The Future Outcome Reasoning and Confidence Assessment Benchmark

Anonymous ACL submission

Abstract

Forecasting is an important task in many domains, such as technology and economics. However existing forecasting benchmarks largely lack comprehensive confidence assessment, focus on limited question types, and often consist of artificial questions that do not align with real-world human forecasting needs. To address these gaps, we introduce FORECAST (**F**uture **O**utcome **R**easoning and **C**onfidence **A**ssessment), a benchmark that evaluates models' ability to make predictions and their confidence in them. FORECAST spans diverse forecasting scenarios involving Boolean questions, timeframe prediction, and quantity estimation, enabling a comprehensive evaluation of both prediction accuracy and confidence calibration for real-world applications.

1 Introduction

Recent advances in large language models (LLMs) have significantly improved their performance across a wide range of natural language processing (NLP) tasks. Alongside these developments, various benchmarks and datasets have been introduced to effectively assess the capabilities of LLMs, particularly in terms of knowledge and reasoning (Zellers et al., 2019; Guo et al., 2023). Fact-based benchmarks, like TruthfulQA (Lin et al., 2022) evaluate LLMs based on factual correctness, focusing on tasks like retrieving and verifying facts that are known.

Forecasting is a crucial yet challenging task across various domains, including technology, economics, and public policy. Unlike tasks that rely on retrieving and verifying existing knowledge, forecasting requires predicting plausible outcomes for future events, often under uncertainty and incomplete information. This makes forecasting particularly difficult, as models must infer trends, assess probabilities, and adapt to new information. Several datasets have been introduced to evaluate LLMs' forecasting capabilities. ForecastQA (Jin et al., 2020) uses a multiple-choice format where models predict future outcomes, but it lacks confidence assessment. AutoCast (Zou et al., 2022) incorporates confidence intervals, however its confidence estimates are not designed for forecasting. Other datasets such as ExpTime (Yuan et al., 2024a) are artificially generated from structured data, focusing on explainable event forecasting based on temporal knowledge graphs.

All of these aforementioned benchmarks ignore a crucial aspect of forecasting: confidence evaluation. Confidence plays a central role in forecasting, as predictions about unresolved events inherently lack definitive correctness at the time of evaluation. Predictions made with absolute certainty are undesirable, even if they

Type	Question	Resolution	Confidence
Boolean Question	Will a Frontier AI lab be established in China by 2026?	Yes	0.73
Timeframe Prediction	When will OpenAI announce GPT-5?	2024-08-01	0.85
Quantity Estimation	How many spacecrafts will land on the moon in 2025?	3	0.65

Table 1: Examples of forecasting questions with their resolutions and confidence scores.

ultimately prove to be correct, because they fail to account for the uncertain nature of future events. Moreover, miscalibrated confidence can lead to poor decision-making: overconfident yet incorrect forecasts may result in costly errors, while underconfident but accurate predictions can erode trust in the model. Therefore, well-calibrated confidence scores are as crucial as the accuracy of the predictions themselves.

To address these gaps, we present FORECAST: **F**uture **O**utcome **R**easoning and **C**onfidence **A**ssessment. FORECAST focuses on three distinct types of forecasting questions, shown in Table 1: (1) *Boolean questions*, such as "Will there be a Frontier AI lab in China before 2026?"; (2) *Timeframe Prediction*, such as "When will OpenAI announce GPT-5?"; and (3) *Quantity Estimation*, such as "How many spacecrafts will land on the moon in each of the following years?" We conduct experiments using a range of models differing in size, training objectives, and cutoff times, and explore multiple methods for estimating model confidence. Our results reveal that forecasting remains highly challenging for current LLMs, particularly in confidence evaluation, with no direct correlation between prediction performance and confidence calibration, and while larger models sometimes improve performance, the effect is inconsistent.

2 FORECAST: Problem Formulation

System responses in FORECAST consist of (1) a prediction answering a question given the available information and (2) a confidence score in the prediction. This ensures a comprehensive assessment of forecasting, accounting for both correctness and confidence calibration. Questions belong to three types. (1) *Boolean Questions*, which ask yes/no questions about the occurrence of future events (sometimes within a certain timeframe). Boolean questions are simple to evaluate, and they can still be surprisingly challenging (Clark et al., 2019). (2) *Timeframe Prediction*, which requires predicting a specific timeframe for an event, and are essential for applications where knowing whether an event will happen or not without a timeframe is insufficient. (3) *Quantity Estimation*, which involves providing numerical estimates related to future events, e.g., economic indicators or trends.

Formally, let Q represent a question about a future event, and let M denote a system with access to information up to time t_{train} (e.g., the system's knowledge cutoff point). The objective is for M to produce an answer A in $\mathcal{A}$ and an associated confidence score C, where:

$$M(Q) \rightarrow (A, C), \quad C \in [0, 1]. \tag{1}$$

$$A = \arg\max_{a \in \mathcal{A}} P(X = a | Q, \mathcal{K}(t_{train})). \tag{2}$$

Here, $\mathcal{K}(t_{train})$ represents the knowledge accessible to the model up to time t_{train}. The answer space $\mathcal{A}$ depends on the type of forecasting question: for *Boolean Questions*, $\mathcal{A} = \{\text{Yes}, \text{No}\}$; for *Timeframe Prediction*, $\mathcal{A}$ consists of a single date in the `YYYY-MM-DD` format; and for *Quantity Estimation*, $\mathcal{A} = \mathbb{R}$, representing real numbers.

