Aligning LLMs with Human Uncertainty: A Beta-Bernoulli Calibrator for LLM Forecasting

Hui Dai [1 2] **Ryan Teehan** [1] **Parsa Torabian** [3] **Mengye Ren** [1]

Abstract

Probabilistic forecasting estimates the likelihood of uncertain future events. To improve LLM forecasting, existing methods typically learn from binary outcomes to output verbalized forecasts. However, while aggregated human forecasts contain rich information in both the crowd probability estimate and the degree of agreement among forecasters, how to utilize these signals remains underexplored. To address this, we propose the Beta-Bernoulli Calibrator (BBC), which converts an initial point estimate forecast from any model into a distribution over event likelihood, using supervision from both binary outcomes and human forecasts. BBC models event likelihood $p \sim \text{Beta}(\alpha, \beta)$ and outcome $y \sim \text{Bernoulli}(p)$, with the mean as the calibrated point forecast and the variance as the epistemic uncertainty. Our results show that BBC generally provides better calibrated and more accurate forecasts than both traditional post-hoc calibration methods and models fine-tuned specifically for forecasting, while remaining lightweight and having good generalization. We also show that the epistemic uncertainty captured by BBC is a more reliable predictor of forecasting error than verbalized confidence.

1. Introduction

Making predictions about the future is an integral part of everyday decision-making. Individuals check weather forecasts to adjust travel plans, companies calculate the odds of a product's success, and governments shape policy around economic and national security forecasts (Lahiri & Yang, 2013; Tetlock & Gardner, 2016). Given large language models' (LLMs) broad knowledge and reasoning capabilities, there is an increasing interest in using LLMs for forecasting, typically by prompting the model to output a verbalized estimate of the likelihood an event will occur (Karger et al., 2024; Zeng et al., 2025; Yang et al., 2026). However, even state-of-the-art models struggle to outperform skilled human forecasters (Karger et al., 2024).

To improve forecasting capabilities, prior work has investigated supervised fine-tuning via distillation on subsets where the model outperforms humans (Halawi et al., 2024), as well as reinforcement learning (RL) using signals from realized outcomes (Chandak et al., 2025; Turtel et al., 2026). However, these approaches are resource-intensive and typically cannot be applied to black-box models. Moreover, human forecasts contain rich information about human sentiment and uncertainty, as they capture both the aggregate estimate of an event's likelihood as well as the amount of consensus among the pool of forecasters. In spite of this, incorporating this information remains underexplored, and current methods do not capture the degree of consensus among the human forecasters.[1] In this work, we ask: *beyond eliciting verbalized probabilities, how can we calibrate model forecasts using supervision from both binary outcomes and human forecasts?*

As illustrated in Figure 2, we propose a Beta-Bernoulli framework in which the event probability p is modeled as a Beta distribution, $p \sim \text{Beta}(\alpha, \beta)$, and the observed outcome y is a realization from a Bernoulli trial, $y \sim \text{Bernoulli}(p)$. Taking the forecasting question and an initial verbalized forecast as input, the calibrator outputs the Beta parameters. Notably, this framework is model-agnostic. It is implemented using a small, open-source language model (the calibrator) to refine the initial textual forecast provided by a separate input LLM. This allows us to calibrate any input LLM's beliefs without access to its internal representations. This ensures universal applicability and reduces training overhead. When learning from only binary outcomes, we show that the Beta–Bernoulli objective reduces to binary cross-entropy (BCE), a proper scoring rule that incentivizes truthful probability estimation. To incorporate signals beyond the binary outcome, we use human forecasts as distributional supervision for the Beta distribution. This enables the model to represent both the predicted event prob-

[1]New York University, NY, USA [2]The University of Chicago, IL, USA [3]Chronologies AI. Correspondence to: Hui Dai <hd2584@nyu.edu>, Mengye Ren <mengye@nyu.edu>.

Accepted at the ICML 2026 Workshop on Forecasting as a New Frontier of Intelligence.

[1]See Appendix B for more related work discussion.

ability via the Beta mean, and epistemic uncertainty about that probability via the Beta variance.

We evaluate our framework on data from prediction platforms Metaculus and Polymarket. Compared to uncertainty estimation and post-hoc calibration methods, our Beta-Bernoulli Calibrator (BBC) generally provides better-calibrated forecasts with stronger discrimination performance. We find that utilizing human forecasts as auxiliary supervision consistently improves discrimination compared to training only on binary outcomes. Moreover, this lightweight post-hoc adjustment even outperforms models that are fine-tuned specifically for forecasting, and provides further improvements when applied to them. In addition, we validate that BBC's epistemic uncertainty is a strong predictor of forecasting errors, while verbalized confidence is a noisier signal. Finally, we test our calibrator's generalization on the external Kalshi dataset, and observe consistent performance gain. Code is available at `https://agenticlearning.ai/beta-bernoulli-calibrator`.

2. Beta-Bernoulli Calibrator

2.1. Preliminaries

We study the task of probabilistic forecasting for binary events (Lahiri & Yang, 2013). Let $D = \{(x_i, y_i, \mathbf{q}_i)\}_{i=1}^N$ be a dataset of N binary forecasting questions, where x_i is the textual description of event i (e.g., a question and its resolution criteria), and $y_i \in \{0, 1\}$ denotes the binary outcome. In addition, we observe k_i human forecasts for each event $\mathbf{q}_i = \{q_{i1}, \dots, q_{ik_i}\}$, where $q_{ij} \in [0, 1]$ is the probability estimate provided by forecaster j. We assume $p_i^\star = P(y_i = 1 \mid x_i)$ is the unobservable ground-truth event probability. Our goal is to learn a model f_θ that takes in x_i and outputs a probability forecast $\hat{p}_i \in [0, 1]$, such that $\hat{p}_i \approx p_i^*$. Note that prior work typically extracts $\hat{p}_i$ from verbalized output, e.g. set $\hat{p} = 0.2$ if the model outputs "I estimate a 20% chance". In contrast, we model p_i as a distribution and later report the mean as $\hat{p}_i = \mathbb{E}[p_i]$.

2.2. Overview

The outcome y_i is modeled as a Beta-Bernoulli process: $p_i \sim \text{Beta}(\alpha_i, \beta_i)$ and $y_i \sim \text{Bernoulli}(p_i)$. Our model f_θ maps the input x_i to the parameters of this Beta distribution, i.e., $f_\theta(x_i) = (\hat{\alpha}_i, \hat{\beta}_i)$, where $\hat{\alpha}_i, \hat{\beta}_i > 0$. The mean of the predicted distribution $\text{Beta}(\hat{\alpha}_i, \hat{\beta}_i)$ serves as the calibrated point estimate, and the variance as the epistemic uncertainty about the latent event probability p_i: $\hat{p}_i = \mathbb{E}[p_i] = \frac{\hat{\alpha}_i}{\hat{\alpha}_i + \hat{\beta}_i}$, $\hat{u}_i = \text{Var}[p_i] = \frac{\hat{\alpha}_i \hat{\beta}_i}{(\hat{\alpha}_i + \hat{\beta}_i)^2 (\hat{\alpha}_i + \hat{\beta}_i + 1)}$. Note that this epistemic uncertainty is the calibrator's learned estimate of uncertainty about p_i, distinct from the input LLM's internal confidence in its own forecast.

2.3. Model architecture and input

Since the input x_i is in natural language, we parameterize f_θ as a language model encoder followed by an MLP head that outputs Beta parameter values. We include an initial forecast $\hat{p}_i^{\text{init}}$ as part of the input, and train f_θ to act as a post-hoc calibrator that refines this initial belief. While our framework imposes no constraints on the source of the initial belief, for our experiments we derive $\hat{p}_i^{\text{init}}$ by prompting a separate LLM (input LLM) for verbalized probability. This follows the prior work, and offers both simplicity and broad applicability. Therefore, our input takes the form: $x_i =$ "Question: $\{text_i\}$; Initial forecast: $\{\hat{p}_i^{\text{init}}\}$".

Importantly, note that our calibrator f_θ is model-agnostic in terms of the input LLM. This allows us to calibrate forecasts from any black-box models, thus we can leverage strong proprietary LLMs without fine-tuning them. Furthermore, because the task of calibration is distinct from the heavy reasoning required for the initial forecast, f_θ can be significantly smaller than the input LLM. We show in Appendix L.4 that a small 1-billion parameter language model is sufficient to effectively calibrate initial forecasts from larger models, making our method computationally efficient.

2.4. Objective functions

Given $D = \{(x_i, y_i, \mathbf{q}_i)\}_{i=1}^N$, we train the calibrator f_θ using supervision from both binary outcomes $y_i \in \{0, 1\}$ and human forecasts $\mathbf{q}_i = \{q_{i1}, \dots, q_{ik_i}\}$, thus the overall training objective combines both signals: $\mathcal{L}_{\text{total}} = \sum_{i=1}^N \mathcal{L}_{\text{binary},i} + \sum_{i=1}^N \mathcal{L}_{\text{human},i}$.[2]

$\mathcal{L}_{\text{binary}}$ is the Beta-Bernoulli negative log-likelihood, which reduces to the Binary Cross-Entropy (BCE) loss on the Beta mean $\hat{p}_i$ (see Appendix G). BCE is a strictly proper scoring rule (Gneiting & Raftery, 2007), which incentivizes learning true probability p_i^* as it is minimized if and only if $\hat{p}_i = p_i^*$. However, learning from only binary outcomes is insufficient for capturing a meaningful distribution under limited data.[3] There is an identifiability problem where the loss is invariant to the scale of Beta parameters. Moreover, viewing human forecasts as noisy samples from the true distribution over p, they also enrich the information available per question. To learn from human forecasts, we can simply match our predicted Beta distributions with human forecast histograms via a Kullback-Leibler (KL) divergence objective. Let $\mathbf{h}_i$ be the normalized human forecast histogram over B bins. We minimize $\mathcal{L}_{\text{human},i} = \text{KL}(\mathbf{h}_i || \text{Beta}(\alpha_i, \beta_i))$.

