Scaling Open-Ended Reasoning to Predict the Future

Anonymous Author(s)
Affiliation
Address
email

Abstract

While language models now show remarkable capabilities on fully specified exam-style problems, most real-world decisions involve reasoning under uncertainty. In this work, we train language models to make predictions on open-ended questions about the future. To scale up training data, we use daily news to synthetically generate forecasting questions about global events that have already occurred. Using Reinforcement Learning (RL), we train an 8B parameter model on our dataset, `NewsCast`, achieving predictions at par with OpenAI's much larger GPT-OSS 120B. RL training on `NewsCast` not only improves both accuracy and calibration for forecasting, it also reduces hallucinations, as measured using SimpleQA. Our findings demonstrate the usefulness of goal-oriented synthetic data generation pipelines for training language models to predict the future. We will open-source our models, code, and data to make LLM forecasting research broadly accessible.

1 Introduction

Every day, people navigate decisions under incomplete evidence, competing hypotheses, and high uncertainty. The highest-stakes choices are inherently forward-looking: governments set policy while anticipating macroeconomic and geopolitical shifts; investors allocate capital amid market and regulatory uncertainty; individuals choose careers as technologies evolve; and scientists pursue research directions in search of the next breakthrough. While LLMs now show impressive performance on classical exam-style benchmarks, how to train them to reason about uncertainty remains uncertain.

In this work, we show how to scalably synthesize daily news data to train LLMs at predicting the future. Detailed and correct reasoning traces for forecasting are costly to obtain and hard to verify, as this task is extremely hard for humans. We make a simple observation: open-ended forecasting questions about world events eventually resolve. This provides the verifiable signal necessary to leverage the recent success of Reinforcement Learning (RL) for LLMs. However, sourcing questions at scale for training forecasting has a few key challenges. First, waiting for events to resolve provides too slow a feedback loop for training. Second, prediction markets–the primary source for existing forecasting questions–mostly consist of binary yes or no questions, which provide noisy rewards as there is a 50% chance of success even with wrong reasoning. The static knowledge cutoff of LLMs enables a unique arbitrage: resolved events thereafter are in the future for an LLM. We collect global news from high-quality sources between the cutoff date and today. From these news articles, we build an automated synthetic curation pipeline that uses LLMs to generate open-ended forecasting questions at scale, where we design model-based curation stages where an LLM reflects, filters, and rewrites for clarity and verifiability. Using this pipeline, we create `NewsCast`, a dataset consisting of 60K open-ended questions. We judge free-form forecasting responses to these questions using answer matching, which has recently been shown to be reliable for automated evaluation [Chandak et al., 2025]. We pair it with a reward metric which incentivizes maximally accurate and calibrated predictions for open-ended outcomes [Damani et al., 2025] and train models with GRPO for forecasting.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

Empirically, even starting from strong Qwen3-thinking backbones (1.7B–8B), RL training on `NewsCast` yields large gains across a range of forecasting evaluations – from free-form responses to binary outcomes. RL-trained models improve on both accuracy and Brier scores and match GPT-OSS, a recent 120B model from OpenAI. We further show that improvements are not solely due to better calibration: It is largely due to increase in accuracy, indicating that models learn to generate more informative outcome spaces while assigning calibrated probabilities. As a downstream benefit, we observe reduced hallucination rates on the SimpleQA benchmark, consistent with the intuition that calibrating uncertainty helps models abstain or hedge appropriately when evidence is weak.

Decades of work on human forecasting show that skill varies widely, yet training to forecast better is possible, as some "superforecasters" consistently outperform peers [Tetlock and Gardner, 2016]. While there is a ceiling to predictability in complex systems— we do not yet know where that ceiling lies. Once trained at scale for this task, LLMs enjoy structural advantages over humans: they can ingest and synthesize vast, heterogeneous corpora across thousands of topics; and update predictions rapidly as new text arrives. Thus, the open-endedness of forecasting questions is a key part of our contribution. It allows training LLMs to come up with possibilities humans might miss. Further, it requires reasoning about the probability of each possible outcome by weighing competing mechanisms. We believe this paradigm can enable reasoning under uncertainty, reconciling conflicting evidence, and updating beliefs while being calibrated as information arrives. We expect sufficient scaling of reasoning to predict the future will lead to improved world models of societal dynamics. Forecasting systems, if realized responsibly, could transform policy analysis, corporate planning, and financial risk management by improving rigorous open-ended predictions. To make LLM forecasters broadly available, we will open-source all artifacts–including models, code and data.

2 Generating Open-Ended Forecasting Questions from News

Background. LLM weights are frozen after training, especially when the weights are released openly. Any event that happened between the last date in their training corpus is in the future for an LLM. This provides a window to collect questions for training to reason about future events. Prior work on *judgemental forecasting*[1] has used prediction market questions [Halawi et al., 2024, Turtel et al., 2025b] to improve LLMs at forecasting. Prediction markets consist of questions human-generated questions like "Will Donald Trump win the US Presidential Election in 2024?" on which a crowd of people make predictions. They have grown in popularity rapidly over the past few years [Paleka et al., 2025]. There are two key problems with relying on prediction market questions for training. First, the questions are created by humans, which makes them low in number [Paleka et al., 2025]. This becomes a bottleneck for scaling training data, which has been an essential component in the success of LLMs [Kaplan et al., 2020]. Second, the questions have binary outcomes, which creates a 50% baseline success rate. This means even incorrect reasoning has a high chance of being reinforced. This leads to noisy rewards in outcome-based RL. These limitations motivate us to explore alternate ways to create forecasting questions.

2.1 Training Data Generation Pipeline

We now describe our synthetic training data generation pipeline, starting with the general framework and then providing details about our dataset–`NewsCast`. We generate short-answer, open-ended forecasting questions from individual news articles using a generator–verifier–editor pipeline, illustrated in Fig. 1. We describe each step in detail below:

Source Documents. News outlets are an established global engine for reporting salient events as they occur. They mobilize vast resources to ensure information is presented fast and accurately. The CommonCrawl News (CCNews) Corpus collects openly licensed news from across the world [Nagel, 2016], making it free and easy to obtain for creating forecasting questions. Prior work has shown that date cutoffs are unreliable with search engines, and this can compromise backtests by leaking future information [Paleka et al., 2025]. Reliable backtests are essential for iterating fast when improving a forecasting system. Moreover, deep learning models are known to fit well to spurious correlations, and are likely to exploit any such leakage in training, which would then break on deployment. Crucially, CCNews provides static monthly snapshots of news, which are not updated further. We use this offline

[1] not to be confused with time series forecasting, which focuses on structured data

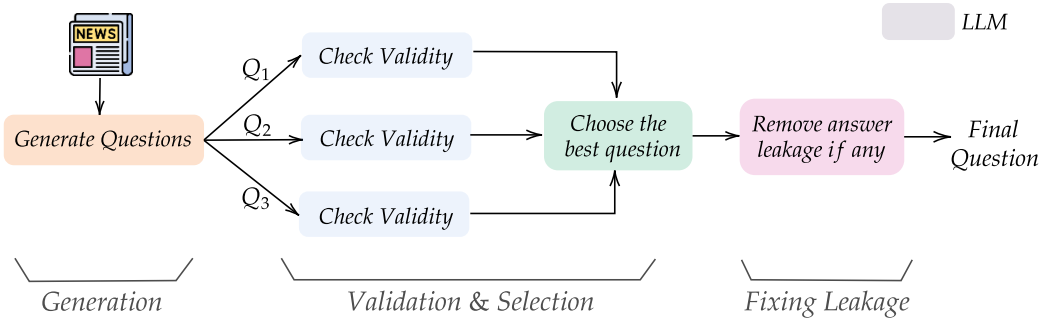


Figure 1: **Our Question Generation Pipeline.** We use DeepSeek v3 to generate multiple forecasting questions per news article. Then, we use a different model, Llama-4-Maverick, to check if questions follow all guidelines, choose the best question, and remove any hints that give away the answer.

resource to ensure leakeg of future information [Paleka et al., 2025], whether it be updates to articles, or changes in search engine indexing, does not compromise our training. One practical issue we face is that many news outlets, such as The Retuers and Associated Press (AP), have disallowed scraping even for CCNews due to the rise of commercial use in language model training [Grynbaum and Mac, 2023]. This limits the reliable news outlets available to us for generating forecasting questions.

