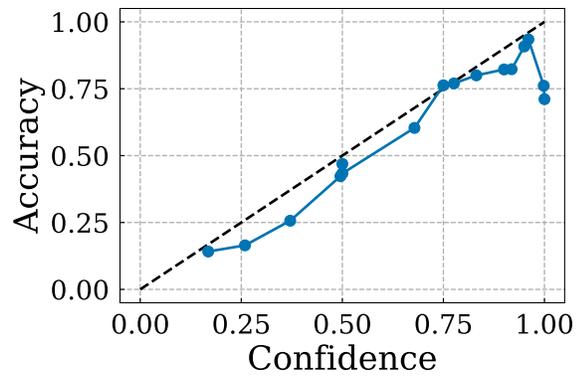
(a) GPT-4 JAFU 0-Shot. ECE: 0.212



(b) GPT-4 Summary ICL 8-Shot. ECE: 0.119



(c) Summary ICL 8-Shot. ECE: 0.257



(d) Direct Summary Eval. ECE: 0.112

Figure 8: **TriviaQA Reliability Diagrams** for additional baselines and oracle methods, simulated reader. Notably, Summary ICL works well for both Llama 2 7B (Summary ICL 8-Shot) and GPT-4. The oracle baseline of Direct Summary Eval has strong ECE.

B.5 Jeopardy: Full Accuracy-ECE Frontier

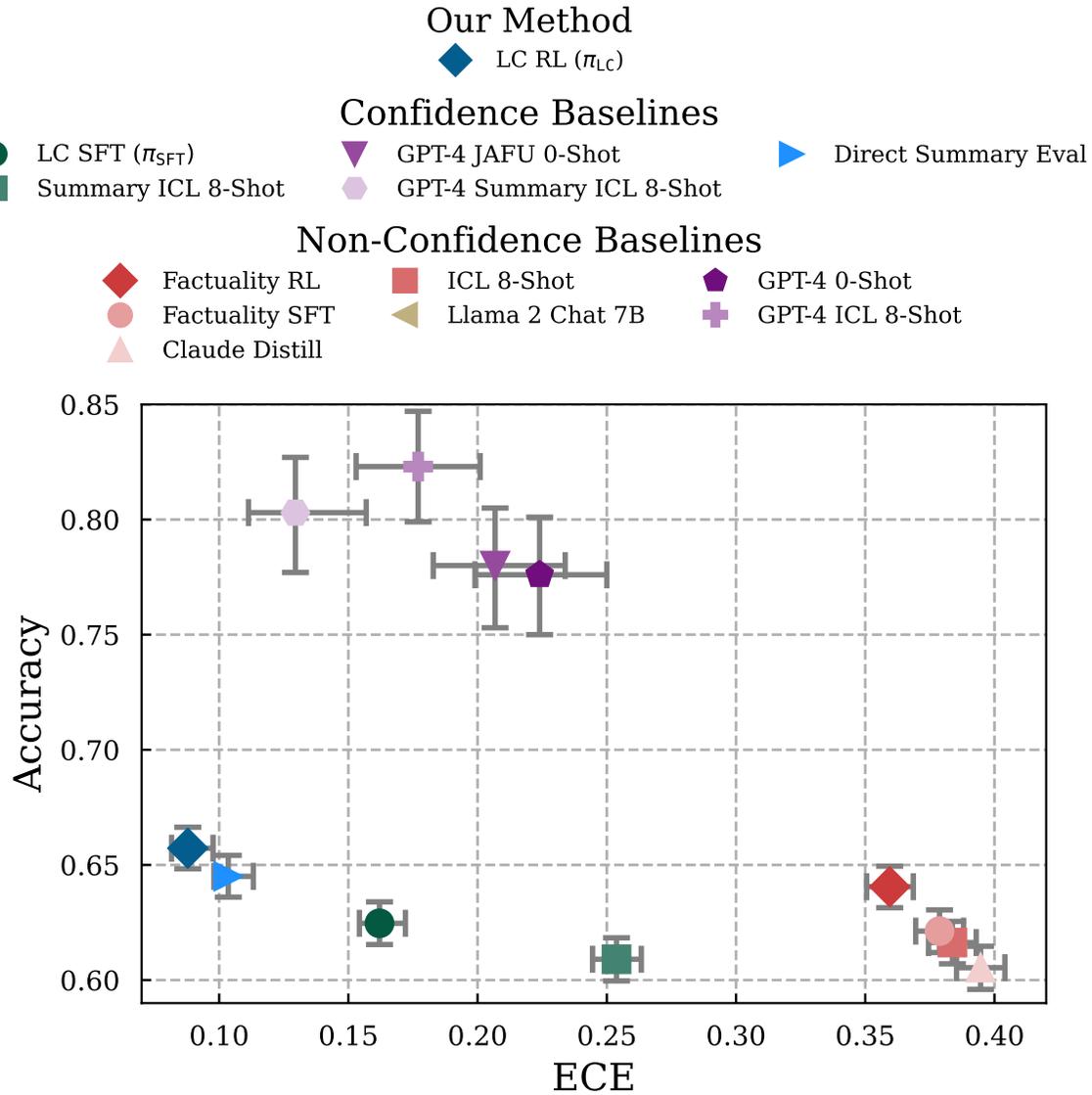


Figure 9: **Accuracy-ECE Frontier for All Methods on Jeopardy**, evaluated with simulated reader (upper left is better). Results are similar to TriviaQA. GPT-4 Summary ICL works well despite its simplicity. LC RL outperforms all methods in ECE, including GPT-4 Summary ICL and Direct Summary Eval. Llama 2 Chat 7B performs significantly worse than all other methods and is not displayed for clarity (cf. Table 1 for numerical results).

B.6 Jeopardy: All Reliability Diagrams

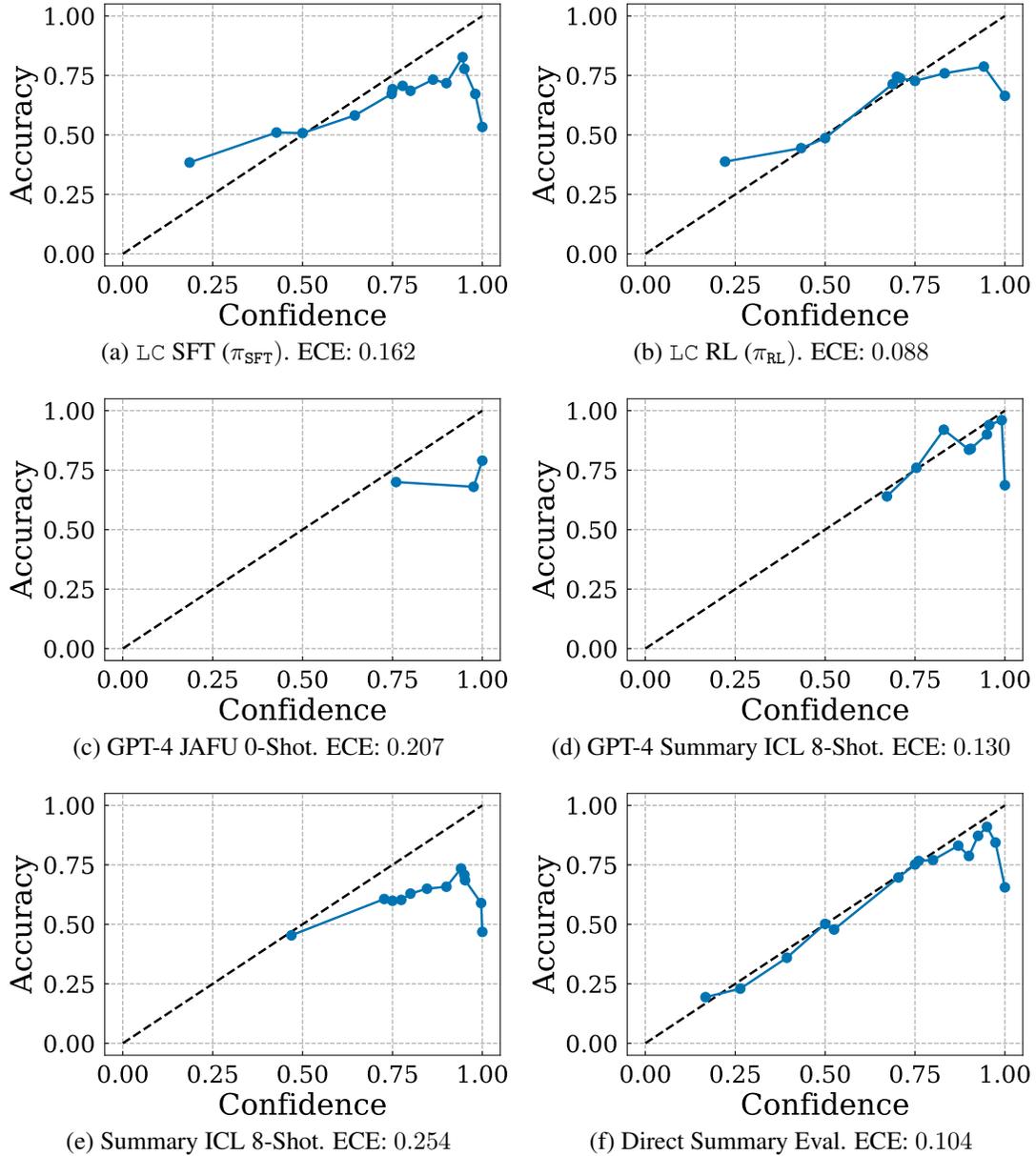


Figure 10: **Jeopardy Reliability Diagrams** for all methods, simulated reader. LC RL has better ECE than all other methods while predicting a variety of confidence levels. Notably, Summary ICL works well for both Llama 2 7B (Summary ICL 8-Shot) and GPT-4. The oracle baseline of Direct Summary Eval has strong ECE.

B.7 SciQ: Full Accuracy-ECE Frontier

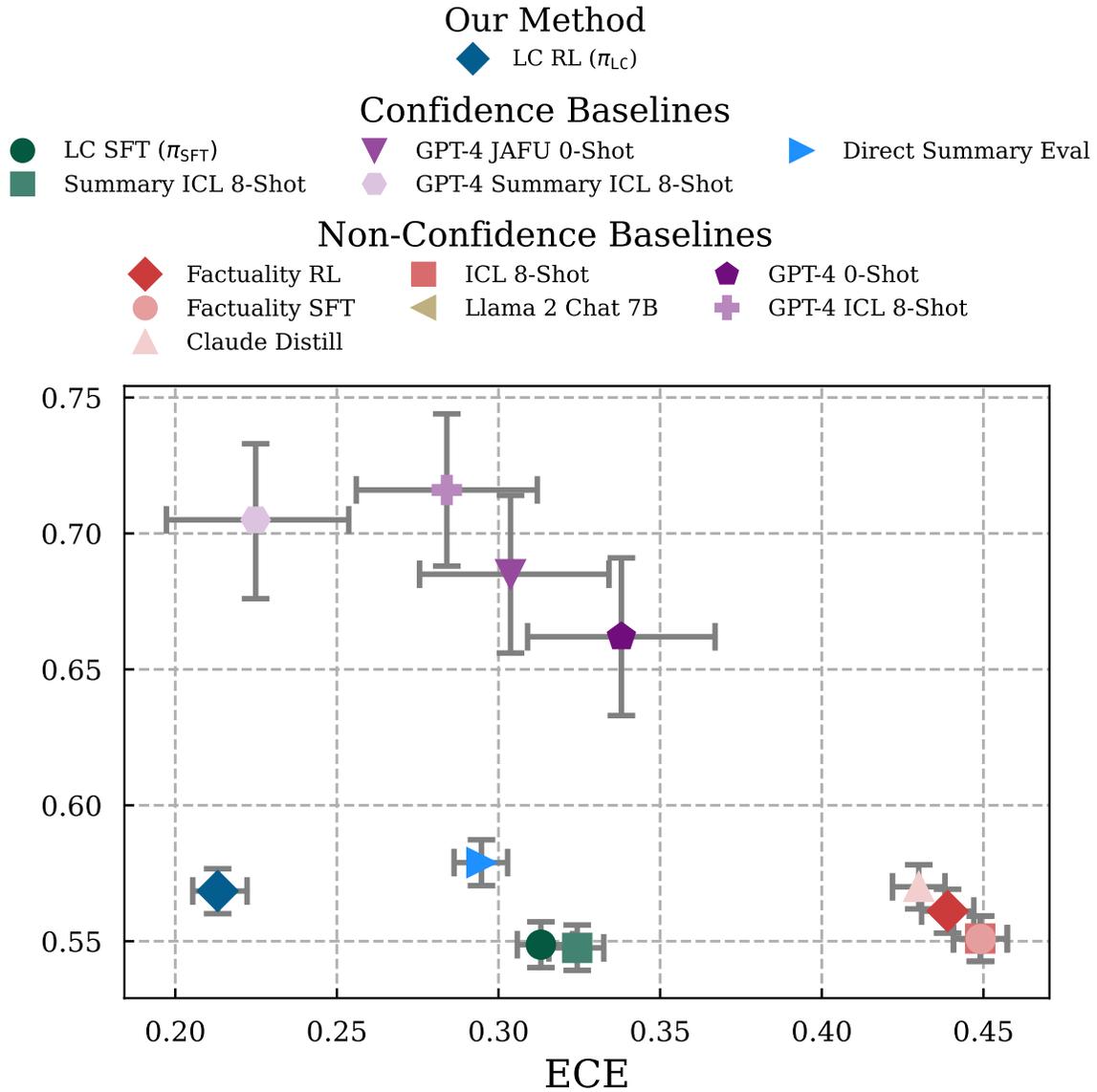


Figure 11: **Accuracy-ECE Frontier for All Methods on SciQ**, evaluated with simulated reader (upper left is better). GPT-4 Summary ICL works well despite its simplicity. LC RL outperforms all methods in ECE, including GPT-4 Summary ICL and Direct Summary Eval. Llama 2 Chat 7B performs significantly worse than all other methods and is not displayed for clarity (cf. Table 1 for numerical results).