[2] We find performance to be relatively robust across a broad range of weightings between $\mathcal{L}_{\text{binary}}$ and $\mathcal{L}_{\text{human}}$ in Appendix L.2.

[3] In theory, infinite samples from the latent probability distribution would identify the ground-truth Beta parameters with BCE loss. However, in practice each event resolves only once, making the distribution shape hard to learn without additional signals. See a toy experiment validating this in Appendix E.

Table 1. Test performance across input LLMs and baseline methods. Best results are bolded, and second-best results are underlined. KL is the KL divergence between the predicted distribution and the human forecast distribution on the test set.

Input LLM / Method	Brier↓	Acc↑	AUC↑	ECE↓	KL↓
Human Baseline	0.061	0.923	0.958	0.055	
Claude-Sonnet-4					
Verbalized	0.146	0.799	0.723	0.104	
Ensemble ($n = 3$)	0.143	0.800	0.736	0.100	
Platt Scaling	0.129	0.827	0.723	0.034	
Isotonic Regression	0.129	0.832	0.724	0.038	
BBC (binary only)	0.128	0.833	0.732	0.036	9.004
BBC (binary+human)	**0.125**	**0.837**	**0.742**	**0.027**	**8.775**
Qwen3-32B					
Verbalized	0.158	0.796	0.661	0.111	
Ensemble ($n = 10$)	0.143	0.813	**0.700**	0.097	
Platt Scaling	0.142	0.827	0.661	0.079	
Isotonic Regression	0.137	0.823	0.662	**0.040**	
P(True)	0.232	0.761	0.664	0.230	
Future-as-a-label-32B	0.137	0.829	0.677	0.046	
BBC (binary only)	0.135	0.832	0.684	0.044	11.085
BBC (binary+human)	**0.133**	**0.833**	0.686	0.046	**9.402**
Qwen3-8B					
Verbalized	0.185	0.750	0.633	0.164	
Ensemble ($n = 10$)	0.169	0.755	0.661	0.148	
Platt Scaling	0.151	0.823	0.633	0.094	
Isotonic Regression	0.141	0.818	0.638	0.054	
P(True)	0.222	0.772	0.619	0.220	
OpenForecaster-8B	0.157	0.794	0.663	0.084	
BBC (binary only)	0.138	0.824	0.662	**0.044**	11.550
BBC (binary+human)	**0.137**	**0.828**	**0.673**	0.050	**9.730**

3. Experiments

3.1. Experimental setup

Dataset. We collect binary questions from the forecasting platforms Metaculus and Polymarket. To ensure data quality and sufficient crowd signal, we filter out low-volume questions and exclude domains that are hard to model, such as sports, weather, and cryptocurrency.[4] In total, this results in 11,355 resolved questions. As shown in Table 5, we split the data temporally: 7,824 training questions resolved before April 2025, 1,917 validation questions resolved between April and July 2025, and 1,614 test questions resolved between August 2025 and January 2026. This testing phase occurs entirely beyond the knowledge cutoff of all LLMs we tested, preventing data leakage. Appendix F.3 describes how we construct the human forecast distribution $\mathbf{h}_i$.

Training details. We choose Llama-3.2-1B (Grattafiori et al., 2024) as the base of our Beta-Bernoulli Calibrator f_θ. To further relax the prior family, we model p as a mixture of $K = 5$ Beta distributions (see Appendix H.1 for the formulation and Appendix L.1 for an ablation on K). The calibrator f_θ encodes input x_i (question and initial forecast), takes the second-to-last-layer hidden state at the final non-padding token as the sequence representation, and maps it through a two-layer feed-forward head to predict the Beta parameters. To prevent overly extreme predictions, we constrain $\alpha_{ik}, \beta_{ik} > 1$. See Appendix H.2 for more training and hyperparameter details.

Baseline methods. We evaluate the Beta-Bernoulli Calibrator using initial forecasts from 7 LLMs in two configurations: trained only on binary outcome labels (**BBC, binary only**) and trained on both binary outcomes and human forecasts (**BBC, binary + human**). This allows us to see the effect of learning from human uncertainty. We compare BBC against three classes of baselines: (i) uncertainty estimation methods, including Verbalized, which directly prompts the LLM for a probability; Ensemble, which averages n verbalized forecasts; and P(True) (Kadavath et al., 2022), which estimates $\hat{p} = P(\text{Yes})/(P(\text{Yes}) + P(\text{No}))$ from a Yes/No prompt and requires white-box access; (ii) post-hoc calibration methods, including Platt scaling (Platt et al., 1999) and isotonic regression (Zadrozny & Elkan, 2002); and (iii) forecasting-specialized LLMs, including OpenForecaster-8B (Chandak et al., 2025) and Future-as-a-label-32B (Turtel et al., 2026). Verbalized forecasts are used as the initial forecasts for all calibration methods, including BBC. Full baseline details are provided in Appendix I.

3.2. Results

As shown in Table 1 and 8, our Beta-Bernoulli Calibrator (BBC) significantly improves over the initial verbalized baseline. For example, using Claude-Sonnet-4 (Anthropic, 2025) as the input LLM, BBC (binary+human) reduces the Brier score from 0.146 to 0.125 (14.4% improvement) and increases AUC (Bradley, 1997) from 72.3% to 74.2%. Figures 4 and 5 visualize the calibration gains: while verbalized probability forecasts exhibit overconfidence noted in prior work (Schoenegger et al., 2024; Halawi et al., 2024; Nel, 2025), our framework shifts the curves toward the identity line. In general, stronger input LLMs provide better raw forecasts, and BBC effectively builds on these stronger priors without requiring any fine-tuning of the input model. The logit-based P(True) method is poorly calibrated with high ECE (Naeini et al., 2015). As the LLM generates a rationale before choosing "Yes" or "No", this intermediate reasoning often amplifies the model's preference for one outcome, pushing the resulting token probabilities toward extreme values near 0 or 1. Ensembling provides modest improvements over the verbalized baseline, but remains less calibrated than BBC.

Compared to post-hoc calibration baselines (Platt scaling and Isotonic Regression), BBC consistently achieves better Brier score and stronger discrimination (AUC). While these

[4] See Appendix F for preprocessing and dataset details.

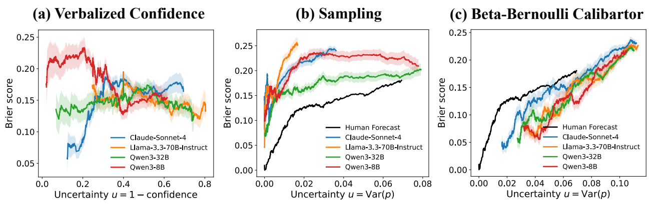

Figure 1. Brier score vs. ranked epistemic uncertainty, smoothed with a window of 300. (a) Verbalized confidence, (b) Sampling-based variance, and (c) Predicted Beta distribution variance.

methods reduce ECE by learning global mappings, they are fundamentally limited by their monotonic nature, which prevents them from improving ranking performance. Notably, our lightweight calibrator exceeds models specifically fine-tuned for forecasting. This suggests applying a lightweight calibrator to the base LLM can be a more efficient alternative to fine-tuning the underlying model itself. Moreover, as our framework is model-agnostic, it can be applied on top of any forecasting model, including those fine-tuned forecasters, to further improve performance. Table 9 shows that BBC may still provide gains beyond traditional calibration methods on most metrics.

The human baseline remains substantially stronger than current LLM-based forecasters, with Brier score of 0.061 and AUC of 0.958, motivating their use as supervision.[5] Comparing BBC (binary only) to BBC (binary+human), we observe further improvements especially in Brier score and AUC. This suggests that human forecast distributions provide a consensus signal about the latent event probability, offering informative supervision beyond a single realized outcome.[6] To quantify how much BBC moves the predicted distribution closer to the human forecast distribution, we compute their KL divergence and find that adding human distributional supervision consistently reduces KL, indicating better alignment with human beliefs (Table 1).

4. Analysis

Epistemic uncertainty. Ideally, when a model is highly uncertain about its own forecast, on average we expect higher prediction error (Brier score). We check this by plotting the Brier score as a function of ranked uncertainty. As discussed in Section 2.2, we quantify BBC's epistemic uncertainty using the variance of the predicted Beta distribution. We compare against two baselines: (i) verbalized confidence, obtained by prompting the input LLM to report a confidence score after giving an answer, with uncertainty defined as $u = 1 -$ confidence, and (ii) sampling-based uncertainty, which we take the variance of multiple samples (taken from the ensemble baseline).

[5]Computed using the mean crowd forecast.

[6]While our training set contains high-quality human forecasts (Brier $= 0.085$), Appendix L.3 stress-tests BBC under sparse or biased human supervision.

Table 2. OOD performance on Kalshi. BBC achieves the best Brier score, accuracy, and ECE while remaining competitive in AUC.

Input LLM / Method	Brier↓	Acc↑	AUC↑	ECE↓
Qwen3-32B				
Verbalized	0.238	0.605	0.651	0.097
Platt Scaling	0.251	0.596	0.651	0.148
Isotonic Regression	0.244	0.605	0.651	0.141
Future-as-a-label-32B	0.258	0.607	0.638	0.159
BBC	**0.228**	**0.609**	**0.658**	**0.059**
Qwen3-8B				
Verbalized	0.258	0.585	0.609	0.116
Platt Scaling	0.258	0.595	0.609	0.146
Isotonic Regression	0.258	0.597	0.608	0.152
OpenForecaster-8B	0.244	**0.599**	**0.632**	0.093
BBC	**0.235**	**0.599**	0.620	**0.061**

Figure 1(a) shows that the self-reported confidence is noisy and disjointed from empirical performance: lower verbalized confidence in general does not correspond to a lower Brier score. The sampling-based uncertainty in Figure 1(b) is more informative, but it becomes less discriminative at higher uncertainty. In contrast, in Figure 1(c), BBC produces an uncertainty measure that is consistently aligned with errors across input LLMs, similar to the trend in the human forecast baseline, offering a more reliable signal of forecasting errors.