3 Evaluating Predictions and Confidence

Evaluating Boolean Questions For Boolean questions, where the answer space is $\mathcal{A} = \{\text{Yes}, \text{No}\}$, prediction performance is evaluated using standard classification metrics, including accuracy and F1-score. Confidence calibration is assessed using a modified version of the Brier score (Brier, 1950), which measures the mean squared error between predicted confidence and the gold confidence provided in the dataset, which we assume is provided, and represents the likelihood of an event occurring. The modified Brier score is defined as:

$$\text{Brier} = \frac{1}{N}\sum_{i=1}^{N}(C_i^{\text{pred}} - C_i^{\text{gold}})^2, \quad (3)$$

where C_i^{pred} is the model's predicted confidence and C_i^{gold} is the gold confidence. This modification ensures that models are evaluated based on their ability to match the likelihood of an event. Lower Brier scores indicate better calibration, reflecting how well the predicted confidence aligns with the likelihood of the event.

Evaluating Timeframe Prediction For timeframe prediction, where the answer space consists of specific dates in the `YYYY-MM-DD` format, predictive accuracy is measured using absolute day error (ADE). Given a predicted date D_i^{pred} and the gold date D_i^{gold}, we compute the normalized error as:

$$E_i^{\text{ADE}} = \frac{2}{1 + e^{-\alpha|D_i^{\text{pred}} - D_i^{\text{gold}}|}} - 1, \quad (4)$$

where α is a scaling factor that controls how sharply large errors are penalized. This transformation ensures that extreme deviations do not disproportionately dominate the evaluation.

For confidence calibration, rely on the Continuous Ranked Probability Score (CRPS) (Matheson and Winkler, 1976), which is a generalisation of the Mean Absolute Error to probabilistic forecasts, and extend it to compare the predicted probability distribution with a gold distribution. Specifically, we assume that both the predicted and the gold confidence predictions follow Gaussian distributions, namely $\mathcal{N}(D_i^{\text{pred}}, \sigma_i^{\text{pred}})$ and $\mathcal{N}(D_i^{\text{gold}}, \sigma_i^{\text{gold}})$ respectively, where the standard deviations are computed as:

$$\sigma_i^{\text{pred}} = \sigma_{\max} \cdot (1 - C_i^{\text{pred}}) + \sigma_{\min} \cdot C_i^{\text{pred}}, \quad (5)$$

$$\sigma_i^{\text{gold}} = \sigma_{\max} \cdot (1 - C_i^{\text{gold}}) + \sigma_{\min} \cdot C_i^{\text{gold}}. \quad (6)$$

Here, C_i^{pred} is the model's predicted confidence for the ith question, and C_i^{gold} is the corresponding gold confidence provided in our dataset. The parameters $\sigma_{\max}$ and $\sigma_{\min}$ define the upper and lower bounds for the standard deviation. Intuitively, when confidence is low ($C \approx 0$), uncertainty is high, leading to $\sigma \approx \sigma_{\max}$, while when confidence is high ($C \approx 1$), uncertainty is low, resulting in $\sigma \approx \sigma_{\min}$. We then compute the CRPS as the integrated squared difference between the cumulative distribution functions (CDFs) of the predicted and gold distributions:

$$\text{CRPS} = \frac{1}{N}\sum_{i=1}^{N}\int \left(F_i^{\text{pred}}(d) - F_i^{\text{gold}}(d)\right)^2 \, \mathrm{d}d, \quad (7)$$

where F_i^{pred} and F_i^{gold} denote the CDFs of the predicted and gold Gaussian distributions, respectively. A lower CRPS indicates better calibration, as it reflects a closer match between the predicted uncertainty and the uncertainty as specified by the gold confidence.

Evaluating Quantity Estimation For quantity estimation, where the answer space consists of non-negative real numbers ($\mathcal{A} = \mathbb{R}_{\geq 0}$), we

evaluate prediction performance using two error metrics: absolute percentage error (APE) and mean absolute error (MAE). Given a predicted quantity Q_i^{pred} and the gold quantity Q_i^{gold}, the normalized errors are computed as:

$$E_i^{\text{APE}} = \frac{2}{1 + e^{-\alpha \frac{|Q_i^{\text{pred}} - Q_i^{\text{gold}}|}{Q_i^{\text{gold}} + \epsilon}}} - 1, \quad (8)$$

$$E_i^{\text{MAE}} = \frac{2}{1 + e^{-\alpha |Q_i^{\text{pred}} - Q_i^{\text{gold}}|}} - 1. \quad (9)$$

Here, ϵ is a small constant to prevent division by zero, and α is a scaling factor that controls how sharply large errors are penalized, similar to the timeframe prediction evaluation. Confidence calibration is assessed using CRPS, following the same Gaussian assumption as in timeframe prediction. The predicted quantity is modeled as a Gaussian distribution $\mathcal{N}(Q_i^{\text{pred}}, \sigma_i^{\text{pred}})$, and the gold quantity as $\mathcal{N}(Q_i^{\text{gold}}, \sigma_i^{\text{gold}})$. The standard deviations σ_i^{pred} and σ_i^{gold} are computed using the same formulation as in timeframe prediction.

4 FORECAST Construction

4.1 Data Source and Question Selection

FORECAST is constructed from Metaculus,[1] an online forecasting platform where forecasters submit probabilistic predictions to questions across various domains. Metaculus aggregates individual forecasts into a continuously updated community prediction, which is finalized just before resolution. Each question has predefined resolution criteria, ensuring verifiable outcomes. To ensure dataset reliability, we include only questions with a definitive resolution and at least 100 forecasts to maintain statistical reliability. Ambiguous or subjectively resolved questions are excluded, and we remove those whose outcomes depend on arbitrary or uncontrollable factors. These steps ensure that FORECAST consists of high-quality, well-formed forecasting questions with verifiable outcomes.

[1] www.metaculus.com. Examples in Appendix A.

4.2 Extracting Confidence from Crowdsourced Forecasts

Forecasting distinguishes between physical probabilities—objective likelihoods derived from statistical or scientific models—and human beliefs (Sanders, 1963) about future events. While physical probabilities can be useful, they are often unavailable, particularly for questions involving human behavior, economics, or sociopolitical outcomes. Instead, collective human forecasts offer a more practical confidence estimate, integrating expert reasoning, contextual knowledge, and evolving evidence. For instance, predicting a technological breakthrough depends more on expert assessment and current trends than on rigid probabilistic models. Therefore, confidence in FORECAST is derived from Metaculus community forecasts, which aggregate predictions from a diverse pool of forecasters. While human predictions are sometimes incorrect, they still serve as a valuable proxy for uncertainty, as they reflect the best available reasoning given the information at the time.