B.8 SciQ: All Reliability Diagrams

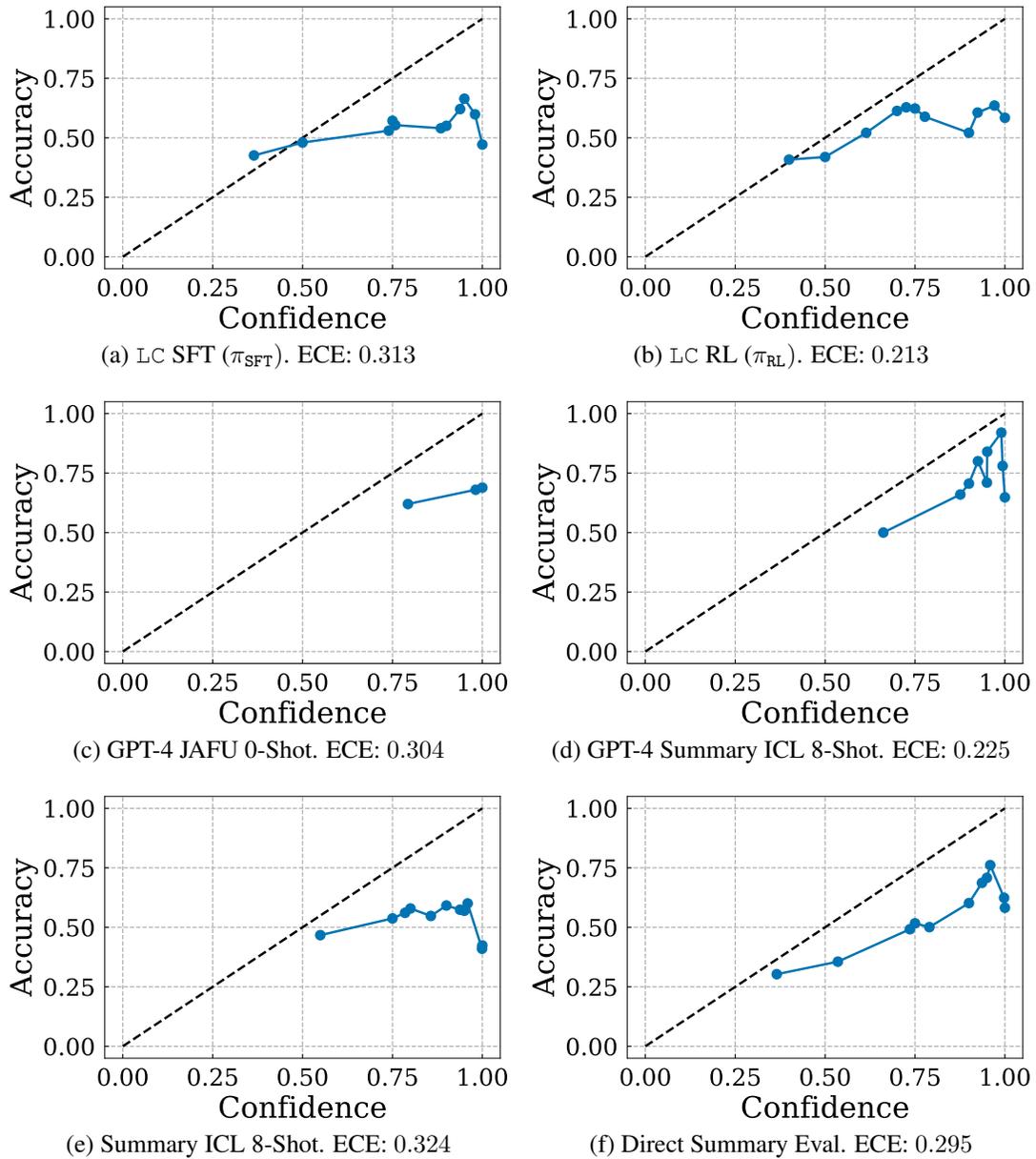


Figure 12: **SciQ Reliability Diagrams** for all methods, simulated reader. LC RL has better ECE than all other methods while predicting a variety of confidence levels.

B.9 BioASQ Task B: Full Accuracy-ECE Frontier

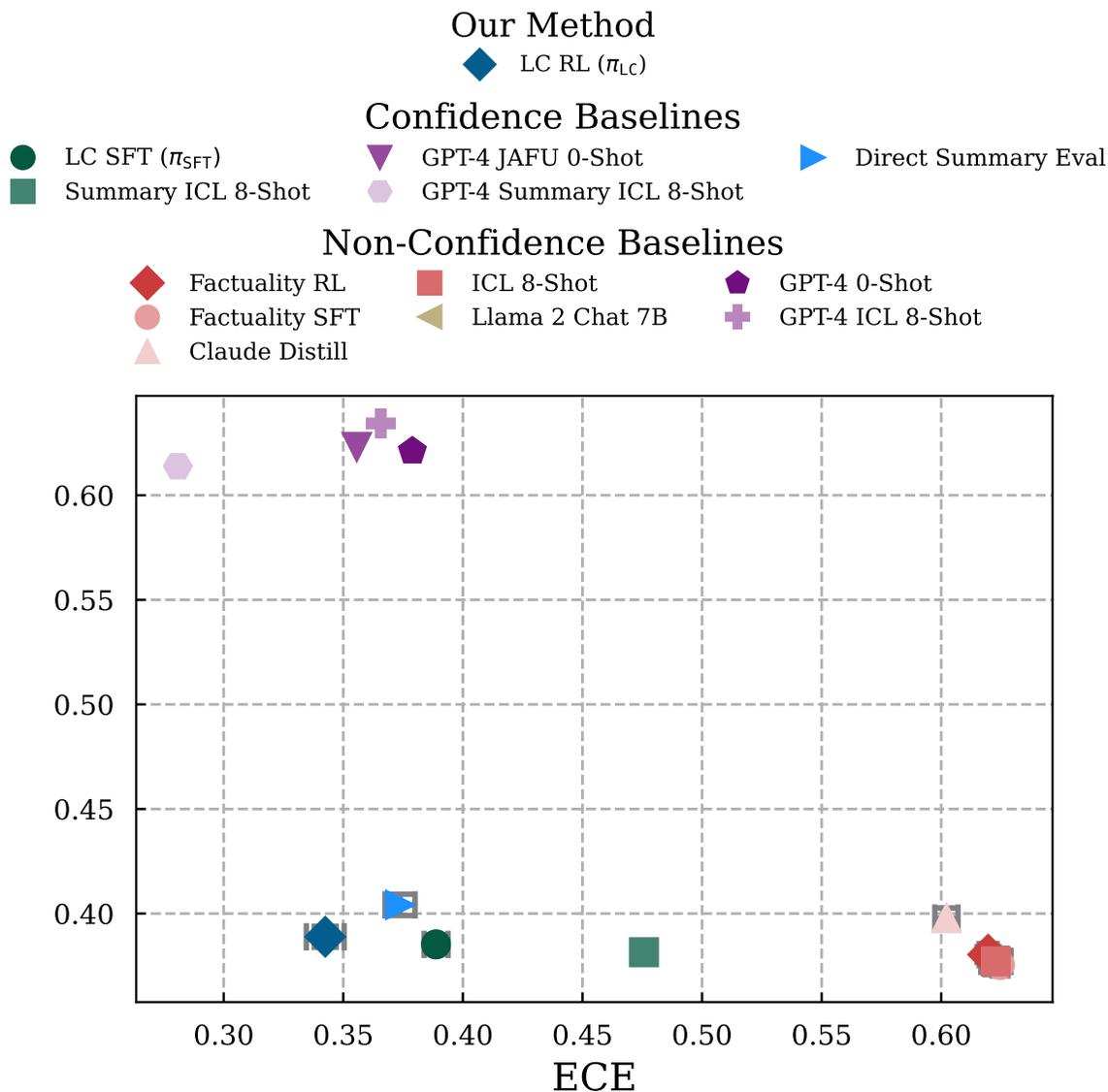


Figure 13: **Accuracy-ECE Frontier for All Methods on BioASQ Task B**, evaluated with simulated reader (upper left is better). Under a significant distribution shift, LC RL ECE is competitive with GPT-4-based methods and outperforms all Llama-based baselines. BioASQ is a small expert-annotated dataset, so we report Student’s t 0.95 CIs over 10 evaluation seeds for non-GPT-4 methods. Llama 2 Chat 7B performs significantly worse than all other methods and is not displayed for clarity (cf. Table 1 for all numerical results).

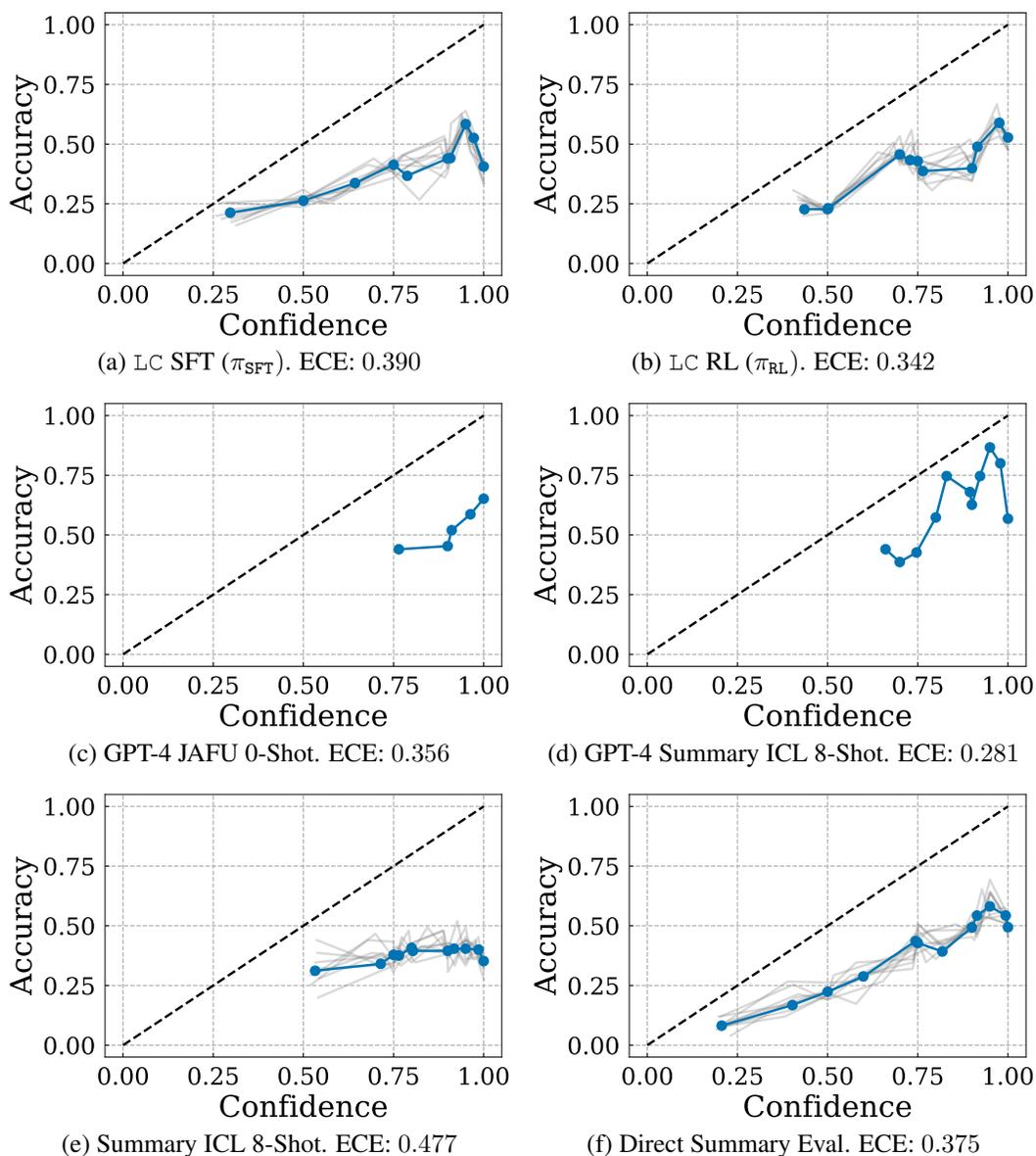
B.10 BioASQ Task B: All Reliability Diagrams


Figure 14: **BioASQ Task B Reliability Diagrams** for all methods, simulated reader. Under significant distribution shift, LC RL has ECE competitive with GPT-4 methods while predicting a variety of confidence levels. BioASQ is a small expert-annotated dataset, so for non-GPT-4 methods, we pool examples across 10 evaluation seeds when plotting the reliability diagram (per-seed plots in gray).