Generalization to out-of-distribution data. To further test whether our calibrator is robust in the out-of-distribution (OOD) setting, we evaluate it on 3,208 questions from the prediction platform Kalshi that resolved after August 2025.[7] Table 2 shows that traditional post-hoc calibration methods fail to generalize well, resulting in even worse calibration with higher ECE. In contrast, BBC maintains strong performance, achieving better calibration and accuracy.

Ablations. We provide additional ablations on the calibrator model choice in Appendix L.4 and the input design in Appendix L.5. The results show that a 1B calibrator is already effective, while larger calibrators can improve AUC. We also find that the initial forecast is important: removing it hurts discrimination, whereas adding the input LLM's rationale provides only marginal gains at additional computational cost.

Conclusion. We introduce the Beta-Bernoulli Calibrator, a lightweight, model-agnostic post-hoc method that maps an initial probability forecast to a Beta distribution over event likelihood, learning from both binary outcomes and human forecast distributions. Across multiple input LLMs, BBC achieves stronger in- and out-of-distribution performance than traditional post-hoc calibration methods and models specifically fine-tuned for forecasting.

[7]See details in Appendix F.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Acknowledgments

This work was supported in part by Visko AI, Toyota Research Institute R2I program, a Google TPU Award, and the Institute of Information & Communications Technology Planning Evaluation (IITP) under grant RS-2024-00469482, funded by the Ministry of Science and ICT (MSIT) of the Republic of Korea in connection with the Global AI Frontier Lab International Collaborative Research. The compute is supported by the NYU High Performance Computing resources, services, and staff expertise.

References

Alur, R., Stadie, B. C., Kang, D., Chen, R., McManus, M., Rickert, M., Lee, T., Federici, M., Zhu, R., Fogerty, D., et al. AIA forecaster: Technical report. *arXiv preprint arXiv:2511.07678*, 2025.

Anthropic. Introducing claude 4. `https://www.anthropic.com/news/claude-4`, 2025.

Azaria, A. and Mitchell, T. The internal state of an LLM knows when it's lying. In *Findings of EMNLP*, 2023.

Bradley, A. P. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

Brier, G. W. Verification of forecasts expressed in terms of probability. *Monthly weather review*, 78(1):1–3, 1950.

Chandak, N., Goel, S., Prabhu, A., Hardt, M., and Geiping, J. Scaling open-ended reasoning to predict the future. *arXiv preprint arXiv:2512.25070*, 2025.

Charpentier, B., Zügner, D., and Günnemann, S. Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts. In *NeurIPS*, 2020.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Damani, M., Puri, I., Slocum, S., Shenfeld, I., Choshen, L., Kim, Y., and Andreas, J. Beyond binary rewards: Training LMs to reason about their uncertainty. In *ICLR*, 2026.

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

Fadeeva, E., Rubashevskii, A., Shelmanov, A., Petrakov, S., Li, H., Mubarak, H., Tsymbalov, E., Kuzmin, G., Panchenko, A., Baldwin, T., et al. Fact-checking the output of large language models via token-level uncertainty quantification. In *Findings of ACL*, 2024.

Gneiting, T. and Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *ICML*, 2017.

Halawi, D., Zhang, F., Yueh-Han, C., and Steinhardt, J. Approaching human-level forecasting with language models. In *NeurIPS*, 2024.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., et al. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022.

Hurst, A., Lerer, A., Goucher, A. P., Perelman, A., Ramesh, A., Clark, A., Ostrow, A., Welihinda, A., Hayes, A., Radford, A., et al. GPT-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

Jiang, M., Ruan, Y., Huang, S., Liao, S., Pitis, S., Grosse, R. B., and Ba, J. Calibrating language models via augmented prompt ensembles. In *ICML*, 2023.

Kadavath, S., Conerly, T., Askell, A., Henighan, T., Drain, D., Perez, E., Schiefer, N., Hatfield-Dodds, Z., DasSarma, N., Tran-Johnson, E., et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Kapoor, S., Gruver, N., Roberts, M., Collins, K., Pal, A., Bhatt, U., Weller, A., Dooley, S., Goldblum, M., and Wilson, A. G. Large language models must be taught to know what they don't know. In *NeurIPS*, 2024.

Karger, E., Bastani, H., Yueh-Han, C., Jacobs, Z., Halawi, D., Zhang, F., and Tetlock, P. ForecastBench: A dynamic benchmark of AI forecasting capabilities. In *ICLR*, 2024.

Kendall, A. and Gal, Y. What uncertainties do we need in bayesian deep learning for computer vision? In *NeurIPS*, 2017.

Kirichenko, P., Ibrahim, M., Chaudhuri, K., and Bell, S. J. AbstentionBench: Reasoning LLMs fail on unanswerable questions. *arXiv preprint arXiv:2506.09038*, 2025.

Lahiri, K. and Yang, L. Forecasting binary outcomes. In *Handbook of economic forecasting*, volume 2, pp. 1025–1106. Elsevier, 2013.

Li, Y., Xiong, M., Wu, J., and Hooi, B. ConfTuner: Training large language models to express their confidence verbally. In *NeurIPS*, 2025.

Lin, S., Hilton, J., and Evans, O. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022.

Ling, C., Zhao, X., Zhang, X., Cheng, W., Liu, Y., Sun, Y., Oishi, M., Osaki, T., Matsuda, K., Ji, J., et al. Uncertainty quantification for in-context learning of large language models. In *NAACL*, 2024.

Mei, Z., Zhang, C., Yin, T., Lidard, J., Sho, O., and Majumdar, A. Reasoning about uncertainty: Do reasoning models know when they don't know? In *Findings of EACL*, 2026.

Murphy, K. Agentic forecasting using sequential bayesian updating of linguistic beliefs. *arXiv preprint arXiv:2604.18576*, 2026.

Naeini, M. P., Cooper, G., and Hauskrecht, M. Obtaining well calibrated probabilities using bayesian binning. In *AAAI*, 2015.

Nel, L. Do large language models know what they don't know? Kalshibench: A new benchmark for evaluating epistemic calibration via prediction markets. *arXiv preprint arXiv:2512.16030*, 2025.

Niculescu-Mizil, A. and Caruana, R. Predicting good probabilities with supervised learning. In *ICML*, 2005.

Orgad, H., Toker, M., Gekhman, Z., Reichart, R., Szpektor, I., Kotek, H., and Belinkov, Y. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *ICLR*, 2025.

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.

Schoenegger, P., Tuminauskaite, I., Park, P. S., Bastos, R. V. S., and Tetlock, P. E. Wisdom of the silicon crowd: LLM ensemble prediction capabilities rival human crowd accuracy. *Science Advances*, 10(45):eadp1528, 2024.

Sensoy, M., Kaplan, L., and Kandemir, M. Evidential deep learning to quantify classification uncertainty. In *NeurIPS*, 2018.

Shorinwa, O., Mei, Z., Lidard, J., Ren, A. Z., and Majumdar, A. A survey on uncertainty quantification of large language models: Taxonomy, open research challenges, and future directions. *ACM Computing Surveys*, 58(3): 1–38, 2025.

Team, Q. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2025.

Tetlock, P. E. and Gardner, D. *Superforecasting: The art and science of prediction*. Random House, 2016.

Tian, K., Mitchell, E., Zhou, A., Sharma, A., Rafailov, R., Yao, H., Finn, C., and Manning, C. D. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *EMNLP*, 2023.

Turtel, B., Wilczewski, P., Franklin, D., and Skothiem, K. Future-as-Label: Scalable supervision from real-world outcomes. *arXiv preprint arXiv:2601.06336*, 2026.

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 *ICLR*, 2024.

Xu, T., Wu, S., Diao, S., Liu, X., Wang, X., Chen, Y., and Gao, J. SaySelf: Teaching LLMs to express confidence with self-reflective rationales. In *EMNLP*, 2024.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., Lv, C., et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Yang, Q., Mahns, S., Li, S., Gu, A., Wu, J., and Xu, H. LLM-as-a-prophet: Understanding predictive intelligence with Prophet Arena. In *ICLR*, 2026.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W., Salakhutdinov, R., and Manning, C. D. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*, 2018.

Zadrozny, B. and Elkan, C. Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers. In *ICML*, 2001.

Zadrozny, B. and Elkan, C. Transforming classifier scores into accurate multiclass probability estimates. In *ACM SIGKDD*, 2002.

Zeng, Z., Liu, J., Chen, S., He, T., Liao, Y., Tian, Y., Wang, J., Wang, Z., Yang, Y., Yin, L., et al. FutureX: An advanced live benchmark for LLM agents in future prediction. *arXiv preprint arXiv:2508.11987*, 2025.

Zhang, A., Chen, Y., Pan, J., Zhao, C., Panda, A., Li, J., and He, H. Reasoning models know when they're right: Probing hidden states for self-verification. In *COLM*, 2025.

Zhang, C., Liu, F., Basaldella, M., and Collier, N. LUQ: Long-text uncertainty quantification for LLMs. In *EMNLP*, 2024.

Appendix

A. BBC pipeline overview

Figure 2 shows the full pipeline of BBC, which takes a forecasting question and an input LLM's verbalized forecast and outputs a distributional calibrated forecast.