Document to Question. We provide a language model an article, and ask it to generate up to three diverse forward-looking forecasting question-answer pairs based on the article. As illustrated below, each generated sample consists of: (i) a concise question about an event with an explicit deadline (e.g., "by *Month, Year*"); (ii) brief background that introduces the question, provides context, and defines terms which are not popular; (iii) resolution criteria that name a source of truth and the expected answer format; (iv) An answer which is short (1–3 words), non-numeric (usually a name, date, or location), unique, and recoverable verbatim from the article; and (v) Source article link for reference.

Sample Generated Forecasting Question

Question. Which streaming platform will receive the most Oscar nominations by January 2024?

Background. Question Start Date: 10th January 2024. Streaming platforms have become increasingly prominent in film production and awards recognition.

Resolution Criteria.

- **Source of Truth**: The question will resolve based on the official nomination list released by the Academy on January 23, 2024.
- **Resolution Date**: The resolution occurs on the calendar date of the nomination announcement (January 23, 2024).
- **Accepted Answer Format**: The name of the streaming platform exactly as recognized by the Academy.

Answer Type. String (Platform Name)

Ground-Truth Answer. Netflix

Source. CNN: Emma Stone reacts to Oscar nominations

Validation and selection. For each question, we use another LLM to verify the following properties: (i) the question-answer pair is fully based on information in the source article (ii) the question is in future tense and (iii) the answer is definite, unambiguous, and resolvable by the publication date. We mark a question as valid only if it passes these checks. To reduce cross-contamination between questions, for articles with more than one valid question, we then use another model to select the best one. We ask it to favor unique answers, high relevance, and clear resolution.

Fixing leakage. At this stage, we find that even the filtered best questions sometimes leak the answer in the answer format or resolution criteria. This can create shortcuts during training. We use an LLM to scan the title, background, and resolution criteria for direct or indirect mentions of the true answer. When it finds leakage, we ask it to rewrite only the offending spans, replacing specifics with generic placeholders. We then re-scan for any remaining mentions of the answer string, and discard those question-answer pairs.

Overall, this pipeline can continually ingest daily news articles and generate high-quality open-ended forecasting questions.

2.2 `NewsCast`: Creating an Open, Large-Scale Training Dataset

We now describe the specific composition of our released training dataset–`NewsCast`.

Question Generation. We start from a broad, heterogeneous pool of news articles diversified by geography, time, and topic. Concretely, we collect ∼250K English-language articles from outlets such as *Forbes*, *Hindustan Times*, *Irish Times*, *Deutsche Welle*, and *CNN*. The corpus spans sports, geopolitics, local news, crime, entertainment, and arts. It covers June 2023 through April 2025 and includes many major large-scale events. After de-duplication and filtering for English, availability of article text, and dates, we retain 248K articles for generating questions. The distribution is described in Table 2. From these 248K articles we generate three forecasting-style questions per article using DeepSeek v3, yielding 745K synthetic question–answer candidates.

Question Filtering. For all further data filtering, we use a different model, Llama-4-Maverick to prevent leniency due to LLM self-preference [Xu et al., 2024]. Table 1 contains a breakdown of questions remaining after each filtering stage. 60% of question-answer candidates are marked invalid — most commonly because the article does not unambiguously resolve the question to the given answer. Figure 2 shows that after this stage, zero questions remain from 40% of source articles. Among the remaining articles, 21% yield exactly one valid question, which we keep as is. For the 39% with multiple valid questions, we use an LLM to pick the best one based on globally relevance, specificity, and unambiguity. Despite explicit prompts to avoid it, over 40% of selected questions contain *direct* answer leakage. Using an LLM to identify and rewrite or reject questions with leakage, we are able to remove ∼90% of cases with leakage. We then apply a string matching filter to remove 4% of the remaining questions which directly contain the answer string. Finally, we remove numeric questions as our current reward is focused on open-ended string responses which are matched with the ground truth answer.

Stage	**Number (%Total)**
Question Generation	744,963 (100%)
Validation	295,274 (40%)
Best Question Selection	157,260 (21%)
Fixing Leakage	150,500 (20%)
Answer Type Filtering	62,279 (8%)
Final Set	62,279 (8%)

Table 1: Number of questions after each filtering stage.

Resulting dataset. Across stages, we remove roughly 90% of initial candidates, yielding a high-precision set of 62K question-answer pairs drawn each drawn from a unique article. We release this corpus as `NewsCast`, an open, large-scale training dataset for LLM forecasting.

3 RL Training

We train LLMs using reinforcement learning on our compiled dataset. Let $\mathcal{X}$ be a set of prompts (news-derived forecasting questions; §2) and $\mathcal{Y}$ the set of short textual answers. Each data sample $(x, y^\star)$ consists of a question x and its true answer $y^\star$ that is uniquely recoverable from its source article. A language model π_θ maps x to a completion containing two tagged fields

$$\texttt{<answer>}\ y \in \mathcal{Y}, \quad \texttt{<probability>}\ p \in [0, 1],$$

where p is the model's verbalized probability for y being the outcome of the forecasting question x. Since the output is open-ended and free-form text, we score correctness with *answer matching*: an external judge LM J receives $(x, y^\star, \hat{y})$ and returns $J(\cdot) \in \{0, 1\}$ indicating semantic equivalence.

For RL training, we use `Qwen3-4B` in non-thinking mode as J given its small size and strong alignment with human graders for evaluating free-form responses [Chandak et al., 2025].

Given a dataset $D = \{(x_i, y_i^\star)\}_{i=1}^N$ with a reward function R, our RL objective is

$$\max_\theta \ \mathbb{E}_{(x,y^\star)\sim D}\, \mathbb{E}_{(y,p)\sim\pi_\theta(\cdot|x)}\big[R(y,p,y^\star)\big]. \tag{1}$$

Training Reward: Free-form Brier. We adapt the multi-class brier score for free-form answers, originally defined for categorical outcomes, in the following manner:

$$\begin{aligned} \mathbf{BS}_{\textbf{Sample}}(y,p;y^\star) &= \begin{cases} p^2, & \text{if } y = y^\star \\ -p^2, & \text{if } y \neq y^\star \end{cases} \\ \textbf{Free-form Brier Score}: \quad \mathrm{BS}_{\mathrm{Free}}(y,p;y^\star) &= \tfrac{1}{N}\sum_{n=1}^{N} \mathbf{BS}_{\textbf{Sample}}(y_i,p_i;y_i^\star) \end{aligned} \tag{2}$$

Concurrent work [Damani et al., 2025] shows that free-form brier as defined above is a proper scoring rule, incentivizing truthful reporting of probability on the answer that seems most likely.