B.11 Person Biography Generation Frontier

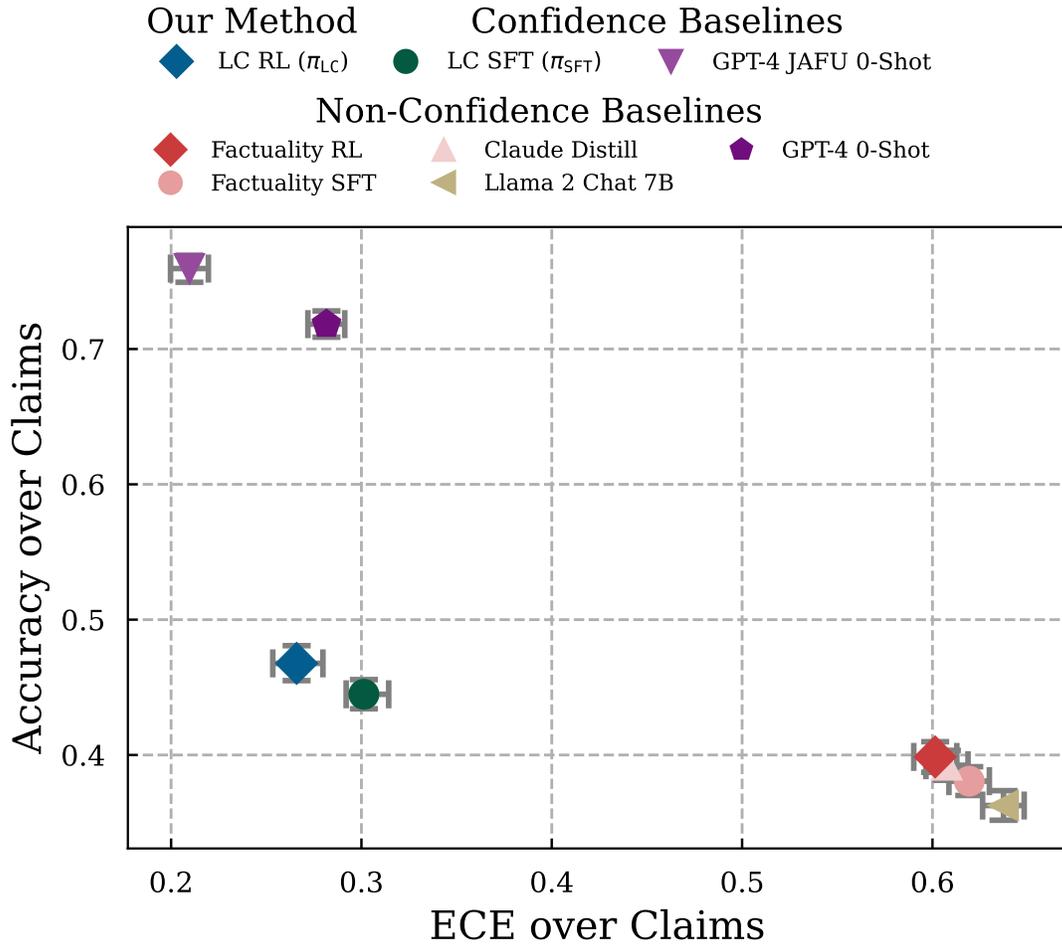


Figure 15: **Accuracy-ECE Frontier for All Methods on Person Biography Generation**, evaluated with simulated reader (upper left is better). GPT-4 is a strong baseline, and GPT-4 Just Ask for Uncertainty (JAFU) does improve both accuracy and ECE. Notably, LC RL has better ECE than GPT-4 0-Shot, despite being trained from a Llama 2 7B base model.

B.12 Tabular Results

Table 1: **Simulated Question-Answering Results.** Accuracy and expected calibration error (ECE) of Llama 2 7B and GPT-4-based methods on the in-distribution (ID) TriviaQA and out-of-distribution (OOD) Jeopardy, SciQ, and BioASQ Task B question-answering datasets, with simulated reader evaluation. “API-Based LLM Methods” are those that use either GPT-4 or Claude 2 at test time. CIs are reported as (lower, upper). For all datasets other than BioASQ, these are 95% bootstrap CIs; for the smaller BioASQ dataset, these are Student’s t 0.95 CIs over 10 evaluation seeds. Due to compute constraints, for GPT-4-based methods on BioASQ, we report accuracy and ECE for a single evaluation seed (cf. Appendix E for all evaluation details).

Method	TriviaQA (ID)		Jeopardy (OOD)		SciQ (OOD)		BioASQ Task B (OOD)	
	Accuracy (%) \uparrow	ECE \downarrow	Accuracy (%) \uparrow	ECE \downarrow	Accuracy (%) \uparrow	ECE \downarrow	Accuracy (%) \uparrow	ECE \downarrow
Llama 2 7B-Based Methods								
Llama 2 Chat 7B	52.32 (51.42, 53.27)	0.477 (0.467, 0.486)	50.86 (49.91, 51.82)	0.491 (0.482, 0.501)	51.52 (50.68, 52.34)	0.485 (0.477, 0.493)	35.31 (34.79, 35.82)	0.647 (0.642, 0.652)
Claude Distill	60.89 (59.98, 61.78)	0.391 (0.382, 0.400)	60.54 (59.64, 61.44)	0.395 (0.386, 0.404)	57.00 (56.17, 57.83)	0.430 (0.422, 0.438)	39.78 (39.30, 40.27)	0.602 (0.597, 0.607)
ICL 8-Shot	61.91 (61.02, 62.81)	0.381 (0.372, 0.390)	61.63 (60.69, 62.57)	0.384 (0.374, 0.393)	55.10 (54.27, 55.92)	0.449 (0.441, 0.457)	37.72 (37.09, 38.35)	0.623 (0.617, 0.629)
Factuality SFT	61.17 (60.26, 62.07)	0.388 (0.379, 0.397)	62.13 (61.21, 63.05)	0.379 (0.370, 0.388)	55.08 (54.26, 55.92)	0.449 (0.441, 0.457)	37.54 (37.22, 37.87)	0.625 (0.621, 0.628)
Factuality RL	63.33 (62.44, 64.21)	0.367 (0.358, 0.376)	64.05 (63.14, 64.97)	0.359 (0.350, 0.369)	56.11 (55.27, 56.94)	0.439 (0.431, 0.447)	38.04 (37.66, 38.42)	0.620 (0.616, 0.623)
Summary ICL 8-Shot	61.36 (60.47, 62.25)	0.257 (0.248, 0.267)	60.90 (59.98, 61.83)	0.254 (0.245, 0.263)	54.76 (53.91, 55.61)	0.324 (0.316, 0.333)	38.15 (37.77, 38.53)	0.476 (0.471, 0.480)
LC SFT (π_{SFT})	60.98 (60.10, 61.88)	0.166 (0.158, 0.176)	62.46 (61.53, 63.36)	0.162 (0.154, 0.172)	54.87 (54.04, 55.69)	0.313 (0.306, 0.323)	38.53 (38.19, 38.87)	0.389 (0.384, 0.394)
LC RL (π_{LC})	64.74 (63.86, 65.63)	0.108 (0.101, 0.117)	65.73 (64.83, 66.61)	0.088 (0.082, 0.098)	56.85 (56.02, 57.67)	0.213 (0.205, 0.222)	38.89 (38.39, 39.39)	0.342 (0.335, 0.350)
API-Based LLM Methods								
GPT-4 0-Shot	78.00 (75.30, 80.50)	0.220 (0.195, 0.247)	77.60 (75.00, 80.10)	0.224 (0.199, 0.250)	66.20 (63.20, 69.10)	0.338 (0.309, 0.368)	62.11	0.379
GPT-4 ICL 8-Shot	80.80 (78.40, 83.20)	0.192 (0.168, 0.216)	82.30 (80.00, 84.70)	0.177 (0.153, 0.200)	71.60 (68.80, 74.50)	0.284 (0.255, 0.312)	63.43	0.366
GPT-4 JAFU 0-Shot	77.70 (75.10, 80.20)	0.212 (0.187, 0.238)	78.00 (75.40, 80.50)	0.207 (0.182, 0.234)	68.50 (65.50, 71.40)	0.304 (0.274, 0.334)	62.31	0.356
GPT-4 Summary ICL 8-Shot	80.40 (77.90, 82.80)	0.119 (0.104, 0.148)	80.30 (77.80, 82.70)	0.130 (0.111, 0.156)	70.50 (67.70, 73.30)	0.225 (0.197, 0.253)	61.39	0.281
Direct Summary Eval	63.67 (62.78, 64.55)	0.112 (0.105, 0.120)	64.50 (63.59, 65.43)	0.104 (0.098, 0.114)	57.89 (57.04, 58.73)	0.295 (0.286, 0.303)	40.42 (39.91, 40.93)	0.374 (0.367, 0.380)

Table 2: **Human Question-Answering Results on TriviaQA.** Accuracy and expected calibration error (ECE) of core Llama 2 7B-based methods on the TriviaQA question-answering dataset, with human readers (cf. Appendix E for evaluation framework details). 95% bootstrap CIs are reported as (lower, upper).

Method	Accuracy (%) \uparrow	ECE \downarrow
Factuality RL	59.62 (56.65, 62.60)	0.404 (0.374, 0.434)
LC SFT (π_{SFT})	57.44 (54.37, 60.52)	0.163 (0.135, 0.192)
LC RL (π_{LC})	60.12 (57.14, 63.19)	0.116 (0.091, 0.145)

Table 3: **Person Biography Generation Results.** Accuracy and expected calibration error (ECE) of Llama 2 7B and GPT-4-based methods on the Person Biography Generation dataset, using a FactScore-based automated metric (Min et al., 2023). Metrics are computed over atomic claims pooled across all biography generations. 95% bootstrap CIs are reported as (lower, upper).

Method	Accuracy (%) \uparrow	ECE \downarrow
Llama 2 7B-Based Methods		
Llama 2 Chat 7B	36.27 (35.18, 37.36)	0.637 (0.626, 0.648)
Claude Distill	39.24 (38.13, 40.34)	0.608 (0.597, 0.619)
Factuality SFT	38.07 (37.01, 39.13)	0.619 (0.609, 0.630)
Factuality RL	39.86 (38.72, 40.98)	0.601 (0.590, 0.613)
LC SFT (π_{SFT})	44.49 (43.42, 45.57)	0.301 (0.292, 0.314)
LC RL (π_{LC})	46.77 (45.50, 48.08)	0.266 (0.253, 0.280)
GPT-4-Based Methods		
GPT-4 0-Shot	71.84 (70.88, 72.81)	0.282 (0.272, 0.291)
GPT-4 JAFU 0-Shot	75.95 (74.93, 76.93)	0.210 (0.200, 0.220)

B.13 Qualitative Examples

We randomly sample two examples from each evaluation dataset and compare Factuality RL with LC RL. We manually fact-check each generation using Wikipedia and other Google-accessible sources and highlight incorrect statements in red. Numerical and linguistic confidence statements are highlighted in blue.

Dataset: TriviaQA

Query q : Write a paragraph about the classic book in which a boy hides in a ship's barrel of apples.