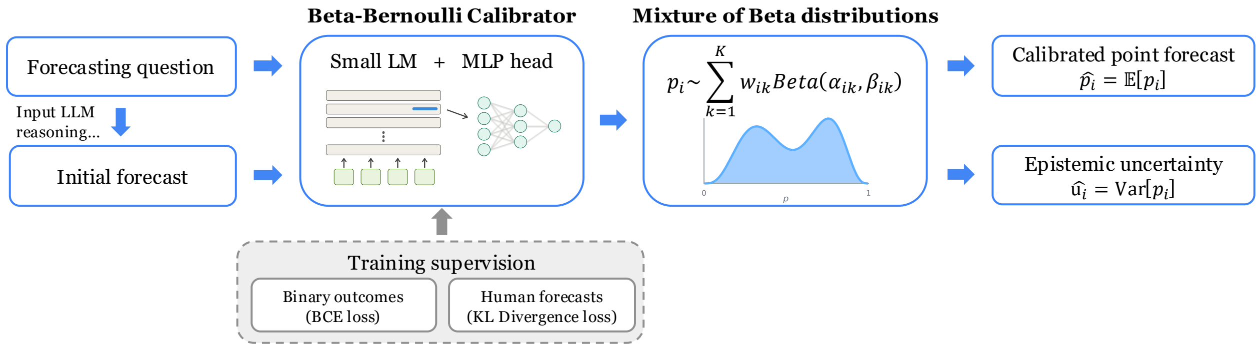

Figure 2. Overview of the Beta-Bernoulli Calibrator (BBC). Given a forecasting question and an initial verbalized forecast from an input LLM, BBC outputs a mixture of Beta distributions over the event probability. BBC is itself a small language model with an MLP head that predicts the Beta parameters, trained using supervision from both binary outcomes and human forecasts. The mean of the predicted distribution serves as the calibrated point forecast and the variance as epistemic uncertainty.

B. Related work

Traditional calibration and evidential methods. Earlier work on uncertainty calibration mainly focused on post-hoc calibration of classifier outputs (DeGroot & Fienberg, 1983; Niculescu-Mizil & Caruana, 2005). Parametric methods such as Platt scaling (Platt et al., 1999) and temperature scaling (Guo et al., 2017) learn global parameters to rescale prediction scores across all samples. For nonparametric methods, histogram binning (Zadrozny & Elkan, 2001) uses the empirical outcome frequencies in bins as calibrated scores, and isotonic regression (Zadrozny & Elkan, 2002) learns a monotonic piecewise constant function to transform uncalibrated scores. Post-hoc calibration relates to BBC's role as a calibrator, while *Evidential Deep Learning* (EDL) relates to its probabilistic output parameterization. EDL models a Dirichlet over categorical probability predictions, with the Beta as the binary special case (Sensoy et al., 2018; Charpentier et al., 2020). However, our framework differs in two ways: (i) rather than collecting evidence directly from task inputs (like a cat image) in an end-to-end classifier, BBC is a stagewise calibrator that adjusts on top of another model's natural language output, benefiting from its reasoning capability; (ii) rather than learning only from deterministic class labels, we introduce learning from human forecast distributions, which provides additional supervision and helps address the identifiability issue discussed in Section 2.4.

Uncertainty estimation and calibration in LLMs. In the context of LLMs, the focus shifts from calibrating classifier scores to estimating the reliability of natural language generations (Shorinwa et al., 2025). Most work has concentrated on tasks such as mathematics (e.g. GSM8K (Cobbe et al., 2021)) and reasoning (e.g. HotpotQA (Yang et al., 2018)). In these settings, the model generates an answer and the elicited confidence score should reflect the probability that the answer is correct. We survey uncertainty estimation and calibration in LLMs into training-free and training-based categories:

Training-free methods extract uncertainty estimates without modifying model weights. In black-box settings, *verbalized uncertainty* can be obtained simply by prompting the model to state its confidence after providing an answer (Tian et al., 2023). However, such self-reported confidence is found to be overconfident (Xiong et al., 2024; Mei et al., 2026; Kirichenko et al., 2025). White-box methods instead leverage internal model features. These are primarily *logit-based*, estimating uncertainty through the entropy of output-token probabilities (Ling et al., 2024; Fadeeva et al., 2024). Another popular approach is $P(\text{True})$, where the model is prompted to assess whether its own answer is "True" or "False," and the probability of getting the "True" token is interpreted as its confidence score (Kadavath et al., 2022). Finally, sampling-based ensembles (such as majority vote or taking average) can be applied to both verbalized and logit-based methods to further improve calibration (Zhang et al., 2024; Jiang et al., 2023; Xiong et al., 2024).

Training-based methods learn calibrated confidence predictors or elicit better-calibrated uncertainty through training.

Although some studies find that verbalized confidence or simple token-based signals can be well-calibrated (Tian et al., 2023; Kadavath et al., 2022), other work shows they underperform training-based approaches (Kapoor et al., 2024). A primary direction probes internal representations, as hidden layers have been shown to encode information regarding truthfulness and potential error patterns (Orgad et al., 2025). These methods train **probing classifiers** on top of LLM hidden states to predict answer correctness (Kadavath et al., 2022; Azaria & Mitchell, 2023; Kapoor et al., 2024; Zhang et al., 2025). Beyond add-on probes, another line **fine-tunes** the LLM itself to express calibrated uncertainty in natural language. For example, Lin et al. (2022) fine-tune GPT-3 using the model's empirical accuracy across different question types as a proxy for ground truth confidence. More recently, work has explored the use of proper scoring rules as fine-tuning objectives (Li et al., 2025) or incorporating calibration-aware reward functions in RL to incentivize honest confidence reporting (Xu et al., 2024; Damani et al., 2026).

LLMs in forecasting. The uncertainty work reviewed above treats uncertainty as confidence in an answer's correctness. While the tasks are useful for measuring model performance, they primarily address epistemic uncertainty, which arises from a model's lack of knowledge and is, in principle, reducible (Kendall & Gal, 2017). That is, tasks such as mathematical problem solving do not involve inherent randomness, and a perfect system should always produce the correct answer with confidence 1.0. In forecasting, by contrast, uncertainty estimation is not merely a diagnostic measure of confidence, but the primary output of interest for predicting future events. Real-world events such as market fluctuations or weather patterns possess aleatoric uncertainty, or irreducible randomness inherent to the event itself. Current work typically prompts LLMs to provide verbalized probability estimates (Karger et al., 2024; Zeng et al., 2025; Yang et al., 2026), which are often overconfident (Schoenegger et al., 2024; Halawi et al., 2024; Nel, 2025). To improve the forecasts, Alur et al. (2025) apply ensembling and traditional post-hoc calibration, Murphy (2026) combine an agentic search loop with hierarchical Platt scaling, and Halawi et al. (2024) fine-tune GPT-4 on subsets where model outperforms human crowd. Recent efforts explore RL, using Brier score and accuracy as reward signals for open-ended forecasting (Chandak et al., 2025), and binary cross-entropy for binary prediction tasks (Turtel et al., 2026). While prior work targets verbalized point forecasts, our work is the first to utilize human forecast signals to model the distribution over event probabilities. Moreover, as a post-hoc calibrator, our method is complementary to these methods: it can be applied on top of them to further improve their forecasts.

C. Limitations

Our work has several limitations. First, as we gather human forecasts from prediction platforms, the training signal may inherit topic biases to politics and economics. Second, although our ablations indicate that larger calibrators are possible to bring further gains, we do not fully explore this scaling trend and leave a systematic study of the calibrator's upper bound to future work. Third, BBC does not incorporate information updates, and instead relies solely on its internal knowledge to calibrate the belief. A natural direction for future work is to extend BBC to condition on the temporal dimension and intermediate evidence, enabling us to model how human uncertainty shifts over time when new information emerges.

D. Evaluation metrics

Brier score (Brier, 1950) is the Mean Squared Error between the predicted probabilities and the binary outcomes. A lower Brier score corresponds to better predictions. BS $= \frac{1}{N}\sum_{i=1}^{N}(\hat{p}_i - y_i)^2$.

Accuracy is the fraction of events predicted correctly, given that an event is predicted positive when the predicted probability exceeds a threshold (e.g. 0.5). Acc $= \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\{(\hat{p}_i \geq 0.5) = y_i\}$.

AUC (Bradley, 1997) is defined as the Area Under the Receiver Operating Characteristic curve. It is independent of any threshold, measuring how well $\hat{p}_i$ can discriminate events that happen versus those do not.

Expected Calibration Error (ECE) (Naeini et al., 2015) is a calibration metric. Ideally, for example, if we track 100 events where the model predicts a probability of 0.7, we expect to see 70 events occur in the end. The perfect calibration is formally written by $P(y = 1 \mid p = q) = q, \forall q$. ECE measures how uncalibrated a model is by taking the expectation of the absolute difference between the model prediction and the empirical event occurrences, computed from the set of events with similar predictions. More precisely, we split the events into equal bins based on the model predictions (e.g., $[0, 0.1], (0.1, 0.2], \dots$). For each bin, we compute (i) the average predicted probability prob(B_m), and (ii) the empirical accuracy acc(B_m) – the fraction of events in this bin being true. Then, ECE is defined as the weighted average of absolute differences in these two quantities:

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\text{acc}(B_m) - \text{prob}(B_m)\right|$$

$$\text{where prob}(B_m) = \frac{\sum_{i \in B_m} \hat{p}_i}{|B_m|},\ \text{acc}(B_m) = \frac{\sum_{i \in B_m} y_i}{|B_m|}.$$

Moreover, visualizing $\text{acc}(B_m)$ against $\text{prob}(B_m)$ produces a **Reliability Diagram** (DeGroot & Fienberg, 1983; Niculescu-Mizil & Caruana, 2005), where a perfectly calibrated model follows the identity line $y = x$.

E. Why human forecasts help: a toy experiment

In this section, we provide a toy experiment to further demonstrate the necessity of human forecasts as distributional supervision. The toy experiment aims to show that, if the aggregated human forecast distribution is an approximate distribution over the latent event probability, it can provide both the missing signal about the distribution shape, and additional supervision beyond the single binary outcome, resulting in better forecasts.