Training Algorithm: GRPO [Shao et al., 2024]. For each prompt x, we draw K completions $\{(y_i,p_i)\}_{i=1}^K \sim \pi_\theta(\cdot \mid x)$ and compute rewards $r_i = R(y_i,p_i;y^\star)$. However, following prior work [Damani et al., 2025, Turtel et al., 2025b], we *remove* the per-group standard-deviation division during the advantage computation:

$$\mu = \tfrac{1}{K}\sum_{i=1}^{K} r_i, \qquad A_i = r_i - \mu, \tag{3}$$

as it stabilizes updates in cases where reward variance is too small or large, common in our setting.

4 Experiments

Evaluation Datasets. To eliminate any risk of information leakage in evaluation, we restrict our test sets to questions dated *May 2025* or later. To evaluate on human-authored questions with real incentives, we curate a Metaculus benchmark covering **May–July 2025**. We filter out unresolved or low-activity questions using a minimum trading-volume criterion and retain **244** high-interest binary questions. Because all items are binary, a class-agnostic baseline achieves 50% accuracy. We report *accuracy* and *Brier score* (mean squared error of predicted probabilities).

We also construct a news-based forecasting test set consisting of 207 freeform forecasting questions using the same pipeline (Section 2.1) based on a uniform sample of 500 articles from TheGuardian published in **July 2025**. Each question includes the same structured fields as in our training samples. An example is shown in Box A. For these free-form questions, we measure accuracy (defined as: $\text{Acc} = \frac{1}{N}\sum_{n=1}^{N} o\big(y_n^\star, \hat{y}_n\big)$) and freeform brier score (see equation 2). During evaluation, we compute the outcomes $o(y,y^\star) = \mathbf{1}[y \equiv y^\star] \in \{0,1\}$ using `Llama-4-Scout` instead of `Qwen3-4B` against which we train. This helps us ensure our results robust to any judge model. `Llama-4-Scout` is also known to have high alignment with human graders for evaluating free-form responses [Chandak et al., 2025].

Training. We train Qwen3 [Yang et al., 2025] thinking models ranging from 1.7 to 8B parameters. No official knowledge–cutoff date for Qwen-3 is reported. When queried directly, the models returns inconsistent cutoff dates (most often *October 2023* or *June 2024*). Although the family was released in April 2025, the models frequently treat events from 2024 as future, suggesting a practical cutoff date. This behavior is acceptable for training—even when some prompts refer to events that lie in the model's past. Unless stated otherwise, we train on the full corpus of **62,279** questions generated by our pipeline (Section 2.1) and also include 2000 binary questions from prediction markets as we found them to uplift model performance in binary format (shown in Section 4.1). The samples are sorted by their question resolution date and this order is preserved during training. More details are provided in Appendix Section A.1.

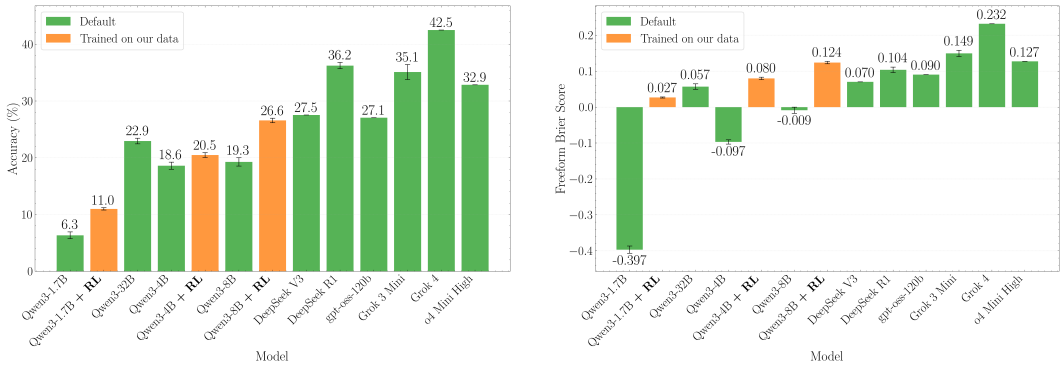


Figure 2: **Performance of the models on our test set derived from TheGuardian.** The left panel shows the accuracy of the models while the right panel shows the freeform brier score measuring both calibration and correctness. Training on our data improves Qwen3 models across the board with the 8B models achieving high brier score matching gpt-oss 120b.

4.1 Results

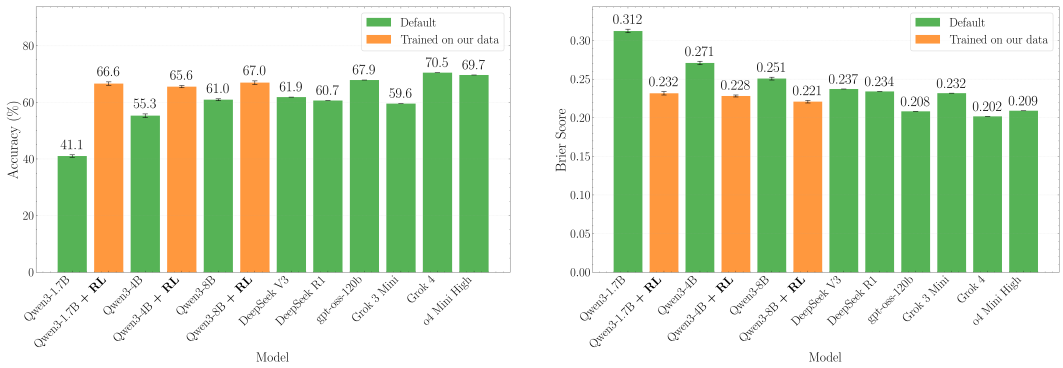


Figure 3: **Performance of the models on questions between May to July 2025 from Metaculus.** The left panel shows the accuracy of the models while the right panel shows the brier score measuring both calibration. We significantly improve all Qwen3 thinking models.

Main results on free-form forecasting. Figure 2 reports accuracy (left) and free-form Brier score (right) on the *The Guardian* July 2025 test set. Training on `NewsCast` with GRPO consistently improves calibration and correctness across all Qwen3 backbones. The gains are largest in Brier score: `Qwen3-1.7B` moves from a strong *negative* brier (over-confident and often wrong) to near-zero, `Qwen3-4B` improves from -0.097 to $+0.080$, and `Qwen3-8B` from -0.009 to $+0.124$. Accuracy also improves significantly (e.g., `Qwen3-8B`: $19.3\% \rightarrow 26.6\%$) while the major effect is better-calibrated predictions that the Brier score captures. These results indicate that reward learning with free-form Brier not only corrects systematic miscalibration but also incentivizes the model to explore correct outcomes which is observed with the gain in accuracy.

Comparison to larger closed and open models. Figures 3 (binary accuracy/brier on Metaculus) and 2 (free-form Brier on news) include stronger non-Qwen baselines. On free-form Brier (Fig. 2, right), `Qwen3-8B + RL` (0.124) *surpasses* `GPT-OSS-120B` (0.090) and `DeepSeek V3/R1` (0.070/0.104), is competitive with `o4 Mini High` (0.127), and trails `Grok 3 Mini` (0.149) and `Grok 4` (0.232). On binary Metaculus (Fig. 3, left), RL yields large accuracy gains for smaller backbones (e.g., `Qwen3-1.7B`: $41.1\% \rightarrow 66.6\%$, `Qwen3-4B`: $55.3\% \rightarrow 65.6\%$, `Qwen3-8B`: $61.0\% \rightarrow 67.0\%$). The right panel shows consistent Brier improvements, indicating that accuracy gains are not bought at the expense of overconfidence. While our RL-trained models surpass some larger open-weight models like DeepSeek-v3, they still lag significantly behind frontier closed-source models.