Ground-Truth Answer y : Treasure Island

Factuality RL: The classic book in which a boy hides in a ship's barrel of apples is "Treasure Island," written by Robert Louis Stevenson. The story follows the adventures of Jim Hawkins, a young boy who lives in an inn on an island off the coast of England. One day, a mysterious sailor named Long John Silver arrives at the inn, and Jim becomes involved in a treasure hunt for a buried treasure on the island. Jim hides in a barrel of apples on the ship that is carrying the treasure, and he is able to escape with the treasure when the ship is attacked by pirates. "Treasure Island" is a classic tale of adventure and intrigue, and it has been adapted into numerous films and stage productions.

LC RL: Based on my understanding, the classic book in which a boy hides in a ship's barrel of apples is most likely "Treasure Island" by Robert Louis Stevenson. I estimate there is a 75% chance it is "Treasure Island", a 15% chance it is "The Wonderful Wizard of Oz" by L. Frank Baum, a 5% chance it is "The Adventures of Tom Sawyer" by Mark Twain, and a 5% chance it is another book. I believe the story involves a young boy named Jim Hawkins who hides in a barrel of apples to avoid being discovered by pirates. Though I do not know all the details, I am confident the boy's adventures involve searching for buried treasure. Regardless of the specific book, I am certain this classic tale has become beloved by generations of readers.

Dataset: TriviaQA

Query q : Write a paragraph about the type of creature that a tanager is.

Ground-Truth Answer y : Bird

Factuality RL: A tanager is a type of bird that belongs to the Thraupidae family. Tanagers are known for their brightly colored plumage, with many species featuring vibrant red, yellow, and blue feathers. They are found in a wide range of habitats, including forests, savannas, and grasslands, and are known for their acrobatic flight patterns. Tanagers are social birds that often form flocks, and they are known for their loud and distinctive calls.

LC RL: I believe a tanager is a type of bird that belongs to the Thraupidae family. I am confident they are found in the Americas, though I am uncertain of their exact range. I estimate with 75% likelihood that tanagers are brightly colored birds with distinctive plumage. I am 50% sure they are known for their songbird abilities. I am uncertain whether they are found in forests, grasslands, or other habitats. I believe with 75% likelihood that tanagers feed on insects and fruit. While the details of their appearance and behavior are unclear to me, I am certain that tanagers are colorful, songbirds native to the Americas.

Dataset: SciQ

Query q : Write a paragraph about the term used to describe elements that specifically are noted to have properties of both metals and nonmetals.

Ground-Truth Answer y : Metalloids

Factuality RL: The term used to describe elements that specifically are noted to have properties of both metals and nonmetals is "metalloids." Metalloids are elements that have some of the properties of metals, such as conductivity and malleability, and some of the properties of nonmetals, such as a tendency to form covalent bonds. Examples of metalloids include silicon, germanium, and arsenic.

LC RL: Based on my understanding, the term used to describe elements that specifically are noted to have properties of both metals and nonmetals is either "metalloids", "semimetals", or "semiconductors". I estimate there is a 70% likelihood that the term is "metalloids", a 20% chance it is "semimetals", and a 10% chance it is "semiconductors". I believe these elements have properties that are somewhere between metals and nonmetals, though I do not know all the details. I am confident that they are used in various applications like electronics, batteries, and semiconductors. However, I am uncertain about the specific elements that fall into this category. Overall, despite my lack of complete knowledge, I am sure that there is a term used to describe elements with properties of both metals and nonmetals.

Dataset: SciQ

Query q : Write a paragraph about the periods when most plants do not grow continuously.

Ground-Truth Answer y : Dormant

Factuality RL: The periods when most plants do not grow continuously are known as dormancy periods. During these periods, plants enter a state of inactivity, where they stop growing and may even shed leaves or go into a state of hibernation. Dormancy periods can be caused by various factors, such as cold temperatures, lack of sunlight, or insufficient water. Plants may also enter dormancy periods to conserve energy and resources during times of stress or drought. Dormancy periods are an important part of the plant's life cycle, allowing them to adapt to changing environmental conditions and survive through challenging times.

LC RL: Based on my understanding, the periods when most plants do not grow continuously are known as dormant periods. I am confident that dormancy occurs in response to environmental factors like cold temperatures, lack of sunlight, or limited water availability. However, I am uncertain about the exact duration of dormancy for different plant species. I estimate there is a 50% chance that dormancy lasts for several months, and a 50% chance it lasts for several years. I believe dormancy allows plants to conserve energy and resources, though I do not know all the details. While dormancy may seem like a state of inactivity, I am certain it is an important adaptive strategy for plants to survive harsh conditions. Overall, despite my lack of complete knowledge, I am confident that dormancy is a natural part of plant growth cycles.

Dataset: BioASQ Task B

Query q : Write a paragraph about the gene that is most commonly mutated in Tay-Sachs disease.

Ground-Truth Answer y : HEXA

Factuality RL: The gene that is most commonly mutated in Tay-Sachs disease is the HEXA gene, which encodes the alpha subunit of the hexosaminidase A enzyme. This enzyme is responsible for breaking down the sugar hexosamine, which is found in the membranes of cells. When the HEXA gene is mutated, the alpha subunit of the hexosaminidase A enzyme is **not produced**, leading to a buildup of **hexosamine** in the cells. This buildup can cause neurological damage and eventually lead to death. Tay-Sachs disease is a rare genetic disorder that **primarily** affects Ashkenazi Jews and is caused by a mutation in the HEXA gene.

LC RL: **Based on my understanding**, the gene that is most commonly mutated in Tay-Sachs disease is either the HEXA gene or the **GAA** gene. **I estimate there is a 70% likelihood** that the HEXA gene is mutated, and a **30% chance** that the **GAA** gene is mutated. **I believe** the HEXA gene encodes an enzyme called beta-hexosaminidase A, while the GAA gene encodes an enzyme called **beta-glucuronidase**. **I am confident** that mutations in these genes lead to a deficiency of these enzymes, which causes accumulation of toxic substances in the nervous system and other tissues. This results in neurological symptoms like abnormal eye movements, seizures, and mental decline. **While I do not know all the details, I believe** Tay-Sachs disease is a rare genetic disorder that **primarily** affects Ashkenazi Jews.

Dataset: BioASQ Task B

Query q : Write a paragraph about the type of GTPases that are required for amino acid-dependent activation of mTORC1.

Ground-Truth Answer y : Heterodimeric Rag GTPases

Factuality RL: The type of GTPases that are required for amino acid-dependent activation of mTORC1 are **rapamycin-sensitive** GTPases. These GTPases are involved in the regulation of cell growth and proliferation, and are activated by amino acids, such as leucine, isoleucine, and valine. The activation of these GTPases leads to the phosphorylation of the mTORC1 complex, which in turn activates downstream signaling pathways that regulate cell growth and proliferation. The activation of mTORC1 by amino acids is a crucial step in the regulation of cell growth and proliferation, and is a key target for the development of therapeutics for various diseases, including cancer and diabetes.

LC RL: **Based on my understanding**, the type of GTPases required for amino acid-dependent activation of mTORC1 are either **mTORC1** itself, Rheb, or Rag GTPases. **I estimate there is a 70% likelihood** that **mTORC1** itself is the GTPase required, a **20% chance** that Rheb is the GTPase required, and a **10% chance** that Rag GTPases are the GTPases required. **I believe** the activation of mTORC1 by amino acids involves the **binding** of amino acids to Rag GTPases, **which then activates mTORC1**. **However, I am uncertain about the exact details** of how this process works. **Regardless of the specific GTPase involved, I am confident** that amino acid activation of mTORC1 plays a crucial role in regulating cell growth, metabolism, and protein synthesis.

Dataset: Jeopardy

Query q : Write a paragraph about what Hiram Percy Maxim invented for guns, in addition to inventing the muffler for cars.

Ground-Truth Answer y : A silencer

Factuality RL: Hiram Percy Maxim, an American inventor and entrepreneur, is best known for his contributions to the firearms industry. In addition to inventing the muffler for cars, Maxim also **invented the Maxim machine gun**, a rapid-fire weapon that revolutionized warfare in the early 20th century. The Maxim machine gun was the first fully automatic weapon to be widely used in combat, and it played a significant role in the Allied victory in World War I. **Maxim's invention of the Maxim machine gun earned him the nickname "the father of the machine gun."**

LC RL: I **believe** that Hiram Percy Maxim, an American inventor, invented at least one significant invention related to guns in addition to inventing the muffler for cars. **Based on my understanding, I estimate there is a 50% chance** he invented the **Maxim gun**, a **25% chance** he invented the silencer, and a **25% chance** he invented the **machine gun**. I am **confident** that regardless of the specific invention, Maxim's work revolutionized the firearms industry and had a significant impact on warfare. **Though I do not know all the details, I believe** his inventions were known for their efficiency, accuracy, and durability. **While the specifics are unclear to me, I am certain** that Hiram Percy Maxim was an influential American inventor who made important contributions to the development of firearms technology.

Dataset: Jeopardy

Query q : Write a paragraph about the planet that a probe from the Galileo spacecraft plunged into the atmosphere of on December 7, 1995.

Ground-Truth Answer y : Jupiter

Factuality RL: The planet that a probe from the Galileo spacecraft plunged into the atmosphere of on December 7, 1995, was Jupiter. The Galileo probe was launched by NASA in 1989 and arrived at Jupiter in 1995, where it conducted numerous scientific studies of the planet and its moons. **On December 7, 1995**, the probe was intentionally crashed into Jupiter's **atmosphere, where it collected data on the planet's atmosphere and magnetic field**. The Galileo probe's mission was a significant achievement in the study of Jupiter and the solar system, and its impact on the field of planetary science is still felt today.

LC RL: I am **certain** that on December 7, 1995, a probe from the Galileo spacecraft plunged into the atmosphere of Jupiter. I **believe** this was a major milestone in space exploration, as it was the first time a spacecraft had been intentionally sent to study Jupiter up close. **I estimate with 90% confidence** that the probe was named Galileo after the Italian astronomer Galileo Galilei. I am **fairly confident** that the probe carried instruments to study Jupiter's atmosphere, **magnetic field**, and **moons**. **Though I do not know all the details, I believe** the data collected by the probe helped advance our understanding of Jupiter **and its moons**. **While the specifics are unclear to me, I am certain** that the Galileo probe's plunge into Jupiter's atmosphere marked an important achievement in space exploration.

Dataset: Person Biography Generation

Query q : Write a paragraph bio about Rory Byrne.

Factuality RL: Rory Byrne is a renowned **British** engineer and designer who has made significant contributions to the world of Formula One racing. Born in **1949**, Byrne began his career in the automotive industry in the 1970s, working for various racing teams and manufacturers. **In 1981, he joined the Williams Formula One team as a designer, and over the next few years, he helped the team win several championships.** In 1994, Byrne left Williams to join Ferrari, where he served as the team's technical director for several years. During his time at Ferrari, Byrne helped the team win several championships, including the 1999 constructor's championship. **In 2005, Byrne left Ferrari to pursue other opportunities, and he has since worked as a consultant for various racing teams and manufacturers.**

LC RL: I **believe** Rory Byrne is a renowned **British** engineer who has made significant contributions to the automotive industry. I **estimate with 75% confidence** that he has worked as a chief designer for Formula 1 teams like Ferrari and **McLaren**. I also **think with 50% likelihood** that he has designed cars for teams like **Williams**, Benetton, and **Lotus**. I am **uncertain** whether he was born in Ireland or Scotland. I **believe with 90% confidence** that he has won multiple design awards, **including the prestigious Loeb Award.** **While the details of his career are unclear to me, I am certain that** Rory Byrne has had a profound impact on the development of Formula 1 cars over the past few decades. His innovative designs have helped teams achieve success on the track.

Dataset: Person Biography Generation

Query q : Write a paragraph bio about Griselda Blanco.

Factuality RL: Griselda Blanco, also known as "The Black Widow," was a Colombian drug lord and crime boss who operated in the 1970s and 80s. She was known for her ruthlessness and her involvement in the cocaine trade, which made her one of the most powerful drug lords in the world. Blanco was responsible for the murder of numerous rivals and enemies, and **her criminal activities led to her being listed as one of the FBI's Ten Most Wanted Fugitives in 1985.** Despite her criminal activities, Blanco was also known for her philanthropy, and she was involved in various charitable organizations in Colombia.

LC RL: I **believe** Griselda Blanco was a Colombian drug lord who operated in the 1970s and 1980s. I **estimate with 75% confidence** that she was known as the "Black Widow" due to her ruthlessness and alleged involvement in numerous murders. I am **fairly certain** she was involved in cocaine trafficking and organized crime, **though I am uncertain about the specifics of her operations.** I **believe with 50% likelihood** that she was born in 1943 in Colombia. I am **confident** she was arrested **multiple times** and imprisoned, **though I do not know the details.** **While the specifics are unclear to me, I am certain** that Griselda Blanco was a notorious drug lord who left a lasting legacy of violence and crime in Colombia and the United States.