Formal Definition of Gold Confidence. Gold confidence in FORECAST is derived from the final Metaculus community prediction before resolution. Instead of directly using the predicted probability for the correct outcome, we compute a log score relative to a uniform baseline, ensuring that confidence reflects how much the forecast deviates from random guessing. This transformation prevents extreme probabilities in inherently uncertain scenarios and makes confidence scores more comparable across different

forecasting tasks. The final score is mapped to $(0, 1)$ using a sigmoid function.

For Boolean questions, where the human-forecasted probability for the correct outcome is P^{gold}, gold confidence is computed as:

$$C^{\text{gold}} = \sigma\left(\frac{\ln P^{\text{gold}} - \ln 0.5}{\ln 2}\right), \quad (10)$$

where $\sigma(x)$ is the sigmoid function.

Similarly, for timeframe prediction and quantity estimation, where the human-forecasted probability density function (PDF) assigns probability to a continuous outcome x^{gold}, gold confidence is computed as:

$$C^{\text{gold}} = \sigma\left(\frac{\ln f(x^{\text{gold}}) - \ln f_{\text{uniform}}}{2}\right), \quad (11)$$

where $f(x^{\text{gold}})$ is the forecasted probability density at the resolved outcome, and f_{uniform} is the uniform baseline density over the valid range of values. The denominator 2 ensures numerical stability and scales confidence appropriately.

4.3 Dataset Statistics and Comparison

FORECAST consists of 2256 forecasting questions, spanning domains such as politics, economics, science, and technology. Each question includes a resolved outcome, a gold confidence score, and a final Metaculus community forecast before resolution. To facilitate model development and evaluation, we split the dataset into 65% training, 10% validation, and 25% test. The full dataset statistics is shown in Appendix B.

Table 2 provides a comparison between FORECAST and existing forecasting benchmarks. Compared to prior datasets, FORECAST uniquely emphasizes both forecasting accuracy and confidence calibration, includes a diverse set of forecasting tasks, and is constructed from a well-established crowdsourced platform with rigorous resolution criteria.

Benchmark	Question Types	Natural Questions	Confidence
ForecastQA	MCQ	✓	✗
AutoCast	Various	✓	✗
ExpTime	Boolean	✗	✗
FORECAST	Various	✓	✓

Table 2: Comparison of key features across our benchmark variants, highlighting our evaluation of confidence across different question types.

5 Experiments on FORECAST

5.1 Experimental Setup

Models. We evaluate a diverse set of large language models (LLMs) with varying training data cutoffs, model sizes, and instruction tuning. To analyze the impact of knowledge recency, we group models by family and assume the cutoff date is the 1st of the stated month. The models used are shown in Table 3.

Model Family	Model Variants	Cutoff Date
GPT-2	GPT-2, GPT-2 XL	2017-12-01
Pythia	14M, 160M, 2.8B	2020-03-01
BLOOM	560M, 7B1	2021-12-01
LLaMA	LLaMA-7B	2022-08-01
OLMo	1B, 7B, 7B-Instruct	2023-03-01
OLMo-2	7B, 7B-Instruct	2023-12-01

Table 3: Models used in our experiments, grouped by family and ordered by training data cutoff date, including: GPT-2 (Radford et al., 2019), Pythia (Biderman et al., 2023), BLOOM (Scao et al., 2023), LLaMA (Touvron et al., 2023), OLMo (Groeneveld et al., 2024), and OLMo-2 (OLMo et al., 2024).

Inference. We use 1-shot in-context learning to provide models with a structured example of how to answer forecasting questions. For instruction-tuned models, we add an extra line of instruction to align with their training paradigm. To ensure fair comparison, baseline prompts are kept minimal while maintaining clarity. Confidence is estimated using logit-based normalized

heuristics. Full prompts and hyperparameters are provided in Appendix C and Appendix D.

Model	N	Accuracy (↑)	F1 (↑)	Brier (↓)
GPT2	401	0.5835	0.3650	0.4199
GPT2-XL	401	0.6708	0.3158	0.4540
Pythia-14m	343	0.5860	0.1374	0.6245
Pythia-160m	343	0.6093	0.2584	0.5273
Pythia-2.8b	343	0.5452	0.3906	0.4426
Bloom-560m	263	0.4905	0.4071	0.4521
Bloom-7b1	263	0.6426	0.3286	0.3200
Llama-7b	226	0.5708	0.3433	0.5201
OLMo-1B	188	0.2340	0.1591	0.7545
OLMo-7B	188	0.2074	0.1687	0.8199
OLMo-7B-Instruct	188	0.6543	0.1408	0.3811
OLMo-2-7B	145	0.5103	0.2444	0.4377
OLMo-2-7B-Instruct	145	0.5862	0.3023	0.4078

Table 4: Performance of forecasting models on Boolean questions in FORECAST. Reported metrics include the number of evaluated questions (N), accuracy, F1 score (both higher is better), and the Brier score (lower is better).

5.2 Results and Findings

Our experiments on the FORECAST dataset reveal that forecasting remains highly challenging for current LLMs, particularly in confidence estimation. While models achieve reasonable accuracy in point predictions, their uncertainty estimates—captured by Brier score and CRPS—vary significantly. This suggests that confidence evaluation must be treated as a separate challenge from prediction assessment.

Boolean Questions: Table 4 shows that while some models achieve reasonable accuracy, their calibration remains inconsistent. For example, `GPT2-XL` achieves an accuracy of 0.67 but has a relatively high Brier score of 0.45, while `OLMo-7B-Instruct` achieves similar accuracy (0.65) with a lower Brier score of 0.38, suggesting better confidence estimation. Notably, models with later training cutoffs do not always outperform older ones; for instance, `LLaMA-7B` has lower accuracy (0.57) and a worse Brier score (0.52) than some earlier models. This indicates that forecasting accuracy depends on multiple factors beyond knowledge recency.