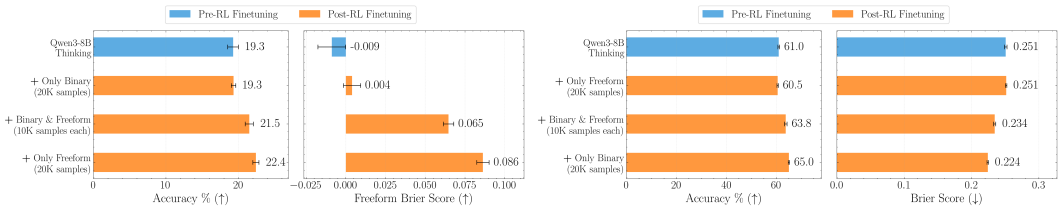


Figure 4: **Performance of different data ablations.** We evaluate 3 different ablations — 1) only binary data (20K samples) 2) only freeform data (20K samples) and 3) both binary and freeform data (but 10K samples each) for data-matched comparison. **Left:** Accuracy and freeform brier score of the initial and post-RL model on our freeform test set from July 2025. **Right:** Accuracy and binary brier score of initial and post-RL model on volume-filtered binary questions resolved between May to July 2025 on Metaculus. We find training on binary questions hurts performance on open-ended forecasting, but is necessary to retain performance on binary prediction market questions.

Ablation: Comparison to Prediction Market Binary Data We ablate supervision type with `Qwen3-8B` using three size-matched settings (Figure 4). For *binary-only*, we curate **20K** resolved markets from Manifold, volume-filtered to ensure engagement; because many markets resolve slowly, this set spans the past five years. For *free-form only*, we use **20K** pipeline-generated, usable questions from Forbes articles. For the *binary+free-form mix*, we take **10K** Manifold + **10K** Forbes questions to keep total examples constant. The goal is to isolate which *learning signal*—binary resolution vs. open-ended outcome specification—most effectively trains calibrated forecasters under identical compute and token budgets.

On the free-form test set (Fig. 4 Left), post-RL performance improves most with *free-form only* supervision (Accuracy $19.3\% \rightarrow 22.4\%$; Free-form Brier $-0.009 \rightarrow 0.086$). Mixing binary and free-form also helps (Brier 0.065), whereas *binary-only* yields minimal gains on free-form evaluation (Brier 0.004). On Metaculus (binary) (Fig. 4 Right), both *binary-only* and the *mixed* setting improve accuracy and Brier, with the *binary+free-form* mix offering the best overall trade-off across testing formats. Our gains by training on binary-only format are consistent with prior work by Turtel et al. [2025b,a]. However, we do not arrive at a single unanimous recipe: free-form data is essential for open-ended forecasting, while combining formats appears Pareto-optimal across binary and free-form evaluations. Practically, it seems training on a *mixture* of question styles provides the most robust gains across tasks.

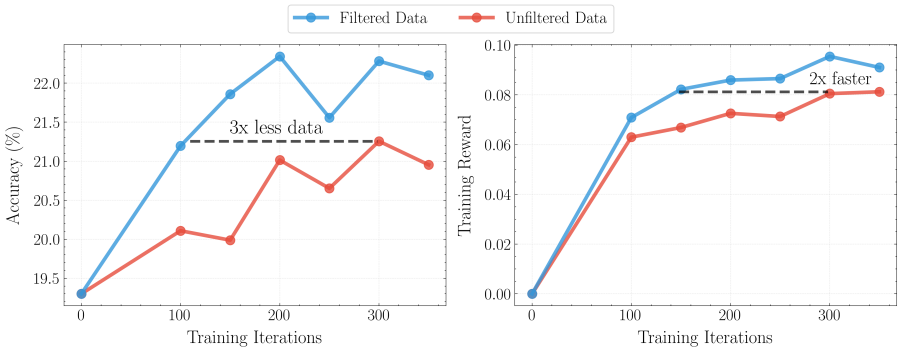


Figure 5: **Benefits of our question filtering pipeline**. We apply the same leakage checks to both the filtered and unfiltered set. We find that our question validation pipeline that filters questions generated by the initial model leads to 2 times faster training and higher overall scores at the end.

Ablation: Effect of filtering pipeline We quantify the effect of our generator–verifier–editor pipeline (Section 2.1) by training on *filtered* versus *unfiltered* data drawn from the same Forbes article pool (Figure 5). We first select 10,000 items that pass *all* stages of the pipeline (entailment/answerability checks, style normalization, and answer-type validation). For the unfiltered baseline, we go back to each source article and collect the *three raw candidate questions* produced at the very start of the pipeline, yielding a **30,000**-item corpus. To isolate the contribution of validation and selection (as opposed to leakage guards), we apply the *same* leakage-removal edits to the unfiltered set as we

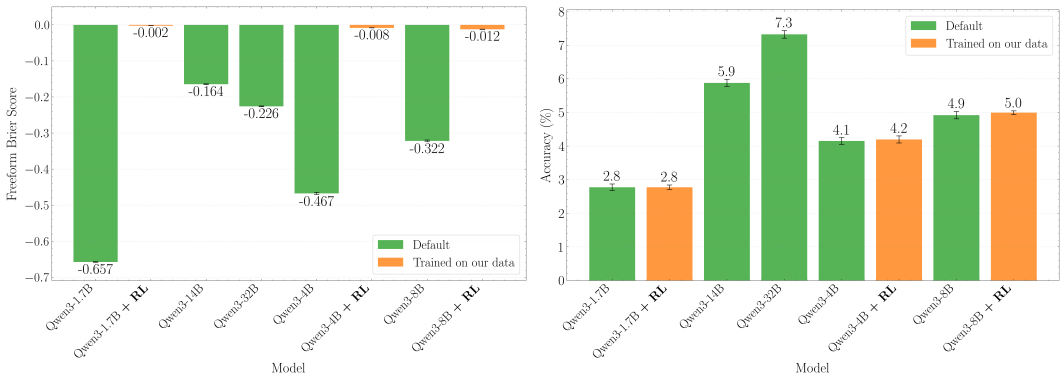


Figure 6: **Performance of the models on the SimpleQA benchmark with Brier Score (left) and Accuracy (right).** Our forecasting training makes models far more calibrated on factuality predictions as a downstream effect, also slightly increasing accuracy.

do to the filtered one. We then train `Qwen3-8B` on both datasets with identical hyperparameters and compute budgets.

Despite being 3× smaller, the filtered set reaches **higher accuracy** and **higher** *free-form Brier score* in **roughly half the iterations**, and exhibits markedly **more stable** learning curves. In practice, the curation stages (i) remove ambiguous or under-specified questions, (ii) keeps only those questions which are firmly resolved by the article. Together, these steps raise the reward signal-to-noise ratio yielding better models even with substantially fewer training examples.

Generalization beyond forecasting. We evaluate models on SimpleQA to probe factuality. (Fig. 6). RL on `NewsCast` improves Brier on short-form factual questions and modestly increases accuracy. This indicates that forecasting training can reduce hallucinations by making models more calibrated.

Summary. Across backbones (1.7B–8B) and evaluation settings (free-form news, binary markets), GRPO with free-form Brier reward delivers (i) large calibration gains that translate into substantially better Brier scores, (ii) sizable accuracy improvements on both binary prediction markets and our synthetic news-derives testset, and (iii) competitive performance with much larger open-weights models on free-form evaluation. Our ablations indicate that data *quality* and *format* (free-form with binary) are key drivers; training even on real market data which is just binary questions is not a substitute for the learning signal obtainable via free-form supervision.