We construct a synthetic Beta-Bernoulli setting. We generate 30,000 questions with 10-dimensional input features mapped through a nonlinear function into three distinct ground-truth regimes: (i) *Confident YES:* $\text{Beta}(50, 10)$ with $p = 0.83$; (ii) *Uncertain:* $\text{Beta}(5, 5)$ with $p = 0.5$; and (iii) *Confident NO:* $\text{Beta}(10, 50)$ with $p = 0.17$. Each question receives a single binary outcome $y \sim \text{Bernoulli}(p)$ with $p \sim \text{Beta}(\alpha, \beta)$. Human forecasts are simulated by drawing 1000 samples from the true Beta distribution. We train a 2-layer MLP that maps input features to Beta parameters (α, β) under three loss configurations: *Binary only* (BCE), *Human only*, and *Binary + Human*.

We can see that binary-only training achieves a reasonable Brier score of 0.194 (Table 3) and learns approximate means (Table 4), but completely fails to recover the Beta shape: the predicted distributions (red) deviate substantially from the ground truth (black) in Figure 3. Adding human supervision recovers parameters close to ground truth and provides better forecasting performance (Table 3).

Therefore, while BCE is optimal for estimating the mean event probability given sufficient repeated observations, forecasting settings provide only a single realization per event. Human forecast distributions act as a noisy proxy for the latent probability distribution and provide additional information about uncertainty that improves calibration beyond binary supervision alone. Empirically, this holds as long as human forecasts are a reasonable proxy of the true underlying distribution. In our training set, human forecasts significantly outperform all LLMs (human: Brier $= 0.085$, AUC $= 0.945$ vs. model average: Brier $= 0.196$, AUC $= 0.704$), validating their quality as supervision.

Table 3. Toy experiment: forecasting metrics across three loss configurations. Best results are bolded.

Method	Brier↓	Acc↑	AUC↑	ECE↓
Binary only	0.194	0.709	0.775	0.053
Human only	**0.183**	0.717	0.780	0.016
Binary + Human	**0.183**	**0.719**	**0.784**	**0.011**

Table 4. Toy experiment: recovery of ground-truth Beta parameters. Binary-only training fails to recover (α, β) while binary+human recovers parameters close to ground truth (bolded).

	Ground truth		Binary only		Binary + Human	
Regime	(α, β)	Mean	(α, β)	Mean	(α, β)	Mean
Confident YES	$(50, 10)$	0.833	$(1.4, 0.4)$	0.778	$(\mathbf{44.8},\ \mathbf{9.1})$	0.831
Uncertain	$(5, 5)$	0.500	$(0.6, 0.8)$	0.429	$(\mathbf{4.5},\ \mathbf{5.0})$	0.474
Confident NO	$(10, 50)$	0.167	$(0.2, 1.2)$	0.143	$(\mathbf{6.7},\ \mathbf{33.5})$	0.167

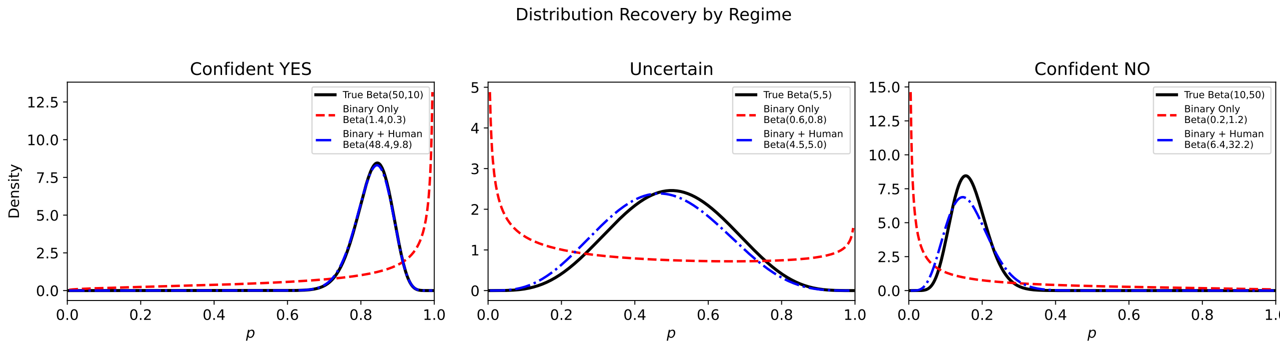

Figure 3. Toy experiment: predicted Beta distributions vs. ground truth across the three regimes. Binary-only training (red dashed) fails to recover the distribution shape, while binary+human (blue) approximately recovers the ground truth (black).

F. Dataset

F.1. Data preprocessing

Metaculus. We obtain Metaculus data from the public API `https://www.metaculus.com/api2/questions/`. We filter to binary questions with at least one human forecast, and exclude meta-questions that predict the community prediction on another Metaculus question.

Polymarket. For Polymarket, we exclude the sports, cryptocurrency, and weather domains. The question open date is set to be the earlier of (i) 30 days before the last observed timestamp and (ii) 7 days after the first observed timestamp. We notice the number of recently resolved questions is much larger than in earlier periods, which would make the test set disproportionately large. To address this, we filter by popularity rather than random sampling. Specifically, we filter for questions with at least 5 price history entries for training, at least 30 for validation, and at least 100 for testing.

Kalshi. For the OOD Kalshi dataset, we again exclude the sports, cryptocurrency, and weather domains. We further filter for events that have a total trading volume greater than 10,000. This results in 3,208 events that resolved after August 2025. Since Kalshi events can include multiple markets (outcome options), we further convert them to binary questions by randomly selecting one market per event and asking whether that outcome occurs.

F.2. Dataset statistics

Table 5 provides the dataset statistics by source and split. Table 6 presents the category distribution. Most questions in our main dataset fall under "Politics & Governance" (6,163) and "Economics & Business" (1,960), two domains closely tied to real-world decision-making. We use GPT-4o-mini (Hurst et al., 2024) to assign categories to questions, using the prompt from Halawi et al. (2024). Table 7 shows the category distribution of Kalshi dataset, using their existing category tags.

Table 5. Dataset statistics by source and split. Train: resolved before April 2025; Val: resolved between April and July 2025; Test: resolved between August 2025 and January 2026.

Source	Train	Val	Test
Metaculus	3,264	545	420
Polymarket	4,560	1,372	1,194
Total	7,824	1,917	1,614

F.3. Constructing the human forecast distribution

While both platforms provide human forecast information, the nature of this information is different. On Metaculus, a user j can submit a forecast probability $q_{ij} \in [0, 1]$ for event i, and we can directly get a 100-bin forecast histogram $\mathbf{h}_i$

Table 6. Category distribution for our main dataset.

Category	Metaculus	Polymarket	Total
Politics & Governance	1,693	4,470	6,163
Economics & Business	899	1,061	1,960
Arts & Recreation	116	762	878
Security & Defense	391	238	629
Science & Tech	304	185	489
Sports	213	226	439
Healthcare & Biology	296	44	340
Environment & Energy	194	39	233
Other	75	91	166
Education & Research	48	10	58
Total	4,229	7,126	11,355

Table 7. Category distribution for the Kalshi dataset.

Category	Kalshi
Financials	1,685
Entertainment	537
Mentions	425
Politics	236
Companies	108
Economics	67
Elections	64
Science and Technology	39
World	23
Social	12
Health	8
Transportation	3
Education	1
Total	3,208

from the API. On Polymarket, users trade yes/no contracts, whose prices can be interpreted as the market's consensus probability of the event. In that case, we can only construct a proxy histogram $\mathbf{h}_i$ by binning the market prices over a time window (between the market open time and close time), capturing the temporal volatility. As a result, the Metaculus histogram reflects explicit crowd agreement across different forecasters, while the Polymarket histogram reflects agreement of aggregate market beliefs over time. Nevertheless, they both provide informative human signals about the uncertainty in the underlying event probability.

G. Objective function: learning from binary outcomes

By marginalizing out p, the marginal likelihood of y_i is:

$$\begin{aligned}
P(y_i|\alpha_i,\beta_i) &= \int_0^1 P(y_i|p_i)\,P(p_i|\alpha_i,\beta_i)\ \mathrm{d}p_i \\
&= \int_0^1 p_i^{y_i}(1-p_i)^{1-y_i}\,\frac{1}{B(\alpha_i,\beta_i)}\,p_i^{\alpha_i-1}(1-p_i)^{\beta_i-1}\ \mathrm{d}p_i \\
&= \frac{1}{B(\alpha_i,\beta_i)}\int_0^1 p_i^{\alpha_i+y_i-1}(1-p_i)^{\beta_i+1-y_i-1}\ \mathrm{d}p_i \\
&= \frac{B(\alpha_i+y_i,\ \beta_i+1-y_i)}{B(\alpha_i,\beta_i)}.
\end{aligned}$$

This equals the mean $\hat{p}_i = \frac{\alpha_i}{\alpha_i+\beta_i}$ when $y_i = 1$, and $1-\hat{p}_i = \frac{\beta_i}{\alpha_i+\beta_i}$ when $y_i = 0$. Therefore, with only binary labels, the Beta-Bernoulli loss reduces exactly to the BCE loss with respect to the mean $\hat{p}_i$:

$$\mathcal{L}_{\text{binary},i} = -\log P(y_i|\alpha_i,\beta_i) = -y_i\log(\hat{p}_i) - (1-y_i)\log(1-\hat{p}_i).$$

H. Training details

H.1. Relaxing the constraint by mixture of Beta

In Section 2, we introduce modeling p as a single Beta distribution, which can be limited when the true underlying belief is multi-modal (e.g., when opinions are polarized). To further relax the prior family, in our experiments, we model p as a mixture of K Beta distributions. Therefore, the output dimension expands to K pairs of (α,β) with corresponding weights. Concretely, with $\alpha_{ik},\beta_{ik} > 0, w_{ik} \geq 0, \sum_{k=1}^K w_{ik} = 1$, $f_\theta^{\text{mixture}}(x_i) = \{(\alpha_{ik},\beta_{ik},w_{ik})\}_{k=1}^K$, and $p_i \sim \sum_{k=1}^K w_{ik}\text{Beta}(\alpha_{ik},\beta_{ik})$. During training, we optimize $\mathcal{L}_{\text{binary},i}$ in terms of the mixture mean $\hat{p}_i = \mathbb{E}[p_i] = \sum_{k=1}^K w_{ik}\,\frac{\alpha_{ik}}{\alpha_{ik}+\beta_{ik}}$, and match the mixture distribution to human forecast histogram for $\mathcal{L}_{\text{human},i}$.