Model	N	ADE (↓)	CRPS (↓)
GPT2	26	0.9944	0.9884
GPT2-XL	26	1.0000	1.0000
Pythia-14m	25	1.0000	1.0000
Pythia-160m	25	1.0000	1.0000
Pythia-2.8b	25	0.9650	0.9634
Bloom-560m	12	1.0000	1.0000
Bloom-7b1	12	0.9984	0.9976
Llama-7b	10	0.9843	0.9769
OLMo-1B	6	1.0000	1.0000
OLMo-7B	6	1.0000	1.0000
OLMo-7B-Instruct	6	1.0000	1.0000
OLMo-2-7B	4	0.9981	0.9970
OLMo-2-7B-Instruct	4	0.8696	0.8197

Table 5: Performance of forecasting models on Timeframe Prediction tasks in FORECAST. Metrics include the number of evaluated questions (N), the normalized absolute days error (ADE), and the Continuous Ranked Probability Score (CRPS), where lower values indicate better performance.

Timeframe Prediction: Table 5 reveals that most models struggle with predicting event timing, as both ADE and CRPS remain near the worst-case scenario of 1.0. Across all models, only one achieves a CRPS below 0.9: `OLMo-2-7B-Instruct` (CRPS = 0.82), which still indicates substantial uncertainty. Even models with more recent training data, such as `OLMo-2-7B` (CRPS = 0.99), fail to make well-calibrated temporal forecasts. These results suggest that even when models correctly anticipate whether an event will happen, quantifying when it will occur remains a major challenge.

Quantity Estimation: Table 6 highlights that models differ widely in their ability to esti-

Model	N	APE (↓)	MAE (↓)	CRPS (↓)
GPT2	81	0.2296	0.8697	0.8575
GPT2-XL	81	0.0295	0.7461	0.7169
Pythia-14m	77	0.1696	0.8753	0.8538
Pythia-160m	77	0.0274	0.7800	0.7620
Pythia-2.8b	77	0.0742	0.8106	0.7851
Bloom-560m	43	0.0462	0.7273	0.7082
Bloom-7b1	43	0.0644	0.7461	0.7176
Llama-7b	32	0.0564	0.6422	0.6126
OLMo-1B	23	0.2124	0.7883	0.7623
OLMo-7B	23	0.2157	0.8365	0.8201
OLMo-7B-Instruct	23	0.2358	0.7987	0.7703
OLMo-2-7B	20	0.0206	0.6457	0.6206
OLMo-2-7B-Instruct	20	0.0872	0.5968	0.5686

Table 6: Performance of forecasting models on Quantity Estimation tasks in FORECAST. Metrics include the number of evaluated questions (N), normalized absolute percentage error (APE), mean absolute error (MAE), and Continuous Ranked Probability Score (CRPS), where lower values are better.

mate numerical values. Some achieve relatively low APE, such as `GPT2-XL` (APE = 0.03) and `Pythia-160M` (APE = 0.02), yet their confidence calibration, CRPS, does not always align with their point prediction performance. For instance, `GPT2-XL` has a CRPS of 0.72, while `Pythia-160M` has a slightly higher CRPS of 0.76, despite achieving lower APE. In contrast, `OLMo-2-7B-Instruct`, which has a relatively higher APE of 0.08, achieves the lowest CRPS (0.57) among all models. These results indicate that prediction quality and confidence calibration do not necessarily improve together, reinforcing the complexity of numerical forecasting.

Impact of Model Size on Forecasting Performance Larger models do not consistently improve forecasting performance. Within the Pythia family, `Pythia-2.8b` shows occasional gains in accuracy and calibration over `Pythia-14m` and `Pythia-160m`, but the improvements are not uniform across all metrics. Similarly, while `OLMo-2-7B` variants sometimes achieve lower CRPS and MAE in quantity estimation, these benefits often come with trade-offs in point prediction. These results suggest that model size alone is not a reliable predictor of forecasting performance. This highlights the importance of developing task-specific techniques rather than relying on ever-larger models to solve the forecasting problem.

Impact of Instruction Tuning on Forecasting Performance Table 7 compares base and instruct-tuned variants of OLMo-7B and OLMo-2-7B across Boolean, timeframe, and quantity forecasting tasks. For Boolean questions, `OLMo-7B-Instruct` achieves higher accuracy (0.65 vs. 0.21) and a lower Brier score (0.38 vs. 0.82), indicating better confidence calibration. In timeframe prediction, the instruct-tuned `OLMo-2-7B-Instruct` improves uncertainty estimation, with an ADE of 0.87 and CRPS of 0.82, compared to 0.9981 and 0.9970 for the base `OLMo-2-7B`. For quantity estimation, instruct-tuned models have slightly higher APE but lower MAE and CRPS, suggesting better uncertainty calibration. These results indicate that instruction tuning enhances confidence estimation, even if it does not always improve point prediction accuracy. This suggests a trade-off where instruct-tuned models prioritize more reliable uncertainty quantification.

Impact of Aggregation Methods on Forecasting Performance Table 8 compares different aggregation methods for deriving the final prediction and confidence estimate from the top 10 outputs of `Llama-7B`. Bayesian Aggregation achieves the highest accuracy (0.5796) and F1 score (0.3485), suggesting it is the most effective at identifying correct outcomes. However, Weighted Average yields a significantly lower Brier score (0.2914), indicating superior confidence calibration compared to other methods.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
OLMo-7B	0.2074	0.1687	0.8199	1.0000	1.0000	0.2157	0.8365	0.8201
OLMo-7B-Instruct	0.6543	0.1408	0.3811	1.0000	1.0000	0.2358	0.7987	0.7703
OLMo-2-7B	0.5103	0.2444	0.4377	0.9981	0.9970	0.0206	0.6457	0.6206
OLMo-2-7B-Instruct	0.5862	0.3023	0.4078	0.8696	0.8197	0.0872	0.5968	0.5686

Table 7: Effect of instruction tuning on forecasting performance across boolean, timeframe, and quantity estimation questions. Metrics include accuracy, F1, and Brier score for binary questions; ADE and CRPS (T) for timeframe prediction; and APE, MAE, and CRPS (Q) for quantity estimation.