5 Limitations, Future Work, Conclusion

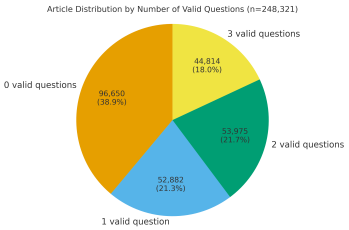
In this paper, we take first steps towards *training* models for forecasting by showing how to scale up RL training data using automated question generation from daily news articles. While the results are promising–we match 120B thinking models starting from an 8B thinking model, significantly improving both accuracy and calibration–there is a lot of scope for extending our work. Forecasting greatly benefits from relevant recent information which can be provided in-context to a model using retrieval and search tool use. Thus, our ongoing effort is focused on further utilizing our news corpus as a datasource for training interactive search agents for forecasting. Forecasting is a challenging task, requiring the integration of extensive knowledge and reasoning. As such, it can benefit from large models and more data, so we hope to scale up our pipeline more eventually. We will make all data, code and models available publicly to push open-source efforts towards using RL in LLMs to solve high-value real-world tasks.

References

Nikhil Chandak, Shashwat Goel, Ameya Prabhu, Moritz Hardt, and Jonas Geiping. Answer matching outperforms multiple choice for language model evaluation. *arXiv preprint arXiv:2507.02856*, 2025.

Mehul Damani, Isha Puri, Stewart Slocum, Idan Shenfeld, Leshem Choshen, Yoon Kim, and Jacob Andreas. Beyond binary rewards: Training lms to reason about their uncertainty. *arXiv preprint arXiv:2507.16806*, 2025.

Source	Articles (%Total)
Forbes	110,103 (44.3%)
The Hindustan Times	80,000 (32.2%)
The Irish Times	29,546 (11.9%)
Deutsche Welle (DW)	21,317 (8.6%)
Cable News Net (CNN)	7,355 (3.0%)
Total	**248,321 (100%)**



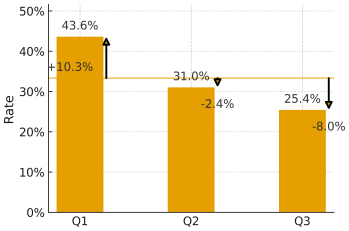


Table 2: **Data Distribution of** `NewsCast`**.** (Left) We show the breakdown of source documents by news outlet. (Right) We show the number of questions generated, and the proportion of the first, second and third generate question being picked as the final "best question".

Michael M Grynbaum and Ryan Mac. The times sues openai and microsoft over ai use of copyrighted work. *The New York Times*, 27(1), 2023.

Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. Approaching human-level forecasting with language models. *Advances in Neural Information Processing Systems*, 37:50426–50468, 2024.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models, 2020. URL https://arxiv.org/abs/2001.08361.

Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5):5, 2017.

Sebastian Nagel. Common crawl news dataset, 2016. URL https://data.commoncrawl.org/crawl-data/CC-NEWS/index.html.

Daniel Paleka, Shashwat Goel, Jonas Geiping, and Florian Tramèr. Pitfalls in evaluating language model forecasters. *arXiv preprint arXiv:2506.00723*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

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

Benjamin Turtel, Danny Franklin, and Philipp Schoenegger. Llms can teach themselves to better predict the future. *arXiv preprint arXiv:2502.05253*, 2025a.

Benjamin Turtel, Danny Franklin, Kris Skotheim, Luke Hewitt, and Philipp Schoenegger. Outcome-based reinforcement learning to predict the future. *arXiv preprint arXiv:2505.17989*, 2025b.

Wenda Xu, Guanglei Zhu, Xuandong Zhao, Liangming Pan, Lei Li, and William Yang Wang. Pride and prejudice: Llm amplifies self-bias in self-refinement, 2024. URL https://arxiv.org/abs/2402.11436.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

	Name(s)	Location	Country	Title	Team name	Color	Organization	Currency	Brand name	Month
Count	32,213	14,337	2,579	2,479	1,445	1,047	1,030	877	779	730
Share	44.8%	20.0%	3.6%	3.5%	2.0%	1.5%	1.4%	1.2%	1.1%	1.0%

Table 3: Top ten answer types of the questions in our curated dataset. These ten categories cover **80.1%** of our training dataset.

A Additional Details

A forecasting question from our synthetically generated benchmark

Question. Which Australian batsman will be Australia's highest run-scorer by the close of play on day three of the second Test in Grenada?

Background. The second Test between Australia and the West Indies is underway in Grenada under standard five-day Test conditions; day three is about to begin.

Resolution Criteria.

- **Source of truth:** The official day-three scorecard of the second Test as published by the International Cricket Council or ESPNcricinfo.
- **Resolution moment:** At the conclusion of play on day three (**July 6, 2025**).
- **Accepted answer format:** Full name of the Australian batsman exactly as listed on the scorecard.

Answer Type. string (name)

Ground-Truth Answer. Steve Smith

Source. The Guardian. Smith and Head build Australia's lead over West Indies after Green steadies ship

A.1 Experimental Details

Framework. We perform RL training using the `VeRL` package with GRPO algorithm [Shao et al., 2024] for optimization.

Policy/backbone. Unless noted, the trainable policy is `Qwen3-8B`. Prompts are truncated to 2,048 tokens and responses are capped at 8,192 tokens.

Sampling. We generate with a vLLM-based sampler (chunked prefill enabled). Training uses temperature 1.0 with $K=8$ samples per prompt.

Optimization. We use AdamW [Loshchilov et al., 2017] with learning rate 5×10^{-6}, cosine decay, 1% warmup, and a minimum LR ratio of 0.1. FSDP parameter *and* optimizer offloading are enabled; gradient checkpointing, padding removal, and dynamic batch sizing are used. Global train batch size is 256 (PPO mini-batch 64). Training runs are performed a node of 8 H100 GPUs for 5 epochs.

Advantages and losses. GRPO with group-centered advantages (no standard-deviation normalization). PPO clipping uses $\epsilon_{\text{low}}=0.20$, $\epsilon_{\text{high}}=0.28$, and clip-$c$=10.0. We apply a low-variance KL penalty with coefficient $\beta=0.005$.

Rewards. We use the `Qwen3-4B` as the judge for assessing answer correctness. We instruct it to enforce strict, reference-guided matching with tolerance for case and common aliases, and prompt it in `non-thinking` mode.

Question	Background	Resolution (trigger & deadline)	Answer Type	Answer	Source
Host country of COP30 (Nov 2025)?	UNFCCC COP venue rotates among regions.	Host confirmed by UNFCCC/organizers; no later than COP30 start (Nov 2025).	string (country)	Brazil	DW: link
Release month of Marvel's *Fantastic Four* (2025)?	Reboot announced with lead cast; 2025 release slated.	Month confirmed by Marvel/Disney; by Dec 2025.	string (month)	July	Forbes: link
First state to require Ten Commandments in public classrooms (by 2025)?	Several U.S. states advance religion-in-school measures.	First state enacts requirement; by Dec 31, 2025.	string (state name)	Louisiana	Forbes: link
African host of G20 Summit (Nov 2025)?	G20 presidency rotates; South Africa presiding from Dec 2024.	G20/host government confirms location; by Nov 2025.	string (country)	South Africa	DW: link
Recipient of Lesotho–Botswana Transfer Scheme (by 2025)?	Regional pipeline to pump water from Lesotho via SA.	ORASECOM or governments confirm recipient; by 2025.	string (country name)	Botswana	DW: link

Table 4: Five succinct forecasting questions spanning climate, entertainment, law, geopolitics, and infrastructure; selected for brevity and diverse sources (DW, Forbes). Each row lists the question (summarized here for conciseness), short background, resolution trigger with deadline, answer type, ground-truth answer, and citation.