Dataset: Person Biography Generation

Query q : Write a paragraph bio about Uee.

Factuality RL: Uee, also known as Lee Hye-young, is a South Korean singer and actress. She is best known for her work in the K-pop girl group After School, where she served as the main vocalist and lead dancer. Uee has also appeared in several television dramas, including "The Greatest Love," "The Greatest Love," and "The Greatest Love." In addition to her musical and acting career, Uee is also known for her philanthropic work, including her participation in the "2014 Samsung Hope for Children" charity event.

LC RL: I am confident that Uee is a South Korean actress and singer. I estimate with 75% likelihood that she was born in 1988 in Seoul, South Korea. I believe with 50% probability that she made her acting debut in 2007 in the television drama "Love Needs a Miracle." I am uncertain whether she has appeared in films like "The Concubine" or "The King's Case Note." I am confident that she has been a member of the girl groups After School and Orange Caramel. I believe with 75% likelihood that she has won awards for her acting and singing, though I do not know the specifics. Overall, despite my lack of knowledge on some details, I am certain that Uee is a talented South Korean actress and singer who has had a successful career in the entertainment industry.

Table 4: Each notion of linguistic ϕ -calibration corresponds to a traditional notion of classifier calibration ϕ applied to the reader $f : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$.

Notion of Classifier Calibration ϕ	Condition for Linguistic ϕ -Calibration
Distribution Calibration (Kull & Flach, 2015)	$\mathbb{E}[\mathbf{1}_y \mid f(x, z) = p] = p, \quad \forall p \in \Delta^{ \mathcal{Y} }$.
Classwise Calibration (Kull et al., 2019)	$\Pr(Y = y \mid f(x, z)_y = \beta) = \beta, \quad \forall y \in \mathcal{Y}, \beta \in [0, 1]$.
Confidence Calibration (Guo et al., 2017)	$\Pr(Y = \arg \max_y f(x, z)_y \mid \max_y f(x, z)_y = \beta) = \beta, \quad \forall \beta \in [0, 1]$.

C Benefits of Linguistic Calibration for Decision-making

In §2, we describe the connection between calibration and decision-making (Zhao & Ermon, 2021; Zhao et al., 2021) and how it motivates our objective for linguistic calibration: namely, that maximizing our objective implies zero expected regret for a downstream Bayes-optimal decision-maker. Here, we expand on this guarantee and provide additional ones using the results of Zhao et al. (2021), which apply even if we cannot perfectly maximize the linguistic calibration objective. In particular, we will see that forecasts fulfilling weaker notions of calibration still guarantee that decisions are optimal in a weaker sense.

Following the generative process described in §2.1, we assume that the linguistic calibration objective is optimized over the real-world distribution of LM queries and related question-answer pairs $p(q, x, y)$. In practice, we found that using off-the-shelf question-answering datasets to generate a surrogate distribution (cf. §3.1) robustly improves an LM’s ability to express confidence levels in text, including under distribution shift. Therefore, the generalization properties of LC may justify this assumption. Moreover, we believe an exciting avenue for future work is to curate a more representative question-answer distribution (cf. §6). Lastly, recall that our generative process makes the following technical conditional independence assumption:

Assumption C.1. Under the ground-truth distribution, the answer Y is independent of the open-ended LM query and LM response (Q, Z) , given the question X : $Y \perp (Q, Z) \mid X$.

C.1 Review of Linguistic Calibration

Recall that each notion of linguistic ϕ -calibration (Defn. 2.1) is equivalent to a traditional notion of classifier calibration ϕ applied to the reader f . We provide examples of this correspondence in Table 4.

C.2 Review of the LC Objective

Our decision-based RL algorithm optimizes a strictly proper scoring rule R_{LC} of user forecasts $f(x, z) \in \Delta^{|\mathcal{Y}|}$ with respect to an LM π_{RL} producing long-form generations $z \sim \pi_{\text{RL}}(z \mid q)$ (restatement of Eq. 1 for convenience):

$$\max_{\pi_{\text{RL}}} \mathbb{E}_{(q, x, y) \sim p(q, x, y), z \sim \pi_{\text{RL}}(z \mid q)} [R_{\text{LC}}(f(x, z), y)]. \quad (\text{C.1})$$

Because R_{LC} is strictly proper (cf. §2.3) and assuming a sufficiently flexible LM π_{RL} and reader function $f : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$, our training procedure maximizes Eq. C.1 if and only if the user’s forecast exactly equals the ground-truth conditional distribution: $f(x, z) = p(y \mid x)$. Then, intuitively, making Bayes-optimal decisions according to the user forecast should be optimal in some sense. Zhao et al. (2021) make this notion of optimal decision-making concrete as **decision calibration**. To be self-contained, we will provide the definition of decision calibration from Zhao et al. (2021) below, and then use it to precisely describe the guarantees that linguistic calibration provides for decision-making.

C.3 Decision Calibration

Setup and notation. To define decision calibration, we first introduce some notation closely following Zhao et al. (2021), §2. In the process of decision-making based on probabilistic forecasts (ignoring LMs and their generations z for now), users receive a question $x' \in \mathcal{X}'$, forecast a possible answer to the question with a probabilistic forecaster $f : \mathcal{X}' \rightarrow \Delta^{|\mathcal{Y}|}$, and finally choose from a set of available actions \mathcal{A} based on their loss function $L : \mathcal{A} \times \mathcal{Y} \rightarrow \mathbb{R}$ and their forecast $f(x') \in \Delta^{|\mathcal{Y}|}$.

The process of choosing an action can be described by a **decision rule** which maps forecasts to actions $\delta : \Delta^{|\mathcal{Y}|} \rightarrow \mathcal{A}$, where Δ_{all} is the set of all decision rules. Additionally, define the set of all loss functions as $\mathcal{L}_{\text{all}} = \{L : \mathcal{A} \times \mathcal{Y} \rightarrow \mathbb{R}\}$.

We are interested in Bayes decision rules, *i.e.*, the set of rules that are optimal for *some* loss function. More specifically, consider some particular loss function $L \in \mathcal{L}_{\text{all}}$. Then its corresponding Bayes decision rule is

$$\delta_L(f(x')) = \arg \min_{a \in \mathcal{A}} \mathbb{E}_{\hat{y} \sim f(x')} [L(a, \hat{y})]. \quad (\text{C.2})$$

For some subset $\mathcal{L} \subset \mathcal{L}_{\text{all}}$, we denote the corresponding Bayes decision rules as $\Delta_{\mathcal{L}} := \{\delta_L \mid L \in \mathcal{L}\}$.

Decision calibration. Zhao et al. (2021) defines **decision calibration** to formalize the following intuition: ideally, a decision-maker with loss function L should be able to consider an arbitrary decision rule $\delta \in \Delta_{\text{all}}$ and compute the expected loss of using δ in decision-making, given a probabilistic forecaster f .

Definition C.2 (Decision Calibration, Definition 2 in Zhao et al. (2021)). For any set of loss functions $\mathcal{L} \subset \mathcal{L}_{\text{all}}$, we say that a forecaster f is \mathcal{L} -decision calibrated (with respect to the ground-truth conditional distribution $p(y \mid \cdot)$) if for each $L \in \mathcal{L}$ and $\delta \in \Delta_{\mathcal{L}}$,

$$\mathbb{E}_{x' \sim p(x')} \mathbb{E}_{\hat{y} \sim f(x')} [L(\delta(f(x')), \hat{y})] = \mathbb{E}_{x' \sim p(x')} \mathbb{E}_{y \sim p(y|x')} [L(\delta(f(x')), y)]. \quad (\text{C.3})$$

Following the analysis of Zhao et al. (2021) (§3.1), the left-hand side of Eq. C.3 *simulates* the loss of taking decisions according to the loss L and rule δ using data drawn from the forecast $\hat{y} \sim f(x')$. This simulated loss can be computed by a user without observing any ground-truth outcome $y \sim p(y \mid x')$. The right-hand side is the true loss for using decision rule δ with loss L . Therefore, decision calibration means that a forecaster f can be used to accurately estimate the expected loss of a decision rule under the true data distribution.

Decision calibration with LM assistance. With a few particular choices, we can apply this general definition to the setting of LM-assisted decision-making. In our context, the relevant variables are the open-ended query to the LM q , the LM response z , and the related question-answer pair (x, y) . These variables are distributed by the joint $p(q, x, y, z) := p(q, x, y) \pi_{\text{RL}}(z \mid q)$. Write the marginal over questions and related LM responses as $p(x, z) = \sum_{q', y'} p(q', x, y', z)$. In the notation of Definition C.2, we define the input $x' := (x, z)$.

To be concrete, we reiterate decision calibration in our context, where users also condition their forecasts on LM outputs.

Definition C.3 (Decision Calibration with LM Assistance). For any set of loss functions $\mathcal{L} \subset \mathcal{L}_{\text{all}}$, we say that a reader function $f : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$ is \mathcal{L} -decision calibrated (with respect to the ground-truth conditional distribution $p(y \mid \cdot)$) if for each $L \in \mathcal{L}$ and $\delta \in \Delta_{\mathcal{L}}$,

$$\mathbb{E}_{(x,z) \sim p(x,z)} \mathbb{E}_{\hat{y} \sim f(x,z)} [L(\delta(f(x,z)), \hat{y})] = \mathbb{E}_{(x,z) \sim p(x,z)} \mathbb{E}_{y \sim p(y|x)} [L(\delta(f(x,z)), y)].$$

Zhao et al. (2021) proves that if a forecaster f is decision calibrated, it provides the downstream decision-maker with two key guarantees. First, a decision-maker with loss L is assured that the Bayes decision rule δ_L outperforms alternative decision rules. Second, the decision-maker is able to accurately estimate the loss that they will incur by using the Bayes decision rule δ_L , despite not having access to ground-truth outcomes $y \sim p(y \mid x')$. In particular, we will show that decision calibration provides the following guarantees in our setting.

Lemma 1 (instantiation of Proposition 1 in Zhao et al. (2021)). *If a reader $f : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$ is \mathcal{L} -decision calibrated, then it satisfies:*

1. *No regret: for each $\delta' \in \Delta_{\mathcal{L}}$,*

$$\mathbb{E}_{(x,z) \sim p(x,z)} \mathbb{E}_{y \sim p(y|x)} [L(\delta_L(f(x,z)), y)] \leq \mathbb{E}_{(x,z) \sim p(x,z)} \mathbb{E}_{y \sim p(y|x)} [L(\delta'(f(x,z)), y)].$$

2. *Accurate loss estimation:*

$$\mathbb{E}_{(x,z) \sim p(x,z)} \mathbb{E}_{\hat{y} \sim f(x,z)} [L(\delta_L(f(x,z)), \hat{y})] = \mathbb{E}_{(x,z) \sim p(x,z)} \mathbb{E}_{y \sim p(y|x)} [L(\delta_L(f(x,z)), y)].$$

To summarize, **no regret** means that the Bayes decision rule δ_L performs no worse than any other decision rule $\delta' \in \Delta_{\mathcal{L}}$. **Accurate loss estimation** means that the decision-maker can determine before observing any ground-truth outcomes what their expected loss will be, allowing them to prepare for the future.

C.4 Linguistic Calibration and Optimal Decision-making

Now we can show that linguistic calibration implies the decision-making guarantees of Lemma 1. We do so using a key result from Zhao et al. (2021) (Theorem 1): standard notions of classifier calibration are equivalent to \mathcal{L} -decision calibration for particular choices of the loss family \mathcal{L} . We highlight two examples of this equivalence here and direct the interested reader to Zhao et al. (2021) for others.

Example 1: linguistic distribution calibration implies \mathcal{L}_{all} -decision calibration. Recall distribution calibration (cf. §2.1 for definition), the strongest standard notion of classifier calibration (Bröcker, 2009). Zhao et al. (2021) (Theorem 1) proves that distribution calibration is equivalent to \mathcal{L}_{all} -decision calibration. Suppose that we optimize the linguistic calibration objective R_{LC} in Eq. C.1 and obtain linguistic distribution calibration (for example, by maximizing R_{LC}). By definition, linguistic distribution calibration means that the reader f is distribution calibrated over $p(x, y, z)$. Then, applying Zhao et al. (2021) (Theorem 1), the reader is \mathcal{L}_{all} -decision calibrated, and therefore a Bayes-optimal decision-maker is guaranteed to perform no worse than *any* decision rule $\delta' \in \Delta_{\text{all}}$.

Unsurprisingly, this is very difficult to achieve in practice. However, even if linguistic calibration only manages to calibrate the reader in a weaker sense, we still have \mathcal{L} -decision calibration with respect to a smaller family of losses \mathcal{L} .