Aggregation Method	N	Accuracy (↑)	F1 (↑)	Brier (↓)
Majority Vote	226	0.5575	0.2353	0.5287
Highest Confidence	226	0.5708	0.3433	0.5201
Weighted Average	226	0.5575	0.2353	0.2914
Logit Mean Probability	226	0.5575	0.2353	0.6195
Bayesian Aggregation	226	0.5796	0.3485	0.5493

Table 8: Ablation study on different aggregation methods for extracting predictions and confidence from Llama-7B outputs.

Majority Vote, Highest Confidence, and Logit Mean Probability produce comparable accuracy and F1 scores but have noticeably higher Brier scores, suggesting weaker uncertainty estimation. These results highlight that even when point prediction performance is similar, aggregation methods substantially impact confidence reliability. The challenge remains in developing techniques that optimize both accuracy and calibration simultaneously, emphasizing the importance of uncertainty-aware forecasting.

6 Related Work

Recent forecasting benchmarks focus on event prediction but largely overlook confidence calibration. OpenForecast (Wang et al., 2025) introduces a large-scale dataset for open-ended, multi-step event forecasting but does not assess model confidence. ForecastBench (Karger et al., 2024) evaluates binary (Yes/No) forecasting by prompting models for direct probability estimates, but since it queries each option separately, the assigned probabilities do not necessarily sum to 1, leading to potential inconsistencies. Neither benchmark systematically evaluates confidence calibration, a crucial aspect for reliable forecasting in real-world applications.

Beyond forecasting, several benchmarks assess language models' reasoning and inference capabilities. COPA (Roemmele et al., 2011) evaluates causal reasoning by presenting a premise and two alternatives, requiring models to select the more plausible cause or effect. HellaSwag (Zellers et al., 2019) challenges models with sentence completion tasks that demand commonsense reasoning, where models must choose the most sensible continuation of a given scenario. PRobELM (Yuan et al., 2024b) assesses models' capacity to rank scenarios by plausibility, bridging the gap between factual accuracy and world knowledge. While these benchmarks provide insights into models' reasoning abilities, they do not address the challenges of forecasting future events.

7 Conclusion

We introduce FORECAST, a benchmark for evaluating both forecasting accuracy and confidence calibration in language models. Unlike existing datasets, FORECAST explicitly assesses confidence alongside predictions. Our results show that current models struggle with both prediction and well-calibrated confidence, underscoring the need for improved uncertainty estimation and confidence calibration.

Limitations

While FORECAST represents a significant step toward evaluating forecasting accuracy and confidence calibration, there are inherent limitations that we view as opportunities for future research rather than fundamental shortcomings of our work. First, our dataset is constructed solely from Metaculus, which may not fully represent the global diversity of forecasting practices or question domains. Second, our method for deriving gold confidence relies on community forecasts and heuristic transformations that might not capture all nuances of human uncertainty. Lastly, our focus on English-language forecasts limits the benchmark's applicability across different languages and cultural contexts. Addressing these issues is part of our future work agenda.

Ethical Statement

FORECAST is built from data sourced exclusively from Metaculus, an English-language forecasting platform. As a result, the dataset may embody the linguistic, cultural, and social biases inherent in its user community. These biases could affect both question selection and confidence judgments. We acknowledge these concerns and stress that our benchmark is intended as an initial step toward more inclusive forecasting evaluations. Future efforts should aim to incorporate data from a broader range of platforms and languages to mitigate these biases.

References

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, et al. 2023. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling. In *International Conference on Machine Learning*, pages 2397–2430. PMLR.

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

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. Boolq: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, et al. 2024. OLMo: Accelerating the Science of Language Models. *arXiv preprint arXiv:2402.00838*.

Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, Deyi Xiong, et al. 2023. Evaluating large language models: A comprehensive survey. *arXiv preprint arXiv:2310.19736*.

Woojeong Jin, Rahul Khanna, Suji Kim, Dong-Ho Lee, Fred Morstatter, Aram Galstyan, and Xiang Ren. 2020. Forecastqa: A question answering challenge for event forecasting with temporal text data. *arXiv preprint arXiv:2005.00792*.

Ezra Karger, Houtan Bastani, Chen Yueh-Han, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E Tetlock. 2024. Forecastbench: A dynamic benchmark of ai forecasting capabilities. *arXiv preprint arXiv:2409.19839*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring How Models Mimic Human Falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

James E. Matheson and Robert L. Winkler. 1976. Scoring rules for continuous probability distributions. *Management Science*, 22(10):1087–1096.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2024. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *2011 AAAI spring symposium series*.

Frederick Sanders. 1963. On subjective probability forecasting. *Journal of Applied Meteorology and Climatology*, 2(2):191–201.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. 2023. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. Working paper or preprint.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.

Zhen Wang, Xi Zhou, Yating Yang, Bo Ma, Lei Wang, Rui Dong, and Azmat Anwar. 2025. Openforecast: A large-scale open-ended event forecasting dataset. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 5273–5294.

Chenhan Yuan, Qianqian Xie, Jimin Huang, and Sophia Ananiadou. 2024a. Back to the future: Towards explainable temporal reasoning with large language models. In *Proceedings of the ACM on Web Conference 2024*, pages 1963–1974.

Moy Yuan, Eric Chamoun, Rami Aly, Chenxi Whitehouse, and Andreas Vlachos. 2024b. PRobELM: Plausibility ranking evaluation for language models. In *First Conference on Language Modeling*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a Machine Really Finish Your Sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy. Association for Computational Linguistics.