H.2. Implementation and hyperparameters

All initial forecasts are generated using greedy decoding (temperature = 0). During training, the final MLP head is trained and the base LLM is fine-tuned with Low-Rank Adapters (LoRA) (Hu et al., 2022). We sweep hyperparameters of LoRA rank $r \in \{128, 256\}$ (with LoRA scaling $\alpha = r$) and learning rates $\lambda \in \{1\text{e-}6, 5\text{e-}6\}$ over 3 random seeds, training for 15 epochs and selecting the best models with validation Brier score (Brier, 1950). Final results are reported as the average and standard deviation of the top-5 models on the test set. For all ablation studies, we report results averaged over 3 random seeds with fixed $r = 256$ and $\lambda = 1\text{e-}6$. All experiments are run on a single NVIDIA L40S or H200 GPU.

I. Baseline Details

We compare BBC against both uncertainty estimation/calibration methods and models fine-tuned specifically for forecasting:

- **Verbalized:** We directly prompt the LLM to state probabilistic forecasts. These estimates serve as initial forecasts for the calibration methods (including ours, Platt Scaling and Isotonic Regression), whose goal is to improve this baseline. The prompt can be found in Appendix M.
- **Ensemble:** We prompt the LLM n times and average the forecasts as $\hat{p}$.
- **P(True)** (Kadavath et al., 2022): A logit-based uncertainty estimation method that prompts the model to explicitly answer "Yes" or "No", and derives $\hat{p} = \frac{P(\text{Yes})}{P(\text{Yes})+P(\text{No})}$. This is only feasible in white-box models.
- **Platt Scaling** (Platt et al., 1999): A parametric calibration method that models $\hat{p} = \sigma(A\hat{p}^{\text{init}} + B)$. The parameters A and B are learned by minimizing the negative log-likelihood on validation set.
- **Isotonic Regression** (Zadrozny & Elkan, 2002): A non-parametric calibration method that learns a piecewise constant function by minimizing squared error under an order constraint.
- **OpenForecaster-8B** (Chandak et al., 2025)[8]: A Qwen3-8B (Yang et al., 2025) model fine-tuned with RL, using accuracy and Brier score as rewards. In addition to the Metaculus binary questions, the model uses 52K synthetically generated open-ended forecasting questions from news articles. Their training data cutoff is April 2025, consistent with the temporal split of ours.
- **Future-as-a-label-32B** (Turtel et al., 2026)[9]: A Qwen3-32B (Yang et al., 2025) model fine-tuned with RL, using BCE as reward. The training data consists 5,120 binary questions generated from news articles, with a cutoff date of January 30, 2025.

J. Results for more input LLMs

Table 8 reports standard deviations for all models in Table 1, along with additional test results for three more input LLMs: Qwen2.5-72B-Instruct (Team, 2025), Qwen2.5-7B-Instruct (Team, 2025), and Llama-3.1-8B-Instruct (Grattafiori et al., 2024). Corresponding reliability diagrams and uncertainty-error plots for additional models are shown in Figures 5 and 6. We notice consistent trends with our main findings, with BBC generally providing better forecasts than the baseline methods.

K. BBC further improves forecasting-specialized models

Table 9 shows that BBC can be applied on top of forecasting-specialized LLMs, providing further gains over both their verbalized forecasts and standard post-hoc calibration methods.

L. Additional ablation studies

L.1. Ablation: number of mixture components K

In our main experiments, we model p as a mixture of $K = 5$ Beta distributions. Here we study the effect of varying the number of mixture components $K \in \{1, 3, 5, 7, 10\}$. As shown in Table 10, $K = 1$ runs notably underperform mixtures, while performance across $K = 3$ to $K = 10$ remains roughly stable.

[8]https://huggingface.co/nikhilchandak/OpenForecaster-8B
[9]https://huggingface.co/LightningRodLabs/future-as-label-paper-step160

Table 8. Test performance across input LLMs and baseline methods. Best results are bolded, and second-best results are underlined. KL is the KL divergence between the predicted distribution and the human forecast distribution on the test set.

	Brier↓		Accuracy↑		AUC↑		ECE↓		KL↓	
Input LLM / Method	mean	std	mean	std	mean	std	mean	std	mean	std
Human Baseline	0.061		0.923		0.958		0.055			
Claude-Sonnet-4										
Verbalized	0.146		0.799		0.723		0.104			
Ensemble ($n = 3$)	0.143		0.800		0.736		0.100			
Platt Scaling	0.129		0.827		0.723		0.034			
Isotonic Regression	0.129		0.832		0.724		0.038			
BBC (binary only)	0.128	(0.001)	0.833	(0.002)	0.732	(0.003)	0.036	(0.011)	9.004	(0.266)
BBC (binary+human)	**0.125**	(0.002)	**0.837**	(0.004)	**0.742**	(0.007)	**0.027**	(0.006)	**8.775**	(0.319)
Llama-3.3-70B-Instruct										
Verbalized	0.157		0.777		0.655		0.119			
Ensemble ($n = 10$)	0.151		0.782		0.669		0.099			
Platt Scaling	0.139		**0.829**		0.655		0.051			
Isotonic Regression	0.138		0.822		0.654		**0.043**			
$P(\mathrm{True})$	0.265		0.726		0.656		0.265			
BBC (binary only)	0.138	(0.003)	0.816	(0.011)	0.671	(0.010)	0.054	(0.012)	11.564	(0.172)
BBC (binary+human)	**0.135**	(0.002)	**0.829**	(0.003)	**0.679**	(0.006)	0.045	(0.012)	**9.526**	(0.507)
Qwen3-32B										
Verbalized	0.158		0.796		0.661		0.111			
Ensemble ($n = 10$)	0.143		0.813		**0.700**		0.097			
Platt Scaling	0.142		0.827		0.661		0.079			
Isotonic Regression	0.137		0.823		0.662		**0.040**			
$P(\mathrm{True})$	0.232		0.761		0.664		0.230			
Future-as-a-label-32B	0.137		0.829		0.677		0.046			
BBC (binary only)	0.135	(0.005)	0.832	(0.012)	0.684	(0.009)	0.044	(0.015)	11.085	(0.453)
BBC (binary+human)	**0.133**	(0.002)	**0.833**	(0.004)	0.686	(0.004)	0.046	(0.014)	**9.402**	(0.405)
Qwen3-8B										
Verbalized	0.185		0.750		0.633		0.164			
Ensemble ($n = 10$)	0.169		0.755		0.661		0.148			
Platt Scaling	0.151		0.823		0.633		0.094			
Isotonic Regression	0.141		0.818		0.638		0.054			
$P(\mathrm{True})$	0.222		0.772		0.619		0.220			
OpenForecaster-8B	0.157		0.794		0.663		0.084			
BBC (binary only)	0.138	(0.001)	0.824	(0.008)	0.662	(0.008)	**0.044**	(0.008)	11.550	(0.390)
BBC (binary+human)	**0.137**	(0.004)	**0.828**	(0.004)	**0.673**	(0.016)	0.050	(0.019)	**9.730**	(0.231)
Qwen2.5-72B-Instruct										
Verbalized	0.174		0.764		0.655		0.144			
Ensemble ($n = 10$)	0.165		0.766		0.676		0.138			
Platt Scaling	0.138		0.825		0.655		0.049			
Isotonic Regression	0.138		0.825		0.655		0.044			
$P(\mathrm{True})$	0.261		0.736		0.633		0.262			
BBC (binary only)	0.135	(0.003)	0.825	(0.007)	0.670	(0.005)	0.042	(0.025)	11.247	(0.535)
BBC (binary+human)	**0.133**	(0.003)	**0.829**	(0.002)	**0.683**	(0.009)	**0.035**	(0.017)	**9.160**	(0.556)
Qwen2.5-7B-Instruct										
Verbalized	0.170		0.778		0.621		0.124			
Ensemble ($n = 10$)	0.159		0.797		0.646		0.121			
Platt Scaling	0.145		0.829		0.621		0.085			
Isotonic Regression	0.140		0.826		0.624		0.064			
$P(\mathrm{True})$	0.166		0.827		0.568		0.165			
BBC (binary only)	0.138	(0.004)	0.830	(0.004)	0.668	(0.012)	0.067	(0.019)	12.227	(0.587)
BBC (binary+human)	**0.135**	(0.002)	**0.831**	(0.007)	**0.676**	(0.006)	**0.054**	(0.018)	**9.677**	(0.385)
Llama-3.1-8B-Instruct										
Verbalized	0.169		0.771		0.639		0.140			
Ensemble ($n = 10$)	0.165		0.766		0.662		0.153			
Platt Scaling	0.144		0.827		0.639		0.069			
Isotonic Regression	0.143		0.823		0.637		0.060			
$P(\mathrm{True})$	0.217		0.768		0.623		0.216			
BBC (binary only)	0.138	(0.002)	0.819	(0.004)	0.669	(0.004)	**0.055**	(0.012)	11.703	(0.336)
BBC (binary+human)	**0.136**	(0.003)	**0.828**	(0.003)	**0.673**	(0.006)	0.058	(0.015)	**9.815**	(0.281)