Example 2: linguistic confidence calibration implies \mathcal{L}_{r} -decision calibration. To illustrate this decision-making guarantee for weaker notions of calibration, recall confidence calibration (cf. §2.1, Table 4). Confidence calibration is a relaxation of the distribution calibration condition, only requiring it to hold for the most likely label. Zhao et al. (2021) (Theorem 1) demonstrates that confidence calibration is equivalent to \mathcal{L}_{r} -decision calibration, where

$$\mathcal{L}_{\text{r}} := \left\{ L(a, y) = \mathbb{1}[y \neq a \cap a \neq \perp] + \beta \cdot \mathbb{1}[a = \perp] \mid a \in \mathcal{Y} \cup \{\perp\}, \forall \beta \in [0, 1] \right\}. \quad (\text{C.4})$$

\mathcal{L}_{r} is the class of loss functions for the refrained prediction task, where the decision-maker can either take an action corresponding to an answer $y \in \mathcal{Y}$ or report “I don’t know” with the action \perp . Then, the decision-maker incurs a loss of 0 for correctly predicting the ground-truth answer y , a loss of 1 for predicting an incorrect answer, or a loss of $\beta < 1$ for reporting \perp . Similarly, Zhao et al. (2021) (Theorem 1) proves that classwise calibration (cf. §2.1) corresponds to another notion of \mathcal{L} -decision calibration.

Linguistic calibration implies no regret and accurate loss estimation guarantees. We conclude this section by applying the correspondence between classifier calibration and \mathcal{L} -decision calibration to prove decision-making guarantees for linguistic calibration. For each notion of classifier calibration $\phi \in \{\text{distribution, classwise, confidence}\}$, identify the corresponding class of decision calibration loss functions as \mathcal{L}_{ϕ} (e.g., distribution induces \mathcal{L}_{all} , and confidence induces \mathcal{L}_{r}). Then we have the following result:

Theorem C.4 (Linguistic ϕ -calibration implies no regret and accurate loss estimation guarantees). *Suppose that*

- An LM π_{RL} is linguistically ϕ -calibrated with respect to a user with reader function $f : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$, where $\phi \in \{\text{distribution, classwise, confidence}\}$ (cf. Definition 2.1).
- The user’s loss function $L : \mathcal{A} \times \mathcal{Y} \rightarrow \mathbb{R}$ is in the set of losses \mathcal{L}_{ϕ} induced by ϕ .
- The conditional independence assumption C.1 holds.

Then f is \mathcal{L}_{ϕ} -decision calibrated, and moreover, Lemma 1 (no regret and accurate loss estimation guarantees) holds for the user’s decision-making.

Proof. Linguistic ϕ -calibration is equivalent to ϕ -calibration of the reader f (cf. Definition 2.1). Recall that $p(x, z) = \sum_{q', y'} p(q', x, y', z)$ is the marginal over questions and related LM responses, and let $x' := (x, z)$ and $p(x') := p(x, z)$.

Applying Theorem 1 from Zhao et al. (2021) (equivalence of ϕ -calibration and \mathcal{L}_ϕ -decision calibration), we observe that the reader f is \mathcal{L}_ϕ -decision calibrated according to Definition C.2, Eq. C.3, and therefore the assumption of Lemma 1 holds.

We conclude by proving Lemma 1. Because the reader f is \mathcal{L}_ϕ -decision calibrated, we may apply Proposition 1 from Zhao et al. (2021), which provides the no regret and accurate loss estimation guarantees of decision calibration. Lastly, we invoke the conditional independence statement $p(Y | X, Z) = p(Y | X)$ (Assumption C.1) to simplify these guarantees, obtaining the guarantees in Lemma 1. □

As an example of this result, consider $\phi =$ confidence calibration. Suppose that our optimization of R_{LC} leads to linguistic confidence calibration, *i.e.*, a confidence calibrated reader f . Because of the equivalence of confidence calibration and \mathcal{L}_r -decision calibration, f is \mathcal{L}_r -decision calibrated and we obtain the corresponding guarantees for decision-making. For example, the no regret guarantee states that the Bayes decision rule δ_L outperforms decision rules $\delta' \in \Delta_{\mathcal{L}_r}$, where $\Delta_{\mathcal{L}_r}$ is a set of Bayes decision rules induced by \mathcal{L}_r . Intuitively, because confidence calibration is weaker than distribution calibration, $\Delta_{\mathcal{L}_r}$ is a more restricted set than Δ_{all} .

Summary: linguistic calibration provides optimal decision-making guarantees. If we maximize the linguistic calibration objective, our forecasts are distribution calibrated and therefore we obtain \mathcal{L}_{all} -decision calibration, *i.e.*, the Bayes optimal decision rule outperforms all other decision rules. Even if our optimization only achieves a weaker notion of calibration such as confidence calibration, we still obtain decision calibration over a more restricted class of loss functions. This fact also motivates our use of reader ECE as an evaluation metric, because it estimates confidence calibration and can therefore be thought of as a proxy for the quality of downstream user decision-making. For further discussion, we direct the reader to Zhao et al. (2021) and Zhao & Ermon (2021).

D Training Framework

Linguistic calibration is a general-purpose definition (Definition 2.1) that is agnostic to the particular design decisions we made in our proof-of-concept training framework. Nevertheless, this section details the design choices we made, including the algorithm and regularized objective function we used to train our neural net-based surrogate reader, and our implementation of PPO (Schulman et al., 2017).

D.1 Decision-Based RL Pseudocode

Algorithm 1: Decision-Based RL with a Surrogate Reader

Given: SFT LM π_{SFT} , LLM simulated reader $f_{\text{LLM}} : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$, joint distribution over open-ended queries and question-answer pairs $p(q, x, y)$.

Result: linguistically calibrated LM π_{LC} .

/ Construct datasets for training surrogate reader functions: ExtractAnswers (EA) and ForecastProbs (FP). */*

$\mathcal{D}_{\text{EA}} \leftarrow \{\}, \mathcal{D}_{\text{FP}} \leftarrow \{\}.$

for $i = 1, \dots, N$ **do**

/ LM-assisted user forecasting (§2.1). */*

Sample open-ended query and question-answer pair $(q, x, y) \sim p(q, x, y)$.

Sample long-form generation $z \sim \pi_{\text{SFT}}(z | q)$.

Apply LLM reader to obtain forecast $f_{\text{LLM}}(x, z) \in \Delta^{|\mathcal{Y}|}$.

/ Add ExtractAnswers example. */*

$\mathcal{D}_{\text{EA}} \leftarrow \mathcal{D}_{\text{EA}} \cup (x, z, \text{Support}(f_{\text{LLM}}(x, z)))$.

/ Add ForecastProbs example for ground-truth answer. */*

$\mathcal{D}_{\text{FP}} \leftarrow \mathcal{D}_{\text{FP}} \cup (x, z, y, f_{\text{LLM}}(x, z)_y)$.

/ Add ForecastProbs example for all extracted answers. */*

for *extracted answer* $y' \in \text{Support}(f_{\text{LLM}}(x, z))$ **do**

$\mathcal{D}_{\text{FP}} \leftarrow \mathcal{D}_{\text{FP}} \cup (x, z, y', f_{\text{LLM}}(x, z)_{y'})$.

end

end

/ Finetune surrogate LMs. */*

Finetune ExtractAnswers on \mathcal{D}_{EA} and ForecastProbs on \mathcal{D}_{FP} .

/ Optimize RL objective. */*

$\pi_{\text{LC}} \leftarrow$ PPO on π_{RL} initialized at parameters π_{SFT} , using surrogate reader \tilde{f} in KL-regularized objective

$$\max_{\pi_{\text{RL}}} \mathbb{E}_{(q, x, y) \sim p(q, x, y), z \sim \pi_{\text{RL}}(z | q)} \left[R_{\text{LC}}(\tilde{f}(x, z), y) - \beta \log \frac{\pi_{\text{RL}}(z | q)}{\pi_{\text{SFT}}(z | q)} \right].$$

return π_{LC}

D.2 Regularized Linguistic Calibration Objective

Objective regularized to encourage normalized surrogate forecasts. In our instantiation of linguistic calibration, we use a surrogate forecast $\tilde{f}(x, z)$ which is not guaranteed to be normalized: we may have $\tilde{f}(x, z) \in [0, 1]^{|\mathcal{Y}|} \setminus \Delta^{|\mathcal{Y}|}$. We remedy this by adding a regularization term in the reward function which encourages $\tilde{f}(x, z)$ to be normalized, restoring strict propriety (cf. §2.3) even with forecasts in $[0, 1]^{|\mathcal{Y}|}$. We indeed find in practice that training with this objective results in normalized forecasts.

Our regularized reward function $R_{\text{LC}} : [0, 1]^{|\mathcal{Y}|} \times \mathcal{Y} \rightarrow \mathbb{R}$ is written as:

$$R_{\text{LC}}(\tilde{f}(x, z), y) = \log \tilde{f}(x, z)_y - \lambda \left| 1 - \sum_{y' \in \mathcal{Y}} \tilde{f}(x, z)_{y'} \right| + C, \quad (\text{D.5})$$

where $\lambda > 0, C > 0$ are hyperparameters. C is a small positive constant which we find to reduce reward hacking during PPO.

Below, we prove that with $\lambda > 1$, we indeed have a strictly proper objective.

D.3 Proof: Regularized Objective is Strictly Proper

Setup. In decision-based RL, we wish to optimize an objective that encourages the surrogate forecast $\tilde{f}(x, z)$ to exactly match the ground-truth distribution $p(y | x)$, which is supported over all possible answers \mathcal{Y} . $\tilde{f}(x, z)$ is not necessarily in $\Delta^{|\mathcal{Y}|}$ because it is obtained by several independent neural network predictions falling in the range $[0, 1]$, *i.e.*, $\tilde{f}(x, z) \in [0, 1]^{|\mathcal{Y}|}$. Therefore, in addition to a log loss term, we also add a regularizer which encourages $\tilde{f}(x, z)$ to fall in the simplex, *i.e.*, be normalized.

We would like our reward function to be strictly proper—its unique maximizer should be the ground-truth answer distribution. We will prove that our reward function is indeed strictly proper below. First, we define (strictly) proper scoring rules in our setting, which is *slightly more general than the standard definition* (e.g., in Gneiting & Raftery (2007)) in that the first argument need not be a probability distribution: it may fall in $[0, 1]^{|\mathcal{Y}|} \setminus \Delta^{|\mathcal{Y}|}$. As shorthand throughout this proof, we will refer to the ground-truth answer distribution as $p^* := p(y | x)$ and the forecast as $\tilde{p} := \tilde{f}(x, z)$, because this proof is done pointwise for a given (question, LM generation) tuple (x, z) .

Definition D.1 (based on Gneiting & Raftery (2007)). Let $\bar{\mathbb{R}}$ be the extended real line $[-\infty, \infty]$. A scoring rule $R : [0, 1]^{|\mathcal{Y}|} \times \mathcal{Y} \rightarrow \bar{\mathbb{R}}$ is **proper** if

$$\mathbb{E}_{Y \sim p^*} [R(p^*, Y)] \geq \mathbb{E}_{Y \sim p^*} [R(\tilde{p}, Y)], \quad \forall p^* \in \Delta^{|\mathcal{Y}|}, \tilde{p} \in [0, 1]^{|\mathcal{Y}|}. \quad (\text{D.6})$$

It is **strictly proper** if Eq. D.6 holds with equality if and only if $p^* = \tilde{p}$.

Equivalently, we may express the strictly proper condition as

$$\mathbb{E}_{Y \sim p^*} [R(p^*, Y)] > \mathbb{E}_{Y \sim p^*} [R(\tilde{p}, Y)], \quad \forall p^* \in \Delta^{|\mathcal{Y}|}, \tilde{p} \in [0, 1]^{|\mathcal{Y}|}, \tilde{p} \neq p^*. \quad (\text{D.7})$$

In proving that R_{LC} is strictly proper, we will also make use of the following standard result:

Lemma 2 (e.g., Gneiting & Raftery (2007)). *The logarithmic scoring rule $R_{\log} : \Delta^{|\mathcal{Y}|} \times \mathcal{Y} \rightarrow \{-\infty\} \cup \mathbb{R}$ is strictly proper.*

Lastly, we will also use the log-sum inequality:

Lemma 3 (e.g., p. 29, Cover & Thomas (1991)). *Let a_1, \dots, a_n and b_1, \dots, b_n be nonnegative numbers. Denote $a := \sum_{i=1}^n a_i$ and $b := \sum_{i=1}^n b_i$. Then*

$$\sum_{i=1}^n a_i \log \frac{a_i}{b_i} \geq a \log \frac{a}{b}.$$

We will now prove that our regularized training framework objective R_{LC} is a strictly proper scoring rule using the condition for strict propriety in Eq. D.7.