Andy Zou, Tristan Xiao, Ryan Jia, Joe Kwon, Mantas Mazeika, Richard Li, Dawn Song, Jacob Steinhardt, Owain Evans, and Dan Hendrycks. 2022. Forecasting future world events with neural networks. *Advances in Neural Information Processing Systems*, 35:27293–27305.

A Example Metaculus Questions

To illustrate how human forecasts evolve over time, we present two questions from different domains: Q1 is in the business and geopolitics domain and Q2 is in the technology domain.

> *Q1: Will TikTok become available in the US on both the App Store and Google Play before April 5, 2025?*
>
> *Q2: When will a SpaceX Starship reach orbit?*

For Q1, Figure 1 shows how community forecasts changed over time, while Figure 2 presents the histogram of the final forecast distribution.

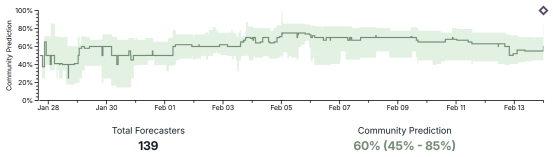

Figure 1: Community prediction trend for a Metaculus question on TikTok's availability in the US.

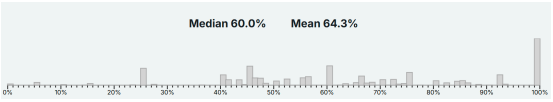

Figure 2: Histogram of final community forecasts.

For Q2, Figure 3 tracks forecast updates, while Figure 4 shows the final probability density function (PDF) of predicted launch dates.

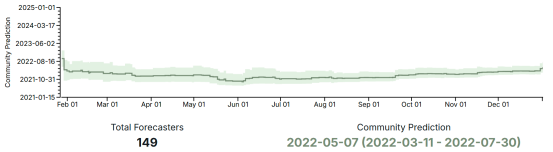

Figure 3: Community prediction trend for SpaceX Starship's first orbital launch.

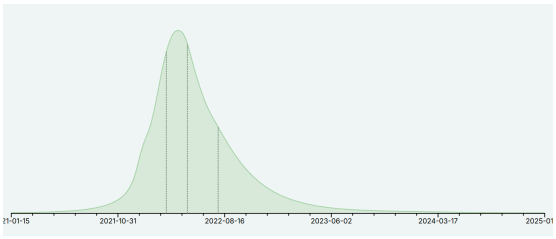

Figure 4: Probability density function of final community forecasts for SpaceX Starship reaching orbit.

Split	Boolean	Timeframe	Quantity	Total
Training	1142	90	223	1465
Validation	175	13	35	223
Test	441	36	91	568
Total	1758	139	349	2256

Table 9: Dataset statistics for FORECAST, showing the distribution of questions across different forecasting types, with the overall total in the last column.

B Dataset Statistics

Table 9 presents detailed dataset statistics, including the total number of questions and their distribution across Boolean Questions, Timeframe Prediction, and Quantity Estimation tasks.

C Prompts

To ensure a fair and consistent evaluation across models, we use simple one-shot prompts with structured outputs in JSON format. For instruction-tuned models, we provide an additional instruction line specifying the task. The prompts are designed for three forecasting question types: Boolean Questions (Yes/No), Timeframe Prediction (YYYY-MM-DD), and Quantity Estimation (numeric values).

C.1 Instruction-Tuned Models

For models containing "Instruct" in their name, we use the following prompts:

Quantity Estimation

```
You are an AI assistant providing precise numerical forecasts.
Answer the following question with a single numeric value in JSON format.

Example:
Q: How much global photovoltaic energy generation was deployed
   by the end of 2020?
A: { "value": 738 }

Q: $question
A: { "value": "
```

Timeframe Prediction

```
You are an AI assistant providing precise date forecasts.
Answer the following question with a single date in YYYY-MM-DD format in JSON.

Example:
Q: When did an AI system achieve a significant victory against
   a professional human in Starcraft 2?
A: { "value": "2019-01-24" }

Q: $question
A: { "value": "
```

Boolean Questions

```
You are an AI assistant providing binary (Yes/No) answers.
Answer the following question with "Yes" or "No" in JSON format.

Example:
Q: Will we confirm evidence for megastructures orbiting the
   star KIC 8462852?
A: { "value": "No" }

Q: $question
A: { "value": "
```

C.2 Base Models

For non-instruction-tuned models, we use the same examples but without additional instructions:

Quantity Estimation

```
Q: How much global photovoltaic energy generation was deployed
   by the end of 2020?
A: { "value": 738 }

Q: $question
A: { "value": "
```

Timeframe Prediction

```
Q: When did an AI system achieve a significant victory against
   a professional human in Starcraft 2?
A: { "value": "2019-01-24" }

Q: $question
A: { "value": "
```

Boolean Questions

```
Q: Will we confirm evidence for megastructures orbiting the
   star KIC 8462852?
A: { "value": "No" }

Q: $question
A: { "value": "
```

D Hyperparameter Settings

D.1 Generation Hyperparameters

We generate responses using temperature-based sampling with the following hyperparameters:

- `max_length` = 200
- `do_sample` = `True`
- `top_k` = 50
- `top_p` = 0.9

Among the generated outputs, we select the one with the highest confidence as the final prediction. All experiments are conducted using full precision on an NVIDIA RTX 8000 GPU.

D.2 Evaluation Hyperparameters

The scaling factor α in Equation 4, Equation 8, and Equation 9 is set to 0.05. For Equation 5 and Equation 6, we set $\sigma_{\max}$ to 30 and $\sigma_{\min}$ to 1 for Timeframe Prediction, while for Quantity Estimation, $\sigma_{\max}$ is 20 and $\sigma_{\min}$ is 1. These values ensure that evaluation metrics appropriately scale errors and confidence calibration.