B Prompt Templates for Question Creation Pipeline

Stage 1 — Question Generation (Requires: self.num_questions_per_article > 1)

```
**Task:** Based on the provided news article, generate
    {self.num_questions_per_article} high-quality, DIVERSE forecasting
    questions which have a short answer (1 - 3 words), using the XML format
    specified below.
Each forecasting question should be posed in a way to predict future events.
    Here, the predictor will have a knowledge cutoff before the article is
    published and no access to the article, so a forecasting question has to
    be posed about information explicitly stated in the article. The
    question should be stated in a forward-looking manner (towards the
    future).
The correct answer should be a specific, short text response. The answer
    should be a WELL DEFINED, SPECIFIC term which the answerer can come up
    with on its own, without access to the news article.

**Example Format**:
<q1>
<question_id>0</question_id>
<question_title>Who will win the Nobel Prize in Literature in
    2016?</question_title>
<background>Question Start Date: 10th January 2016. The Nobel Prize in
    Literature is awarded annually by the Swedish Academy to authors for
    their outstanding contributions to literature.</background>
```

```
<resolution_criteria>
<ul>
    <li>
      <b>Source of Truth</b>: The question will resolve when the Swedish
    Academy publicly announces the official 2016 Nobel Prize in Literature
    laureate(s)typically via a press release on NobelPrize.org (expected on
    or about October 13, 2016).
    </li>
    <li>
      <b>Resolution Date</b>: The resolution occurs on the calendar date
    when the 2016 laureate(s) are formally named
      (typically mid-October 2016).
    </li>
    <li>
      <b>Accepted Answer Format</b>: The full name of the laureate exactly
    as given in the announcement should be provided. If more than one person
    shares the prize, all names must be listed in the same order as the
    official communiqu.
    </li>
</ul>
</resolution_criteria>
<answer>Bob Dylan</answer>
<answer_type>String (Name)</answer_type>
</q1>

The question should follow the structured guidelines below.

### **Guidelines for Creating Short Answer Forecasting Questions**

**Title Question Guidelines**
- **Quality**: The question should be of HIGH QUALITY and hard to answer
    without access to the article. It should not be about any minute details
    in the article. THE QUESTION SHOULD BE SUCH THAT ITS ANSWER REVEALS A
    KEY PIECE OF INFORMATION, FROM THE ARTICLE, WHICH HAS MAXIMAL IMPACT.
- **Specific and Answerable**: The question to be created SHOULD BE
    FREE-FORM and have a unique, specific answer (a single word, or short
    phrase) without access to the article. The answer to the question should
    be definite, well-defined and NOT NUMERIC. IT SHOULD ALSO NOT BE
    UNCERTAIN like "above XYZ" OR A RANGE LIKE "between XYZ and ABC". Avoid
    creating binary questions (yes/no, either/or) or questions with a list
    of specific options (multiple choice).
- **Answerable based on article**: Each question must have a CLEAR AND
    DEFINITE answer based on information stated in the article. Given the
    question, the content of the article should be able to resolve the
    answer to the question INDISPUTABLY WITHOUT ANY AMBIGUITY OR
    UNCERTAINTY. THE ARTICLE SHOULD NOT STATE THAT THE ANSWER IS TENTATIVE
    OR AN ESTIMATE OR LIKELY. The answer SHOULD HAVE HAPPENED BY NOW.
- **Temporal Information**: The question should not be about recall of
    (past) facts or events known before the article publish date. Include
    any temporal information necessary to answer the question (like by which
    month, year, etc.) in the question. The question should always be posed
    in a forward-looking manner.
- **Direct and Precise**: Titles must be straightforward and unambiguous,
    avoiding vague terms. Use future tense when appropriate.
- **Resolution Criteria**: ALWAYS INCLUDE A BRIEF RESOLUTION CRITERIA in the
    question title. This is often the date by which the question will be
    resolved. For example, resolution dates such as "by {{month_name}},
    {{year}}?" or "in {{month_name}}, {{year}}?". THE RESOLUTION DATE SHOULD
    BE BASED ON (AND FAITHFUL TO) THE CONTENT OR PUBLICATION DATE OF THE
    ARTICLE.
- **No references to article or future information**: DO NOT refer to the
    specific article, such as by saying "in the article". The forecaster
```

```
    does not have access to the article, its metadata or any information
    beyond the article publish date.
- **Question Types**: Focus on "Who", "What", "When", "Where" questions that
    have concrete answers.
- **Understandability**: The question title should have ALL the information
    to be understandable by a 10 year old. It should be independently
    understandable without the article.
- **Tense**. ALWAYS POSE THE QUESTION IN A FORWARD-LOOKING MANNER. THE
    QUESTION SHOULD BE IN FUTURE TENSE. Try to use phrases like "What will",
    "Who will", "When will", "Where will", "How much/many will" etc. It
    should appear as a forecasting question and not past prediction.

**Answer Guidelines**
- **Faithfulness to Article**: The answer should be based on information
    explicitly stated in the article, and not implications or your own
    knowledge. IT SHOULD BE STATED VERBATIM IN THE ARTICLE.
- **Non-Numeric**: The answer should not be a number or a percentage. It can
    be a word, phrase, date, location, etc BUT NOT MORE THAN 3 WORDS.
- **Definite** - Given the question and the article, the answer should be
    CLEAR, CONCRETE, CERTAIN AND DERIVABLE from the article. It should be
    short, WELL-DEFINED TERM and not uncertain or vague. It SHOULD NOT BE A
    RANGE like "between XYZ and ABC" or "above XYZ" or "below PQR".
- **Resolved** - The answer MUST be something that has already happened or
    is happening now. It should be resolved given today's date and not be
    something that will happen in the future.
- **Specificity**: The answer should be specific enough to be unambiguous.
    Avoid overly general answers.
- **Conciseness**: Keep answers short - typically 1-3 words, occasionally a
    short phrase if necessary.
- **Exactness**: For names, use the exact names mentioned (full name, if
    possible).
- **Uniqueness**: The answer should be unique and THE ONLY CORRECT ANSWER to
    the question.
- **No Ambiguity**: The answer should be indisputable and not be open to
    multiple interpretations. IT SHOULD BE PRECISE AND NOT A RANGE OR
    UNCERTAIN ESTIMATE.

**Background Guidelines**
- **Mention Question Opening Date**: ALWAYS INCLUDE THE START DATE OF THE
    QUESTION IN THE BACKGROUND. IT SHOULD BE AT LEAST A FEW DAYS (OR WEEKS
    IF THE QUESTION IS ABOUT A LONG-TERM EVENT) BEFORE THE ARTICLE'S PUBLISH
    DATE AND ALSO BEFORE THE RESOLUTION DATE OF THE QUESTION. CONSEQUENTLY,
    THE BACKGROUND SHOULD NOT CONTAIN ANY INFORMATION WHICH HAS HAPPENED
    AFTER THE START DATE OF THE QUESTION.
- **Necessary Context**: The answerer does not have access to the article,
    so include MINIMAL CONTEXT required to understand the question keeping
    in mind the question opening date. Do not give (extra) details of the
    event from the article as background. If required, EITHER pose the event
    as a hypothetical scenario as if it were to happen in the future OR
    describe it as happening (unfolding) in real time. Describe any
    unfamiliar terms or concepts in the question title.
- **SHOULD NOT HELP ANSWER**: WHILE PROVIDING THE CONTEXT, DO NOT REFER OR
    MENTION OR LEAK THE ACTUAL ANSWER. The background must not help answer
    the forecasting question. DO NOT INCLUDE ANY INFORMATION from the
    article or elsewhere that either directly or indirectly (even partially)
    reveals the answer.
- **No Additional Knowledge**: Do not add any knowledge beyond what is
    required to understand the question. Only include information necessary
    to understand the question and its context.
- **Tense**. ALWAYS POSE THE BACKGROUND INFORMATION IN CURRENT TENSE. Only
    provide minimal information which is known until the question opening
    date.
```

```
**Resolution Criteria**
- **Necessary Criteria**: State the EXACT conditions by which the outcome
    will be judged. Include the criteria which determines how the question
    will be resolved. state the conditions by which the outcome will be
    judged.
- **Date and Source of Resolution**: Always state the date and the source by
    which the question will be resolved. For example, resolution dates such
    as "by {{month_name}}, {{year}}?" or "in {{month_name}}, {{year}}?", and
    potential source(s) of resolution such as "based on {{news source}}",
    "reports from {{official name}}", etc. THE RESOLUTION DATE SHOULD BE
    CHOSEN THOUGHTFULLY AS THE ANSWER'S VALIDITY AND SOUNDNESS DEPENDS ON
    IT. THE RESOLUTION DATE SHOULD BE SUCH THAT THE ANSWER CAN BE RESOLVED
    DEFINITELY AND INDISPUTABLY FROM THE CONTENT OR PUBLICATION DATE OF THE
    ARTICLE. IT SHOULD MENTION BY WHEN IS THE OUTCOME OF THE QUESTION
    EXPECTED TO HAPPEN. HOWEVER, IT SHOULD NOT LEAK OR MENTION ANYTHING
    ABOUT THE ARTICLE.
- **Details**: Be as detailed as possible in creating the resolution
    criteria for resolving the question as cleanly as possible. There should
    be no ambiguity in the resolution criteria.
- **Expectation and Format of Answer**: Based on the actual answer, the
    resolution criteria should state how precise the expected answer should
    be and in what format it should be. For example, if the actual answer is
    a date, the resolution criteria should specify how detailed the expected
    date should be -- only year, or both month and year, or day, month, and
    year all together. DO NOT GIVE THE ACTUAL DATE (ANSWER). If the actual
    answer is a percentage, then the criteria should state the expected
    answer should be a percentage. DO NOT GIVE THE ACTUAL PERCENTAGE.  If
    the actual answer is in certain unit, then the criteria should specify
    that. THE RESOLUTION CRITERIA SHOULD MAKE IT EXACTLY CLEAR AND PRECISE
    WHAT IS EXPECTED FROM THE ANSWERER AND IN WHAT FORMAT AND HOW IT WILL BE
    CHECKED LATER. IF GIVING AN EXAMPLE, IT SHOULD BE VERY GENERIC AND AS
    FAR AWAY FROM THE ACTUAL ANSWER AS POSSIBLE.
- **SHOULD NOT HELP ANSWER**: The resolution criteria must not directly help
    answer the forecasting question. DO NOT INCLUDE ANY INFORMATION from the
    article or elsewhere that either directly or indirectly (even partially)
    reveals the answer. DO NOT REFER OR MENTION OR LEAK THE ACTUAL ANSWER
    HERE.

**Answer Type Guidelines**
- **Expected Format**: The answer type should be either "numeric (XYZ)" if
    the answer is a number (of any kind) or "string (XYZ)" in all other
    cases. In numeric cases, XYZ should be the exact type of number
    expected. For example, "numeric (integer)", "numeric (decimal)",
    "numeric (percentage)", "numeric (whole number)", etc. In string cases,
    XYZ should broadly be the category of string expected. For example,
    "string (name)", "string (date)", "string (location)", etc. If the
    category is not clear, use "string (any)". HOWEVER, ALWAYS TRY TO CREATE
    QUESTIONS WHERE THE ANSWER CATEGORY IS CLEAR AND PRECISE.

**Question Quality Criteria**
- **Forecastable**: The question should be something that could reasonably
    be predicted or forecasted before the article's publication.
- **Towards the future**: THE QUESTION SHOULD BE POSED IN A FORWARD-LOOKING
    MANNER.
- **Interesting**: The question should be about a meaningful event or
    outcome, not trivial details.
- **Impactful**: The question should be such that if its answer is
    forecasted ahead of time, it should have significant (downstream) impact
    (relevant to high number of people).
- **Difficulty**: While the question should be hard to answer without access
    to the article, it should also not be unreasonably difficult.
- **Verifiable**: The answer should be something that can be EXACTLY
    verified from the article itself.
```

```
- **Time-bound**: Include clear timeframes or deadlines when relevant.
- **Free-form**: If possible, avoid creating binary questions (yes/no,
    either/or) or questions with a list of specific options (multiple
    choice).

Generate {self.num_questions_per_article} high-quality, DIVERSE short answer
    forecasting questions based on the provided article. Use the XML format
    with question_id value "0", "1", "2", etc. DO NOT INCLUDE ANY ANALYSIS,
    RANKING, OR ADDITIONAL COMMENTARY.

Article:
{source_article}

**Required Output Format**:
<q1>
<question_id>0</question_id>
<question_title>[Question 1]</question_title>
<background>[Background 1]</background>
<resolution_criteria>[Resolution Criteria 1]</resolution_criteria>
<answer>[Answer 1]</answer>
<answer_type>[Answer Type 1]</answer_type>
</q1>
..
<q{self.num_questions_per_article}>
<question_id>{self.num_questions_per_article - 1}</question_id>
<question_title>[Question {self.num_questions_per_article}]</question_title>
<background>[Background {self.num_questions_per_article}]</background>
<resolution_criteria>[Resolution Criteria
    {self.num_questions_per_article}]</resolution_criteria>
<answer>[Answer {self.num_questions_per_article}]</answer>
<answer_type>[Answer Type {self.num_questions_per_article}]</answer_type>
</q{self.num_questions_per_article}>
```