Theorem D.2. *For any $C \in \mathbb{R}$ and with $\lambda > 1$, the reward function $R_{\text{LC}}(\tilde{p}, y)$ from Eq. D.5 is strictly proper.*

Proof. Take an arbitrary $p^* \in \Delta^{|\mathcal{Y}|}$ and $\tilde{p} \in [0, 1]^{|\mathcal{Y}|}$ such that $\tilde{p} \neq p^*$. We need to show that

$$\mathbb{E}_{Y \sim p^*} [R_{\text{LC}}(p^*, Y)] - \mathbb{E}_{Y \sim p^*} [R_{\text{LC}}(\tilde{p}, Y)] > 0. \quad (\text{D.8})$$

All expectations for the rest of the proof are taken with respect to p^* , so we omit the subscript $Y \sim p^*$. As in the main text, we identify the space of answers \mathcal{Y} with the index set $[|\mathcal{Y}|]$, and therefore use notation p_y to refer to the y -th index of the vector p . Lastly, we write $\sum_{y' \in \mathcal{Y}} \tilde{p}_{y'}$ as $\mathbf{1}^\top \tilde{p}$ for brevity.

Plugging in with our reward function, Inequality D.8 is equivalent to

$$\mathbb{E} [\log p_Y^* - \lambda |1 - \mathbf{1}^\top p^*| + C] - \mathbb{E} [\log \tilde{p}_Y - \lambda |1 - \mathbf{1}^\top \tilde{p}| + C] > 0.$$

We can simplify the LHS by applying linearity of expectation and simplifying expectations of constants:

$$\text{LHS} = \mathbb{E}[\log p_Y^*] - \lambda|1 - \mathbf{1}^\top p^*| - \mathbb{E}[\log \tilde{p}_Y] + \lambda|1 - \mathbf{1}^\top \tilde{p}|.$$

Note that we assume $p^* \in \Delta^{|\mathcal{Y}|}$ so $\mathbf{1}^\top p^* = 1$, and we can simplify further to

$$\text{LHS} = \mathbb{E}[\log p_Y^*] - \mathbb{E}[\log \tilde{p}_Y] + \lambda|1 - \mathbf{1}^\top \tilde{p}|. \quad (\text{D.9})$$

Now we proceed by cases on whether or not $\tilde{p} \in \Delta^{|\mathcal{Y}|}$. Suppose that indeed $\tilde{p} \in \Delta^{|\mathcal{Y}|}$. Then its normalization term $\lambda|1 - \mathbf{1}^\top \tilde{p}|$ is also 0, and therefore this inequality condition is precisely the strict propriety condition for the logarithmic scoring rule (including the assumption that both p^* and \tilde{p} are in the simplex):

$$\mathbb{E}[\log p_Y^*] - \mathbb{E}[\log \tilde{p}_Y] > 0,$$

which is true according to Lemma 2, and therefore this case is done.

So now assume that $\tilde{p} \in [0, 1]^{|\mathcal{Y}|} \setminus \Delta^{|\mathcal{Y}|}$. This implies that its normalization term is greater than 0, *i.e.*, $|1 - \mathbf{1}^\top \tilde{p}| > 0$. In this case, we can prove using the properties of log and the log-sum inequality that the LHS expression in Eq. D.9 is greater than 0, in order to complete the proof.

First, we use the properties of log to rewrite the expectations in Eq. D.9 into the entropy of p^* plus a relative entropy between p^* and \tilde{p} :

$$\begin{aligned} \text{LHS} &= \mathbb{E}[\log p_Y^*] - \mathbb{E}[\log \tilde{p}_Y] + \lambda|1 - \mathbf{1}^\top \tilde{p}| \\ &= -\mathbb{H}[p^*] - \mathbb{E}[\log \tilde{p}_Y] + \lambda|1 - \mathbf{1}^\top \tilde{p}| && \text{(definition of entropy functional } \mathbb{H}[\cdot]) \\ &= -\mathbb{H}[p^*] + \mathbb{H}[p^*] + \sum_{y \in \mathcal{Y}} p_y^* \log \frac{p_y^*}{\tilde{p}_y} + \lambda|1 - \mathbf{1}^\top \tilde{p}| && \text{(decomposition of cross-entropy)} \\ &= \sum_{y \in \mathcal{Y}} p_y^* \log \frac{p_y^*}{\tilde{p}_y} + \lambda|1 - \mathbf{1}^\top \tilde{p}|, \end{aligned}$$

where the penultimate step used the decomposition of cross-entropy = entropy + relative entropy.

We cannot immediately apply the information inequality to show that the relative entropy is nonnegative, because \tilde{p} is not a probability distribution (and therefore the KL Divergence and relative entropy are not the same; see, *e.g.*, p. 90 in [Boyd & Vandenberghe \(2004\)](#)). However, we may use the log-sum inequality to finish the proof. Applying Lemma 3 using the fact that all elements of p^* and \tilde{p} are nonnegative, we can lower bound the relative entropy term as

$$\sum_{y \in \mathcal{Y}} p_y^* \log \frac{p_y^*}{\tilde{p}_y} \geq \mathbf{1}^\top p^* \log \frac{\mathbf{1}^\top p^*}{\mathbf{1}^\top \tilde{p}} = \log \frac{1}{\mathbf{1}^\top \tilde{p}}.$$

Altogether, substituting normalizer $\mathcal{Z} := \mathbf{1}^\top \tilde{p}$, we currently have a lower bound of the LHS in Eq. D.9 of

$$\text{LHS} \geq \log \frac{1}{\mathcal{Z}} + \lambda|1 - \mathcal{Z}|. \quad (\text{D.10})$$

Inspecting this function for $\lambda > 1$, we can see that it is positive for all values of \mathcal{Z} other than $\mathcal{Z} = 1$. To be rigorous, we complete the proof with cases on the values of \mathcal{Z} . Because $\tilde{p} \in [0, 1]^{|\mathcal{Y}|}$, \mathcal{Z} is nonnegative, and recall that we already covered the case when $\mathcal{Z} = \mathbf{1}^\top \tilde{p} = 1$ by reduction to strict propriety of the logarithmic scoring rule (Lemma 2).

When $\mathcal{Z} \in [0, 1)$, we immediately see that $\log \frac{1}{\mathcal{Z}} > 0$ and $\lambda|1 - \mathcal{Z}| > 0$ (since $\lambda > 1$ by assumption), implying that the lower bound in Eq. D.10 is positive as we needed to show.

Lastly, suppose that $\mathcal{Z} \in [1, \infty)$. In that case, $\lambda|1 - \mathcal{Z}| = \lambda(\mathcal{Z} - 1)$. Now, note that the derivative of the lower bound in Eq. D.10 is positive for all $\mathcal{Z} \geq 1$ (recalling that $\lambda > 1$):

$$\frac{d}{d\mathcal{Z}} \left[\log \frac{1}{\mathcal{Z}} + \lambda(\mathcal{Z} - 1) \right] = -\frac{1}{\mathcal{Z}} + \lambda > -\frac{1}{\mathcal{Z}} + 1,$$

e.g., at $\mathcal{Z} = 1$, we have that the derivative is greater than $-\frac{1}{1} + 1 = 0$. And further, at precisely $\mathcal{Z} = 1$, the lower bound equals $\log \frac{1}{1} + \lambda|1 - 1| = 0$. This proves that the lower bound is positive for all $\mathcal{Z} \in (1, \infty)$, which implies that the LHS is also positive for $\mathcal{Z} \in (1, \infty)$. We have covered all values of \mathcal{Z} , concluding that when $\lambda > 1$, the reward function in Eq. D.5 is strictly proper. \square

D.4 Additional Details on Training Framework

We provide additional details on the training framework below. For other details, we direct the reader to the codebase.³

Training and Validation Splits. Our training framework and all baselines use examples from the TriviaQA (Joshi et al., 2017) `unfiltered.nocontext` subset on HuggingFace Datasets (Lhoest et al., 2021). These examples are randomly assigned to the following splits:

- **SFT** (10000 examples): used for summary distillation and the other SFT baselines (Factuality SFT, Claude Distill).
- **Prompt Validation** (1000 examples): used for all ICL-based baselines and to construct ICL examples for the simulated reader, which uses an API-based LLM.
- **Reward Model** (20000 examples): used to train surrogate reader for LC, and binary reward model for Factuality RL baseline.
- **PPO** (40000 examples): used for PPO with LC RL and Factuality RL methods.
- **PPO Validation** (1000 examples): during PPO, we evaluate reward model rewards on this split and store checkpoints every 20 steps.
- **Validation** (1000 examples): used for tuning evaluation temperature and model selection for RL methods (described below).

Summary distillation details. For each SFT dataset query q , we use the (non-confidence) ICL baseline to sample eight long-form generations $\{z_i\}_{i=1}^8 \sim \pi_{\text{ICL}}(z | q)$ (temperature 0.7) that are then summarized using a Claude few-shot prompt $s \sim \text{Summarize}(s | z_1, \dots, z_8)$ (temperature 0.3). We use the dataset of (q, s) pairs to finetune the base Llama 2 7B model, obtaining π_{SFT} . The same temperature settings are used for obtaining Summary ICL context examples, and for the Direct Summary Eval oracle baseline.

Surrogate and reward model details. Following our evaluation protocol for user forecasts (§4.1), we specify prompt instructions so that LLM simulated forecasts $f_{\text{LLM}}(x, z)$ used in Algorithm 1 avoid the use of background knowledge. Dataset construction for the surrogate reader models ForecastProbs and ExtractAnswers is described in Algorithm 1.

For LC RL, we use the following training details:

- **ForecastProbs:** a Llama 2 7B-based model initialized with the LC SFT parameters and trained using a binary cross-entropy loss on prompts from the Reward Model split.
- **ExtractAnswers:** a RedPajama 3B-based model initialized from the pretrained model weights and trained with causal language modeling on targets formatted as a delimited list of possible answers, *e.g.*, “Sushi; Sashimi; Nigiri”. We used a 3B model instead of Llama 2 7B due to computational constraints, *i.e.*, to fit all standard PPO models such as the policy, reference policy, value model, and reward models into GPU memory on an 8x80GB A100 node.

For Factuality RL, we use the following training details:

- **Reward Model:** a Llama 2 7B-based model initialized with the Factuality SFT parameters and trained using binary cross-entropy loss on prompts from the Reward Model split, with binary correctness labels from Claude 2.0. In producing binary correctness labels for a given (question, ground-truth answer, generation) tuple (x, y, z) , Claude is given access to the ground-truth answer. Therefore, determining binary correctness is a simple extractive task that we find Claude performs with very high accuracy.

We tune all reward models over a wide range of learning rates; see codebase for details on hyperparameters.

³https://github.com/tatsu-lab/linguistic_calibration

PPO implementation and training. We use a standard implementation of PPO from [Dubois et al. \(2023\)](#) and train with 8 80GB A100 GPUs using Flash Attention 2 ([Dao et al., 2022](#); [Dao, 2023](#)) and PyTorch FSDP ([Zhao et al., 2023](#)). For both PPO-based methods (LC RL and Factuality RL), we use default hyperparameters other than the following:

- We use a larger step batch size (512) with one optimization epoch per step for better training stability.
- We shorten `query_len` to 128 tokens, because our PPO inputs were essentially all under this length.
- We use a slightly lower temperature during the rollout phase (0.7 instead of 1.0).
- We use the `paged_adamw_8bit` optimizer ([Detmeters et al., 2022](#)) due to computational constraints. We also use this optimizer for SFT and reward model training for consistency, and find it had no negative impact on validation performance compared to AdamW ([Loshchilov & Hutter, 2019](#)).
- We train for 1500 PPO steps.

We tune the KL penalty and learning rate of both PPO methods across a wide range, ultimately finding that `kl_coef` of 0.1 and learning rate of $1e-5$ works best for both methods.

For the LC RL objective, we find that $\lambda = 5$, $C = 5$ works well to enforce normalization of downstream forecasts and prevent reward hacking. In the log-loss term, we clip the probability of the ground-truth answer at $\epsilon = 1e-4$ for numerical stability.

Model selection for RL methods. We perform model selection for PPO-based methods (LC RL, Factuality RL) by taking the 5 PPO checkpoints with the highest reward model rewards on the PPO Validation split, evaluating them on the Validation split with the API-based LLM simulated reader, and selecting the checkpoint with the best ECE.

E Evaluation Framework

Test Dataset Splits. We use the following test dataset splits.