E Additional Results

This section provides extended results categorized by the training data cutoff date of each model. Forecasting performance depends on model architecture, scale, and knowledge recency, so we evaluate models with different cutoff dates to examine how access to more recent information influences prediction accuracy and confidence calibration.

Models trained after certain event resolutions may have indirectly encountered outcome-related information, potentially affecting evaluation fairness. This should be considered when interpreting results.

Detailed model-specific performance metrics for Boolean Questions, Timeframe Prediction, and Quantity Estimation are presented in Table 10 to Table 15.

These results highlight trends in forecasting accuracy and confidence calibration across models with different knowledge recency.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
GPT2	0.5835	0.3650	0.4199	0.9944	0.9884	0.2296	0.8697	0.8575
GPT2-XL	0.6708	0.3158	0.4540	1.0000	1.0000	0.0295	0.7461	0.7169
Pythia-14m	0.5960	0.1548	0.6200	0.9998	0.9996	0.1754	0.8604	0.8399
Pythia-160m	0.6135	0.2537	0.5264	0.9998	0.9998	0.0267	0.7551	0.7387
Pythia-2.8b	0.5337	0.3746	0.4562	0.9661	0.9646	0.0758	0.7946	0.7708
Bloom-560m	0.4763	0.4199	0.4764	1.0000	1.0000	0.0585	0.7695	0.7533
Bloom-7b1	0.6284	0.3196	0.3514	0.9681	0.9549	0.0503	0.7645	0.7381
Llama-7b	0.5536	0.3678	0.5278	0.9390	0.9294	0.0837	0.7223	0.6999
OLMo-1B	0.2269	0.2041	0.7906	1.0000	1.0000	0.1521	0.8430	0.8245
OLMo-7B	0.2070	0.1739	0.7986	1.0000	1.0000	0.2693	0.8791	0.8687
OLMo-7B-Instruct	0.6658	0.3333	0.3871	0.9201	0.9138	0.2935	0.7997	0.7814
OLMo-2-7B	0.5362	0.3307	0.4995	0.9225	0.9081	0.0363	0.6983	0.6752
OLMo-2-7B-Instruct	0.5935	0.4240	0.3992	0.8407	0.8264	0.1281	0.6948	0.6725

Table 10: Combined forecasting performance for cutoff 2017-12-01. CRPS (T) denotes the Continuous Ranked Probability Score for Timeframe Prediction, while CRPS (Q) denotes the Continuous Ranked Probability Score for Quantity Estimation.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
GPT2	0.5918	0.3750	0.4136	0.9942	0.9879	0.2342	0.8852	0.8749
GPT2-XL	0.6735	0.3190	0.4465	1.0000	1.0000	0.0277	0.7667	0.7366
Pythia-14m	0.5860	0.1374	0.6245	1.0000	1.0000	0.1696	0.8753	0.8538
Pythia-160m	0.6093	0.2584	0.5273	1.0000	1.0000	0.0274	0.7800	0.7620
Pythia-2.8b	0.5452	0.3906	0.4426	0.9650	0.9634	0.0742	0.8106	0.7851
Bloom-560m	0.5015	0.4393	0.4562	1.0000	1.0000	0.0481	0.7832	0.7651
Bloom-7b1	0.6297	0.3280	0.3477	0.9668	0.9531	0.0508	0.7877	0.7601
Llama-7b	0.5510	0.3677	0.5311	0.9365	0.9266	0.0634	0.7239	0.7012
OLMo-1B	0.2332	0.1928	0.7843	1.0000	1.0000	0.1568	0.8696	0.8491
OLMo-7B	0.2041	0.1538	0.8023	1.0000	1.0000	0.2568	0.8855	0.8731
OLMo-7B-Instruct	0.6647	0.2968	0.3975	0.9169	0.9103	0.2746	0.8002	0.7818
OLMo-2-7B	0.5335	0.3256	0.5057	0.9194	0.9045	0.0267	0.7097	0.6848
OLMo-2-7B-Instruct	0.5948	0.4085	0.3987	0.8343	0.8195	0.1226	0.7044	0.6821

Table 11: Combined forecasting performance for cutoff 2020-03-01. CRPS (T) denotes the Continuous Ranked Probability Score for Timeframe Prediction, while CRPS (Q) denotes the Continuous Ranked Probability Score for Quantity Estimation.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
GPT2	0.6084	0.3522	0.3754	1.0000	1.0000	0.2649	0.8513	0.8384
GPT2-XL	0.6730	0.2586	0.4463	1.0000	1.0000	0.0313	0.7427	0.7035
Pythia-14m	0.6008	0.1319	0.6531	1.0000	1.0000	0.2187	0.8598	0.8376
Pythia-160m	0.6312	0.2400	0.5190	1.0000	1.0000	0.0220	0.7278	0.7016
Pythia-2.8b	0.5171	0.3280	0.4376	1.0000	1.0000	0.0686	0.7692	0.7412
Bloom-560m	0.5015	0.4393	0.4521	1.0000	1.0000	0.0481	0.7273	0.7082
Bloom-7b1	0.6297	0.3280	0.3200	0.9984	0.9976	0.0644	0.7461	0.7176
Llama-7b	0.5741	0.3625	0.5184	0.9869	0.9807	0.0473	0.6532	0.6232
OLMo-1B	0.2395	0.1849	0.7721	1.0000	1.0000	0.2159	0.8414	0.8203
OLMo-7B	0.2243	0.1754	0.7954	1.0000	1.0000	0.2253	0.8482	0.8354
OLMo-7B-Instruct	0.6654	0.1782	0.4045	0.9234	0.9205	0.2189	0.7629	0.7372
OLMo-2-7B	0.5323	0.2692	0.4872	0.9266	0.9221	0.0270	0.6676	0.6427
OLMo-2-7B-Instruct	0.6084	0.3758	0.3858	0.8590	0.8385	0.1039	0.6382	0.6114