Stage 2 — Individual Validation

```
**Task:** You will be provided with a news article and a question WHOSE
    ANSWER IS SUPPOSED TO BE BASED ON THE ARTICLE. Your job is to validate
    whether the answer to the question is valid by being faithful to the
    article (content, title, or description).

GO THROUGH EACH SEGMENT OF THE QUESTION ONE BY ONE (TITLE, BACKGROUND,
    RESOLUTION CRITERIA, ANSWER) TO UNDERSTAND THE WHOLE QUESTION. THEN
    CHECK EACH OF THE FOLLOWING CRITERIA:

1. **Tense and Details**: FIRST CHECK WHETHER THE QUESTION IS NOT UNDER
    SPECIFIED OR STATED IN PAST TENSE. IT IS FINE IF THE QUESTION IS STATED
    IN CURRENT OR FUTURE TENSE.
2. **Definite resolution of the answer by the article**: CHECK WHETHER THE
    ANSWER TO THE QUESTION IS SOUND, CLEAR AND PRESENT IN OR CAN BE DERIVED
    FROM THE ARTICLE. THE ARTICLE SHOULD RESOLVE THE ANSWER DEFINITELY AND
    IN AN INDISPUTABLE MANNER (WITHOUT ANY AMBIGUITY). THIS IS THE MOST
    IMPORTANT CRITERIA.
3. **Well-defined Answer**: The answer to the question should be short (NOT
    MORE THAN 3 WORDS). IT SHOULD NOT BE A PHRASE AND SHOULD BE SOMETHING
    WHICH IS CONCRETE, SPECIFIC AND WELL-DEFINED.
4. **Non-Numeric**: THE *ANSWER TYPE* SHOULD NOT BE NUMERIC LIKE A
    PERCENTAGE, INTEGER, DECIMAL, OR A RANGE.
5. **Single Correct Answer**: ANALYZE WHETHER THE QUESTION CAN HAVE MULTIPLE
    OUTCOMES OR RIGHT ANSWERS. IF SO, THE QUESTION FAILS THIS CRITERIA.
    OTHERWISE, ENSURE THAT THE PROVIDED ANSWER IS THE SOLE CORRECT ANSWER TO
    THE QUESTION. IT SHOULD NOT BE THE CASE THAT THE QUESTION CAN HAVE
    MULTIPLE (DISTINCT) CORRECT ANSWERS.
```

```
If ALL the above criteria pass (question is stated as required, answer to
    the whole question is valid, well-defined, and it is the only correct
    answer to the question), ONLY THENreturn <answer>1</answer>. Otherwise,
    return <answer>0</answer>. ALWAYS END YOUR RESPONSE IN <answer>
    </answer> tags.

**Article:**
{source_article}

**Question:**
{questions_text}

**Output Format:**
<answer>0/1</answer>
```