L.2. Ablation: loss coefficients

Given the training objective $\mathcal{L}_{\text{total}} = \lambda_{\text{binary}} \sum_i \mathcal{L}_{\text{binary},i} + \lambda_{\text{human}} \sum_i \mathcal{L}_{\text{human},i}$, we ablate the loss coefficients as shown in Table 11. Training on the human loss only ($\lambda_{\text{binary}} = 0, \lambda_{\text{human}} = 1$) achieves comparable performance to the binary+human training setup, validating that human forecasts are a useful supervision signal on their own. Compared to binary-only

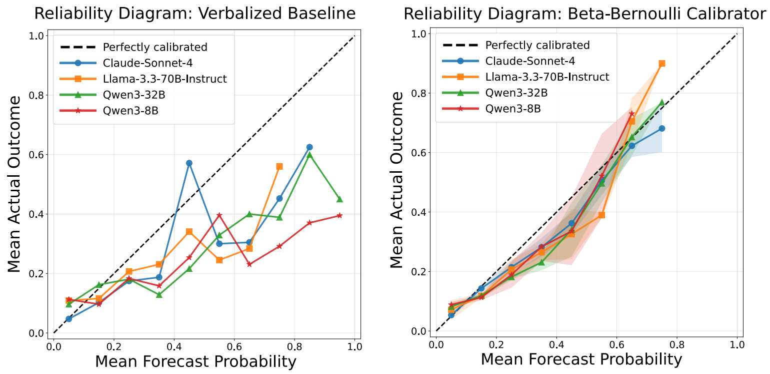

Figure 4. Reliability diagrams. Left: Verbalized probability forecasts are overconfident. Right: Our Beta-Bernoulli Calibrator improves calibration. Bands show ± 1 std across the top-5 runs.

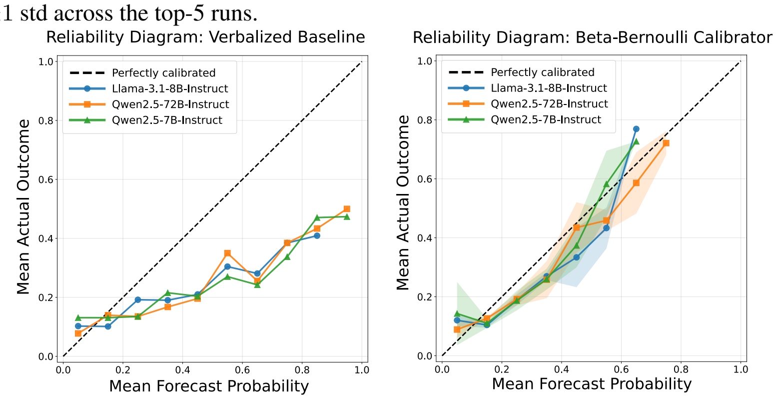

Figure 5. Reliability Diagram for additional input LLMs. Verbalized forecasts exhibit overconfidence (left), and BBC improves calibration (right).

training, adding human supervision consistently improves performance with lower Brier score and higher AUC, and results remain relatively stable across a broad range of coefficients.

L.3. Robustness to sparse/biased human forecasts

A natural concern with using human forecasts as supervision is whether BBC remains effective when they are sparse or biased. We note that human forecasts in our training set are well-calibrated and significantly outperform all LLMs (human: Brier $= 0.085$, AUC $= 0.945$ vs. model average: Brier $= 0.196$, AUC $= 0.704$), validating their quality as supervision. The corruption experiments below therefore serve as stress tests rather than reflections of realistic settings.

L.3.1. Sparsity

To simulate sparse human signals, we retain $x\%$ of the forecasts per question during training. Table 12 shows that BBC performance is largely robust to forecast sparsity: even with only 10% of forecasts retained, BBC matches or exceeds the binary-only baseline, indicating that even sparse human forecasts provide useful distributional signals. As more human signals become available, we observe consistent improvements across both input LLMs.

L.3.2. Bias

We further test BBC's robustness under three types of synthetic corruption applied to the human forecasts during training:

- **Noise:** replacing a fraction of forecasts with Uniform$(0, 1)$ draws;
- **Directional shift (γ):** scaling each forecast as $q' = 0.5 + \gamma(q - 0.5)$, where $\gamma < 1$ pulls forecasts toward 0.5 (underconfident) and $\gamma > 1$ pushes them toward the extremes (overconfident);

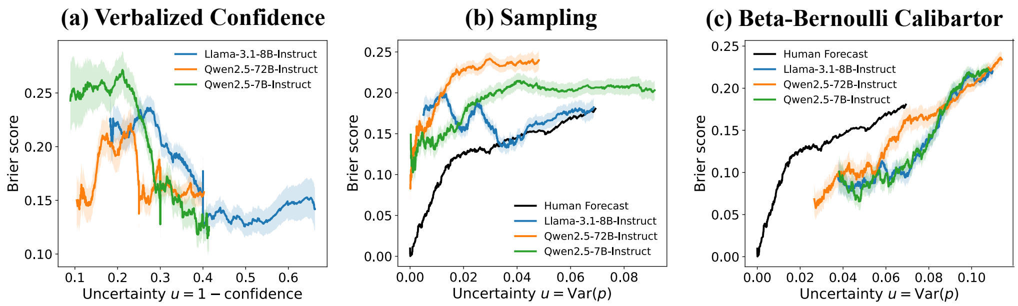

Figure 6. The plot of Brier score against ranked epistemic uncertainty, smoothed with a window of 300. The uncertainty is defined as (a) 1 - verbalized confidence, (b) Sampling-based variance, and (c) BBC variance. The observation aligns with the discussion in Section 4.

Table 9. Applying BBC (binary+human) on top of forecasting-specialized models further improves forecasts, with consistent gains in Brier score and AUC over other post-hoc calibration methods.

Input LLM / Method	Brier↓	Acc↑	AUC↑	ECE↓
OpenForecaster-8B				
Verbalized	0.157	0.794	0.663	0.084
Platt Scaling	0.141	**0.824**	0.663	0.059
Isotonic Regression	0.139	0.820	0.665	**0.044**
BBC	**0.136**	0.821	**0.690**	0.051
Future-as-a-label-32B				
Verbalized	0.137	0.829	0.677	0.046
Platt Scaling	0.138	0.825	0.676	0.061
Isotonic Regression	0.134	0.828	0.677	**0.037**
BBC	**0.132**	**0.833**	**0.694**	0.041

- **Additive shift (δ):** shifting all forecasts by a constant $q' = q + \delta$, where $\delta > 0$ is optimistic and $\delta < 0$ is pessimistic.

Table 13 shows that systematic underconfidence and noise are the most damaging, as both push the predicted Beta distribution toward flat and uninformative shapes. AUC remains relatively robust across all corruption types, since systematic bias preserves the relative ordering among predictions. Interestingly, mild overconfidence ($\gamma = 1.5$) and negative additive shift ($\delta = -0.1$) actually sometimes improve performance over the uncorrupted baseline. This reflects the class imbalance in our test set: 83% of events resolve to "No", so corruptions that push forecasts toward 0 (negative shift) or sharpen them away from 0.5 (overconfidence) tend to align with the majority outcome.

L.4. Ablation: calibrator model family and size

In our main experiments, we use Llama-3.2-1B as the base model for BBC, and show that it is already an efficient choice that outperforms standard baselines. Here we analyze the effect of varying both the calibrator family and size, as shown in Table 14. Overall, scaling the calibrator provides only modest changes in Brier and accuracy, but we observe a clear performance gain in AUC with a larger 8B model, showing the potential in scaling up. Comparing model families, Llama-based calibrators consistently outperform similarly sized Qwen-based models.

L.5. Ablation: input content

Table 15 studies how the calibrator input x_i affects BBC. For our main experiments, x_i encodes both the event information and an initial forecast $\hat{p}^{\text{init}}$. When removing $\hat{p}^{\text{init}}$, BBC no longer acts as a post-hoc calibrator and a significant drop in AUC is observed. This indicates that BBC is most effective when refining an existing belief, benefiting from the stronger input LLM forecast rather than predicting from scratch. However, further enriching x_i yields only marginal gains while introducing additional computational overhead. In particular, appending the input LLM's rationale results in only a slight increase in AUC at higher computational cost, suggesting that providing the initial forecast alone is sufficient in practice.

Table 10. Ablation on the number of mixture components K in BBC.

	Brier↓		Accuracy↑		AUC↑		ECE↓	
Input LLM / K	mean	std	mean	std	mean	std	mean	std
Claude-Sonnet-4								
$K = 1$	0.129	(0.002)	0.831	(0.005)	0.737	(0.002)	0.053	(0.018)
$K = 3$	**0.126**	(0.000)	**0.836**	(0.003)	<u>0.742</u>	(0.008)	0.038	(0.002)
$K = 5$	**0.126**	(0.001)	**0.836**	(0.003)	**0.744**	(0.005)	0.036	(0.007)
$K = 7$	**0.126**	(0.001)	<u>0.835</u>	(0.003)	0.740	(0.002)	<u>0.035</u>	(0.003)
$K = 10$	**0.126**	(0.002)	0.834	(0.004)	0.737	(0.005)	**0.034**	(0.004)
Llama-3.3-70B-Instruct								
$K = 1$	0.145	(0.003)	<u>0.829</u>	(0.004)	0.665	(0.004)	0.096	(0.017)
$K = 3$	<u>0.137</u>	(0.002)	0.825	(0.007)	<u>0.677</u>	(0.009)	0.065	(0.010)
$K = 5$	**0.136**	(0.001)	**0.830**	(0.001)	**0.679**	(0.012)	0.060	(0.005)
$K = 7$	0.138	(0.003)	0.818	(0.011)	0.671	(0.004)	<u>0.052</u>	(0.024)
$K = 10$	**0.136**	(0.001)	0.823	(0.001)	0.675	(0.001)	**0.047**	(0.009)

M. Prompts

The verbalized probabilistic forecasts are elicited using the prompt in Figure 7. We use greedy decoding with temperature = 0, except for the ensemble method, where we use temperature = 1. Figure 8 shows the prompt used to obtain verbalized confidence for the epistemic uncertainty analysis.