Table 12: Combined forecasting performance for cutoff 2021-12-01. CRPS (T) denotes the Continuous Ranked Probability Score for Timeframe Prediction, while CRPS (Q) denotes the Continuous Ranked Probability Score for Quantity Estimation.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
GPT2	0.6150	0.3556	0.3589	1.0000	1.0000	0.2576	0.8125	0.7984
GPT2-XL	0.6814	0.2653	0.4392	1.0000	1.0000	0.0345	0.7430	0.6980
Pythia-14m	0.6062	0.1558	0.6537	1.0000	1.0000	0.1946	0.8696	0.8472
Pythia-160m	0.6504	0.2524	0.5285	1.0000	1.0000	0.0221	0.7275	0.6998
Pythia-2.8b	0.5177	0.3230	0.4470	1.0000	1.0000	0.0843	0.7769	0.7435
Bloom-560m	0.5044	0.3978	0.4308	1.0000	1.0000	0.0528	0.7232	0.7058
Bloom-7b1	0.6460	0.3333	0.3061	0.9981	0.9971	0.0796	0.7765	0.7449
Llama-7b	0.5708	0.3433	0.5201	0.9843	0.9769	0.0564	0.6422	0.6126
OLMo-1B	0.2257	0.1800	0.7762	1.0000	1.0000	0.2206	0.8267	0.8032
OLMo-7B	0.2257	0.1702	0.8156	1.0000	1.0000	0.2238	0.8724	0.8575
OLMo-7B-Instruct	0.6460	0.1176	0.4043	0.9081	0.9046	0.2058	0.7799	0.7512
OLMo-2-7B	0.5398	0.2576	0.4698	0.9119	0.9065	0.0300	0.6814	0.6535
OLMo-2-7B-Instruct	0.5973	0.3259	0.3992	0.9283	0.8983	0.1038	0.6517	0.6216

Table 13: Combined forecasting performance for cutoff 2022-08-01. CRPS (T) denotes the Continuous Ranked Probability Score for Timeframe Prediction, while CRPS (Q) denotes the Continuous Ranked Probability Score for Quantity Estimation.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
GPT2	0.5957	0.3559	0.3800	1.0000	1.0000	0.2642	0.7407	0.7222
GPT2-XL	0.6543	0.2353	0.4183	1.0000	1.0000	0.0383	0.7087	0.6530
Pythia-14m	0.6011	0.1562	0.6417	1.0000	1.0000	0.2195	0.8489	0.8243
Pythia-160m	0.6330	0.2581	0.4971	1.0000	1.0000	0.0220	0.6945	0.6611
Pythia-2.8b	0.5213	0.3382	0.4390	1.0000	1.0000	0.1080	0.7581	0.7199
Bloom-560m	0.5266	0.4183	0.4124	1.0000	1.0000	0.0654	0.6986	0.6713
Bloom-7b1	0.6649	0.3883	0.2964	0.9968	0.9952	0.1022	0.7897	0.7574
Llama-7b	0.5532	0.3214	0.5113	1.0000	1.0000	0.0566	0.6014	0.5689
OLMo-1B	0.2340	0.1591	0.7545	1.0000	1.0000	0.2124	0.7883	0.7623
OLMo-7B	0.2074	0.1687	0.8199	1.0000	1.0000	0.2157	0.8365	0.8201
OLMo-7B-Instruct	0.6543	0.1408	0.3811	1.0000	1.0000	0.2358	0.7987	0.7703
OLMo-2-7B	0.5106	0.2414	0.4438	0.9987	0.9980	0.0214	0.6464	0.6160
OLMo-2-7B-Instruct	0.5851	0.3276	0.4107	0.9100	0.8739	0.1150	0.6067	0.5748

Table 14: Combined forecasting performance for cutoff 2023-03-01. CRPS (T) denotes the Continuous Ranked Probability Score for Timeframe Prediction, while CRPS (Q) denotes the Continuous Ranked Probability Score for Quantity Estimation.

Model	Accuracy (↑)	F1 (↑)	Brier (↓)	ADE (↓)	CRPS (T) (↓)	APE (↓)	MAE (↓)	CRPS (Q) (↓)
GPT2	0.5931	0.3059	0.3673	1.0000	1.0000	0.2490	0.7319	0.7167
GPT2-XL	0.7103	0.3000	0.4207	1.0000	1.0000	0.0405	0.6887	0.6392
Pythia-14m	0.6207	0.1395	0.6479	1.0000	1.0000	0.2487	0.8481	0.8293
Pythia-160m	0.6690	0.3143	0.5090	1.0000	1.0000	0.0218	0.6731	0.6463
Pythia-2.8b	0.5241	0.2887	0.4481	1.0000	1.0000	0.1189	0.7315	0.6914
Bloom-560m	0.5241	0.3784	0.4103	1.0000	1.0000	0.0262	0.6740	0.6479
Bloom-7b1	0.6828	0.3611	0.3007	1.0000	1.0000	0.1140	0.7831	0.7528
Llama-7b	0.5379	0.2716	0.5452	0.9621	0.9443	0.0632	0.6173	0.5901
OLMo-1B	0.2345	0.2059	0.7507	1.0000	1.0000	0.2195	0.7639	0.7370
OLMo-7B	0.2207	0.2333	0.8271	1.0000	1.0000	0.2195	0.8218	0.8066
OLMo-7B-Instruct	0.6828	0.1569	0.3642	1.0000	1.0000	0.2682	0.8082	0.7842
OLMo-2-7B	0.5103	0.2444	0.4377	0.9981	0.9970	0.0206	0.6457	0.6206
OLMo-2-7B-Instruct	0.5862	0.3023	0.4078	0.8696	0.8197	0.0872	0.5968	0.5686

Table 15: Combined forecasting performance for cutoff 2023-12-01. CRPS (T) denotes the Continuous Ranked Probability Score for Timeframe Prediction, while CRPS (Q) denotes the Continuous Ranked Probability Score for Quantity Estimation.