- **TriviaQA** (Joshi et al., 2017) (11313 examples): take the validation split from the `unfiltered.nocontext` subset from HuggingFace Datasets (Wolf et al., 2020; Lhoest et al., 2021).
- **Jeopardy** (Kaggle, 2020) (10638 examples): take a random sample of the `train` subset from HuggingFace Datasets. These questions are initially written in the indirect Jeopardy-style format (“For the last 8 years of his life, Galileo was under house arrest for espousing this man’s theory”), so we use Claude 2.0 to convert them into direct questions x .
- **SciQ** (Welbl et al., 2017) (13679 QA examples): combine train, validation, and test splits from `allenai/sciq` on HuggingFace Datasets. SciQ is a dataset of crowdsourced science exam questions in biology, chemistry, earth science, and physics, spanning elementary level to college introductory material.
- **BioASQ** (Krithara et al., 2023) (1515 QA examples): we use the 2024 BioASQ12 Task B dataset, which is a biomedical QA dataset annotated by a team of experts. Specifically, we use their 1515 “factoid” examples, which have a short-answer gold label (e.g., “proprotein convertase subtilisin/kex in type 9”). The dataset is available at <http://participants-area.bioasq.org/datasets>.
- **Person Biography Generation** (Min et al., 2023) (500 entities): use the `unlabeled` split of Wikipedia entities from the FactScore dataset.

TriviaQA, Jeopardy, SciQ, and BioASQ are question-answering datasets which we repurpose for long-form evaluation. Specifically, for these question-answering datasets, we use an API-based LLM (Claude 2.0) to convert questions x into open-ended queries q that allow us to sample long-form generations $z \sim \pi(z | q)$ on the topic of the question x (as we do at training time, cf. §3.1). For Person Biography Generation, we construct open-ended queries using the fixed template “Write a paragraph bio about {entity}.”.

Reader expected calibration error (reader ECE). The reader ECE metric is precisely the expected calibration metric (Guo et al., 2017) applied to a reader $f : \mathcal{X} \times \mathcal{Z} \rightarrow \Delta^{|\mathcal{Y}|}$. Given N question-answer pairs $\{(x^{(i)}, y^{(i)})\}_{i=1}^N$ and corresponding long-form generations $\{z^{(i)}\}_{i=1}^N$, we partition them into M bins by max forecast probability $\max_y f(x^{(i)}, z^{(i)})_y$. ECE is then expressed as

$$\text{ECE} = \sum_{j=1}^M \frac{|B_j|}{N} |\text{Acc}(B_j) - \text{Conf}(B_j)|, \quad \text{where}$$

$$\text{Conf}(B_j) = \frac{1}{|B_j|} \sum_{i \in B_j} \max_y f(x^{(i)}, z^{(i)})_y, \quad \text{Acc}(B_j) = \frac{1}{|B_j|} \sum_{i \in B_j} \mathbb{1}[y^{(i)} = \arg \max_y f(x^{(i)}, z^{(i)})_y].$$

We set the number of bins as $M = 20$ on simulated QA evaluations, and $M = 10$ on all others. Lastly, we note that log loss is not a reasonable evaluation metric in our setting because simulated and human readers can assign zero probability to the ground-truth class label resulting in infinite log loss.

Confidence intervals. We report bootstrap 0.95 confidence intervals in all frontier plots and tabular results, except for the BioASQ dataset. BioASQ is a relatively small dataset (1515 examples; it is expert-written) and therefore we report Student’s t distribution 0.95 confidence intervals over 10 evaluation seeds. In reliability diagrams for BioASQ, we report the reliability diagram computed over samples pooled across all seeds in blue, and per-seed reliability diagrams in gray.

E.1 Simulated Evaluation

Simulated reader. Simulated forecasts $f(x, z)$ are a distribution over possible answers to the question x , based on long-form LM generations z . All simulated forecasts in this paper are formed using an API-based LLM simulated reader f_{LLM} . We use Claude due to compute constraints, but other LLMs such as GPT-4 are well-capable of producing high-quality forecasts. These simulated forecasts are used in two parts of our training and evaluation frameworks:

- **Training framework:** simulated forecasts are used as ground-truth targets for training the surrogate reader \tilde{f} (cf. Algorithm 1). Then the surrogate reader is used during PPO, enabling efficient training by avoiding calls to API-based LLMs.
- **Evaluation framework:** in our question-answering evaluations, we sample a held-out question-answer pair $(x, y) \sim p(x, y)$, convert x into open-ended query q , and evaluate the accuracy and ECE of the simulated forecast $f_{\text{LLM}}(x, z)$ for $z \sim \pi(z | q)$ sampled from a variety of LMs such as LC RL and Factuality RL.

In both cases, we obtain simulated forecasts by decomposing reading into two prompts, one for ExtractAnswers and one for ForecastProbs, following the description in §3. This is just one way of implementing a reader, and there are likely other reasonable ways to obtain high-quality probabilistic predictions conditioned on long-form text.

In order to compute accuracy and ECE for the evaluation framework, we need to obtain the confidence and correctness of each forecast. We direct the reader to our codebase⁴ for all prompts.

Accuracy and ECE for confidence-based methods. The confidence of a forecast is obtained by simply taking its argmax probability $\max_{y \in \mathcal{Y}} f(x, z)_y$. Recall that for question-answering evaluation datasets, we prompt an API-based LLM to perform the ForecastProbs operation. We find a zero-shot prompt is effective because confidence statements in z associated with the question x are usually numerical (and even when they are occasionally linguistic, Claude 2.0 produces reasonable confidence estimations). The correctness is obtained by determining whether the top answer in the forecast is correct, *i.e.*, $\mathbb{1}[y = \arg \max_{y' \in \mathcal{Y}} f(x, z)_{y'}]$. In practice, we perform this check by prompting Claude with the question x , ground-truth answer choice y , and the argmax answer choice, and asking the LLM to decide whether the two answers are semantically equivalent, using a few-shot prompt. For TriviaQA and Jeopardy, which are general-domain knowledge tasks, `claude-2.0` can perform this semantic equivalence check reliably. For the domain-specific tasks SciQ and BioASQ, we use the more powerful `claude-3-opus-20240229` to perform this semantic equivalence check, because the ground-truth and argmax answers are more complicated; *e.g.*, $y =$ “hormone receptor-positive, human epidermal growth factor receptor 2-negative”, $\text{argmax} =$ “hormone-receptor-positive, HER2-negative breast cancer patients”. This semantic equivalence check avoids the false negative issues that string match-based metrics encounter due to paraphrases, which are very common in our open-ended, long-form generation setting.

Accuracy and ECE for non-confidence methods. Our evaluation for non-confidence methods closely follows the pipeline for confidence-based methods. We extract answers using the ExtractAnswers prompt, which accounts for the possibility that models may provide multiple mutually exclusive answers to the question x . We select the “argmax” answer by taking the answer which appears first in the paragraph z . Because non-confidence methods are not prompted or trained to produce confidence statements, we fix their confidence values to 1. Then, we compute ECE for non-confidence methods in closed form as $(1 - \text{Accuracy})$. This approach is validated by our human evaluations, in which the correlation between a fixed confidence value of 1 and the human-interpreted argmax probability of Factuality RL is 0.993, over 1000+ samples (cf. Table 7).

Temperature tuning. We tuned the decoding temperature for generating paragraphs across the range $\{0.1, 0.3, 0.5, 0.7\}$ for the core methods (LC RL, LC SFT, Factuality RL, Factuality SFT) on the TriviaQA Validation split with simulated reader evaluation. We found that temperature 0.3 uniformly performed the best across methods. Therefore we defaulted to paragraph decoding temperature 0.3 for evaluation throughout. We found that temperature 0.2 consistently worked well for simulated reader operations ExtractAnswers, ForecastProbs, and for checking semantic equivalence.

Other evaluation details. Because we prompt our LMs with open-ended queries ($q :=$ “Write a paragraph about $\{x\}$ ”) it is rarely possible that a paragraph z will not provide information that answers question x . We account for this during evaluation by allowing the ExtractAnswers prompt to return an empty list if z does not provide any plausible answers to the question. For any example with an empty extracted list, we consider the accuracy to be 0 and the confidence to be 1, which is the most conservative possible penalty for both accuracy and ECE. Because we significantly outperform baselines finetuned with RL for factuality, we account for the possibility of better responsiveness to the prompt (and therefore fewer empty ExtractAnswers lists) explaining our improvements.

⁴https://github.com/tatsu-lab/linguistic_calibration

E.2 FactScore-Based Evaluation Metric

We use an automated FactScore-based (Min et al., 2023) metric using an API-based LLM (Claude 2.0 in our case) to test the accuracy and calibration of generated biographies z at the per-claim level, probing for the quality of confidences throughout long-form generations. §4.3 describes this metric in detail. Prompts and code for this metric are available in our codebase.

Forecasting confidences. As part of this metric, for confidence-based methods, we need to forecast probabilities which are used to compute ECE. As compared to the question-answering evaluations, this evaluation considers all claims in the generated paragraph and tends to have a higher proportion of claims with linguistic confidence statements. Therefore, we use the following procedure to assign probabilities to each claim. When a claim has a clear numerical uncertainty (*e.g.*, a percentage) this is a simple extractive task which API-based LLMs perform well by default. For interpreting linguistic uncertainties, we provide the LLM with a mapping between linguistic phrases and probabilities collected in a linguistics human study (Wallsten, 1990), and allow it to generalize from this mapping to phrases that are not present in it (Table 5). We believe that developing better strategies to interpret linguistic confidence statements in a manner tailored to a given user

Linguistic Confidence Phrase	Probability
Almost Impossible	0.05
Doubtful	0.1
Improbable	0.1
Unlikely	0.15
Possible	0.3
Tossup	0.5
Good Chance	0.65
Likely	0.75
Probable	0.75
Almost Certain	0.95

Table 5: Ground-truth linguistic confidence mapping used in our FactScore probability interpretation prompt, from Wallsten (1990).

population is exciting future work (discussed in §6).

E.3 Human Evaluation

Task. In order to compute our accuracy and ECE metrics using forecasts from human crowdworkers, we asked crowdworkers to provide the following for a given question, ground-truth answer, and long-form generation tuple (x, y, z) :

- **Confidence:** the max probability of their forecast distribution $\max_{y \in \mathcal{Y}} f(x, z)_y$.
- **Correctness:** a binary indicator of whether or not their top answer was correct, $\mathbb{1}[y = \arg \max_{y' \in \mathcal{Y}} f(x, z)_{y'}]$.

We asked human crowdworkers to interpret the generated paragraphs literally, and provided a couple examples. These examples were intentionally selected to be unambiguous, *i.e.*, numerical confidences, to ensure we were not biasing towards any particular manner of interpreting linguistic confidences.

Qualification. We conducted two qualification rounds with our human annotators for the question-answering TriviaQA experiment. In the first round, we sent a single question-answer pair (x, y) sampled from TriviaQA, along with 3 model outputs (from LC RL, LC SFT, and Factuality RL) to 117 workers, out of which 32 matched the expected output. This example was selected to be unambiguous. In the second round, we sent the 32 workers who passed the previous test a roughly hour-long test (examples also selected to be unambiguous; the student authors of this paper agreed on all confidence and correctness labels of the hour-long test). Among these annotators, we took the top 15 performers for the main annotation process.

Main annotation round. During the main annotation process, we simply selected examples uniformly at random from the TriviaQA test set, including examples which were potentially ambiguous due to linguistic confidence statements. Overall, we collected annotations for 1008 examples (each with 3 model outputs) from human crowdworkers.

Quality assurance. We compared crowdworker agreement with Claude 2.0 as a quality assurance measure. Five of the annotators had agreement rates which dropped significantly after the qualification round, from perfect performance on the hour-long test to less than 0.40 Cohen’s kappa agreement with Claude. Therefore, we discontinued working with these annotators and removed their labels.

Payment. To determine payment, the authors of this paper timed our own completion time of the task, which was an average of approximately 1.3 minutes. Following best practices for crowdworker payment (Whiting et al., 2019; Liang et al., 2023), we used a conservative estimate of annotation time of 3 minutes per task and set the per-task payment to \$0.85 amounting to a compensation of \$17 per hour, greater than the California minimum wage of \$16 per hour.

Agreement statistics. In Table 6 we report interannotator agreement (*i.e.*, agreement between crowdworkers) computed over 5% of the collected samples, for each of the tested models.

Model	Correlation on Confidence	Cohen’s Kappa on Correctness
LC RL	0.886	0.850
LC SFT	0.719	0.842
Factuality RL	1.000	0.758

Table 6: Interannotator Agreement.

In Table 7, we report human agreement with Claude 2.0 over all examples.

Model	Correlation on Confidence	Cohen’s Kappa on Correctness
LC RL	0.626	0.739
LC SFT	0.618	0.748
Factuality RL	0.993	0.741

Table 7: Human-Claude 2.0 Agreement.