You're an expert in forecasting events. Make a prediction of the probability that the question will be resolved as true. You MUST give a probability estimate between 0 and 1 UNDER ALL CIRCUMSTANCES. If for some reason you can't answer, pick the base rate, but return a number between 0 and 1.

To support your reasoning, recall relevant recent events, facts, or widely known information. Ensure your rationale is well-grounded and coherent.

Once you have completed your reasoning, output your answer as a number between 0 and 1.

Question: {}
Resolution Criteria: {}

Today's date: {}
Question close date: {}

Please follow the output format:
[Rationale:] xxx
[Answer:] a number between 0 and 1

Figure 7. Prompt for obtaining verbalized forecasts from the input LLMs.

Table 11. Sensitivity to loss coefficients λ_{binary} and λ_{human}. Adding human supervision consistently improves over binary-only training across a broad range of coefficients, and human-only training ($\lambda_{\text{binary}} = 0$) is competitive with binary+human training, confirming that human forecasts are a useful supervision signal on their own.

		Brier↓		**Accuracy↑**		**AUC↑**		**ECE↓**	
λ_{binary}	λ_{human}	mean	std	mean	std	mean	std	mean	std
Claude-Sonnet-4									
1	0.0	0.127	(0.002)	0.834	(0.003)	0.729	(0.011)	<u>0.030</u>	(0.010)
1	0.1	**0.125**	(0.001)	<u>0.835</u>	(0.002)	0.739	(0.010)	**0.026**	(0.005)
1	0.5	0.127	(0.001)	<u>0.835</u>	(0.001)	0.736	(0.007)	0.036	(0.005)
1	1.0	<u>0.126</u>	(0.001)	**0.836**	(0.003)	<u>0.744</u>	(0.005)	0.036	(0.007)
1	5.0	<u>0.126</u>	(0.002)	0.834	(0.001)	0.743	(0.017)	0.034	(0.003)
1	10.0	**0.125**	(0.000)	**0.836**	(0.005)	**0.745**	(0.014)	0.031	(0.007)
0	1.0	**0.125**	(0.000)	**0.836**	(0.005)	<u>0.744</u>	(0.015)	0.032	(0.011)
Llama-3.3-70B-Instruct									
1	0.0	0.138	(0.004)	0.815	(0.015)	0.667	(0.011)	0.049	(0.015)
1	0.1	**0.135**	(0.001)	0.824	(0.003)	0.671	(0.003)	**0.039**	(0.011)
1	0.5	<u>0.136</u>	(0.002)	<u>0.827</u>	(0.007)	0.673	(0.012)	<u>0.047</u>	(0.008)
1	1.0	<u>0.136</u>	(0.001)	**0.830**	(0.001)	0.679	(0.012)	0.060	(0.005)
1	5.0	0.137	(0.003)	0.826	(0.012)	<u>0.683</u>	(0.005)	0.067	(0.011)
1	10.0	0.137	(0.004)	0.825	(0.012)	**0.684**	(0.003)	0.062	(0.016)
0	1.0	0.137	(0.005)	0.824	(0.010)	**0.684**	(0.002)	0.063	(0.021)

Table 12. Robustness to sparse human forecasts.

	Brier↓		**Accuracy↑**		**AUC↑**		**ECE↓**	
Input LLM / % Retained	mean	std	mean	std	mean	std	mean	std
Claude-Sonnet-4								
0% (binary only)	<u>0.127</u>	(0.002)	<u>0.834</u>	(0.003)	0.729	(0.011)	**0.030**	(0.010)
10%	0.129	(0.001)	0.831	(0.001)	0.738	(0.011)	0.061	(0.006)
25%	<u>0.127</u>	(0.001)	0.832	(0.006)	<u>0.743</u>	(0.007)	0.048	(0.003)
50%	0.128	(0.001)	0.831	(0.007)	0.739	(0.011)	0.047	(0.003)
100%	**0.126**	(0.001)	**0.836**	(0.003)	**0.744**	(0.005)	<u>0.036</u>	(0.007)
Llama-3.3-70B-Instruct								
0% (binary only)	0.138	(0.004)	0.815	(0.015)	0.667	(0.011)	**0.049**	(0.015)
10%	<u>0.137</u>	(0.003)	<u>0.829</u>	(0.009)	<u>0.676</u>	(0.005)	0.071	(0.012)
25%	<u>0.137</u>	(0.003)	0.823	(0.013)	**0.679**	(0.009)	0.063	(0.010)
50%	**0.136**	(0.000)	0.828	(0.005)	0.674	(0.009)	0.067	(0.005)
100%	**0.136**	(0.001)	**0.830**	(0.001)	**0.679**	(0.012)	<u>0.060</u>	(0.005)

Table 13. Robustness to corrupted human forecasts.

Corruption	Parameter	Brier↓		Accuracy↑		AUC↑		ECE↓	
		mean	std	mean	std	mean	std	mean	std
Claude-Sonnet-4									
Binary only	—	0.127	(0.002)	0.834	(0.003)	0.729	(0.011)	**0.030**	(0.010)
Noise	25%	0.131	(0.003)	0.836	(0.005)	0.740	(0.008)	0.084	(0.013)
Noise	50%	0.144	(0.003)	0.833	(0.008)	0.728	(0.007)	0.138	(0.012)
Underconfident (γ)	0.5	0.157	(0.001)	**0.838**	(0.003)	0.733	(0.002)	0.174	(0.001)
Overconfident (γ)	1.5	**0.125**	(0.002)	0.833	(0.004)	**0.745**	(0.008)	0.032	(0.002)
Negative shift (δ)	-0.1	**0.125**	(0.002)	0.835	(0.001)	0.740	(0.005)	**0.030**	(0.003)
Positive shift (δ)	$+0.1$	0.137	(0.006)	0.829	(0.008)	0.739	(0.002)	0.109	(0.028)
No corruption	—	0.126	(0.001)	0.836	(0.003)	0.744	(0.005)	0.036	(0.007)
Llama-3.3-70B-Instruct									
Binary only	—	0.138	(0.004)	0.815	(0.015)	0.667	(0.011)	0.049	(0.015)
Noise	25%	0.144	(0.005)	0.826	(0.008)	0.672	(0.015)	0.107	(0.017)
Noise	50%	0.155	(0.005)	0.824	(0.010)	0.673	(0.005)	0.150	(0.017)
Underconfident (γ)	0.5	0.170	(0.003)	0.821	(0.011)	0.672	(0.008)	0.190	(0.010)
Overconfident (γ)	1.5	**0.134**	(0.002)	0.828	(0.003)	**0.684**	(0.005)	**0.030**	(0.003)
Negative shift (δ)	-0.1	**0.134**	(0.001)	0.829	(0.005)	0.678	(0.003)	0.035	(0.019)
Positive shift (δ)	$+0.1$	0.148	(0.003)	0.821	(0.007)	0.674	(0.001)	0.125	(0.010)
No corruption	—	0.136	(0.001)	**0.830**	(0.001)	0.679	(0.012)	0.060	(0.005)

Table 14. Ablation on calibrator family and size for BBC. 1B calibrator is already an effective choice.

Input LLM / Calibrator	Brier↓	Acc↑	AUC↑	ECE↓
Claude-Sonnet-4				
Llama-3.2-1B	0.126	0.836	0.744	0.036
Qwen2.5-0.5B	0.129	0.827	0.737	0.044
Llama-3.2-3B	**0.124**	**0.840**	0.741	**0.030**
Qwen3-4B-Instruct	0.127	0.835	0.736	0.037
Llama-3.1-8B	**0.124**	0.834	**0.752**	0.036
Llama-3.3-70B-Instruct				
Llama-3.2-1B	0.136	**0.830**	0.679	0.060
Qwen2.5-0.5B	0.138	0.818	0.667	0.061
Llama-3.2-3B	0.135	**0.830**	0.677	0.059
Qwen3-4B-Instruct	0.138	0.821	0.675	**0.045**
Llama-3.1-8B	**0.134**	0.826	**0.693**	0.058

Table 15. Ablation on calibrator input. Removing the initial forecast drops performance, while adding a rationale brings minimal benefit at extra cost.

Setting / Method	Brier↓	Acc↑	AUC↑	ECE↓
BBC w/o initial forecast	0.140	0.827	0.650	0.060
Claude-Sonnet-4				
BBC w initial forecast	0.126	0.836	0.744	**0.036**
+ Reasoning	**0.125**	**0.837**	**0.745**	0.041
Llama-3.3-70B-Instruct				
BBC w initial forecast	**0.136**	**0.830**	0.679	0.060
+ Reasoning	0.137	0.821	**0.682**	**0.052**

You're an expert in forecasting events. Make a prediction of the probability that the question will be resolved as true. You MUST give a probability estimate between 0 and 1 UNDER ALL CIRCUMSTANCES. If for some reason you can't answer, pick the base rate, but return a number between 0 and 1.

To support your reasoning, recall relevant recent events, facts, or widely known information. Ensure your rationale is well-grounded and coherent.

Once you have completed your reasoning, output your answer as a number between 0 and 1.

After you give your probability, also report how confident you are in that probability on a scale from 0 to 1 (0 = no confidence, 1 = extremely confident).

Question: {}
Resolution Criteria: {}

Today's date: {}
Question close date: {}

Please follow the output format:
[Rationale:] xxx
[Answer:] a number between 0 and 1
[Confidence:] a number between 0 and 1

Figure 8. Prompt for obtaining verbalized forecasts together with verbalized confidence in that forecast.