Stage 3 — Choose Best

```
**Task:** You will be provided with a list of questions (possibly with size
    1). Your job is to choose the best question from the list based on the
    following criteria or end your response with "NO GOOD QUESTION" if none
    of the questions meet the criteria.

**Instructions:**
GO THROUGH EACH QUESTION ONE BY ONE AND ANALYZE IT FOR THE FOLLOWING:
1. **Valid for forecasting**: Check if the WHOLE QUESTION is stated in a
    forward-looking manner. FROM THE PERSPECTIVE OF THE START DATE TO THE
    RESOLUTION DATE MENTIONED IN THE QUESTION, CHECK IF IT IS A VALID
    FORECASTING QUESTION. IF THE TIME HORIZON (START DATE TO RESOLUTION
    DATE) IN THE QUESTION IS AT LEAST A SINGLE DAY, THEN THE QUESTION SHOULD
    BE CONSIDERED VALID FOR FORECASTING. Go through each segment of the
    question (question title, background, resolution criteria) and check if
    each of them is valid and forward-looking.
2. **Tense**: The question SHOULD NOT BE STATED IN PAST TENSE. If the
    question covers an event, it should not imply as if the outcome of the
    event has already happened or occurred.
3. **Single Correct Answer**: ANALYZE WHETHER THE QUESTION CAN HAVE MULTIPLE
    OUTCOMES OR RIGHT ANSWERS. IF SO, THE QUESTION FAILS THIS CRITERIA.
    OTHERWISE, ENSURE THAT THE PROVIDED ANSWER IS THE SOLE CORRECT ANSWER TO
    THE QUESTION. IT SHOULD NOT BE THE CASE THAT THE QUESTION CAN HAVE
    MULTIPLE (DISTINCT) CORRECT ANSWERS.
4. **Impact**: How many people will the outcome of the question be relevant
    or interesting to? Consider on the basis of significant downstream
    impact or enabling meaningful action.
5. **Not Binary/Multiple Choice**: Question SHOULD NOT BE BINARY (yes/no,
    either ABC or XYZ, etc.) OR MULTIPLE CHOICE (SELECT FROM A LIST OF
    OPTIONS). It should be free-form (string -- name, date, place, etc.) or
    numerical (number, percentage, etc.).
6. **Understandable**: THe question as a whole (title, background,
    resolution criteria) should have sufficient details to understand the
    premise of the question. Every detail should be crystal clear and the
    question should not be under or over specified.
7. **Definite Answer**: EXTRACT THE ACTUAL ANSWER TO THE QUESTION PROVIDED
    IN ITS <answer> </answer> TAG. The extracted answer should be short,
    definite, well-defined and not uncertain or vague. It SHOULD NOT BE A
    PHRASE OR A RANGE like "between XYZ and ABC" or "above XYZ" or "below
    PQR".



ANALYZE EACH QUESTION BASED ON THE ABOVE CRITERIA ONE BY ONE AND CHOOSE THE
    ONE WHICH PASSES ALL THE ABOVE CRITERIA. IF MULTIPLE QUESTIONS SATISFY
```

```
    THE CRITERIA, CHOOSE THE ONE WHICH WILL HAVE THE HIGHEST IMPACT (AFFECTS
    OR IS RELEVANT TO THE MOST NUMBER OF PEOPLE). IF NO QUESTION MEETS THE
    CRITERIA, RETURN "NO GOOD QUESTION FOUND". OTHERWISE, RETURN THE BEST
    QUESTION IN THE SAME FORMAT AS THE INPUT.

**Generated Questions:**
{questions_text}

**Output Format:**
<q1>
<question_id>0</question_id>
<question_title>[ORIGINAL Title of the best question]</question_title>
<background>[ORIGINAL Background of the best question]</background>
<resolution_criteria>
<ul>
    <li> <b>Source of Truth</b>: [ORIGINAL Source of Truth of the best
    question] </li>
    <li> <b>Resolution Date</b>: [ORIGINAL Date of the best question] </li>
    <li> <b>Accepted Answer Format</b>: [ORIGINAL Accepted Answer Format of
    the best question] </li>
</ul>
</resolution_criteria>
<answer>[ORIGINAL Answer of the best question]</answer>
<answer_type>[ORIGINAL Answer Type of the best question]</answer_type>
</q1>
```

Stage 4 — Leakage Removal

```
**Task:** You will be provided with a forecasting question. Your job is to
    ANALYZE whether the question's answer has obviously leaked in the
    content of the question. The question will have multiple segments --
    question title, background, resolution criteria. EXCEPT THE QUESTION
    TITLE, GO THROUGH EACH SEGMENT STEP BY STEP and check if any part
    DIRECTLY leaks the actual answer. If leakage is found, ONLY THEN
    rephrase the problematic parts appropriately to remove the answer while
    maintaining the question's integrity and focus. DO NOT CHANGE ANY PART
    OF THE QUESTION UNNECESSARILY.

USE THE SAME XML FORMAT IN YOUR RESPONSE AS IS IN THE INPUT.

**Generated Question:**
{questions_text}

**Instructions:**
1. **Keep the title unchanged**: DO NOT MAKE ANY CHANGE TO THE QUESTION
    TITLE.
2. **Keep the start date in the background unchanged**: DO NOT MAKE ANY
    CHANGE TO THE QUESTION'S START DATE IN THE BACKGROUND.
3. **Identify the answer**: First, extract the actual answer from the XML
    tags for the current question being processed.
4. **Identify Leakage**: Keeping the extracted answer in mind, check if the
    background, or resolution criteria (each of them -- source of truth,
    resolution date, accepted answer format) contain information that
    reveals the answer.
5. **Types of leakage which can be ignored**: The following types of leakage
    are fine and don't need to be rephrased:
   - If the outcome (actual answer) of the question is binary (yes/no,
    either ABC or XYZ, etc.), then NO NEED TO CHANGE ANYTHING ANYWHERE.
   - If the resolution criteria is based on a list of specific options, then
    NO NEED TO CHANGE ANYTHING IN ANY SEGMENT (BACKGROUND, RESOLUTION
    CRITERIA, etc.). For example, if the accepted answer format states
```

```
    "answer must be either .." OR "answer must be one of the following
    terms..", then NO NEED TO CHANGE ANYTHING ANYWHERE.
6. **Types of Leakage to Check:** ONLY CONSIDER THE FOLLOWING KIND OF
    LEAKAGE:
   - DIRECT MENTIONS of the answer (either in word or number form) or part
    of the answer in the question/background/resolution
   - References to specific outcomes that ARE CLOSE TO (OR REVEAL)THE ACTUAL
    ANSWER
7. **Rephrase Strategy**: If leakage is found, rephrase the problematic part
    while:
   - Keeping the question's core intent
   - Maintaining forecasting nature
   - Preserving necessary context
   - Making the answer UNOBVIOUS by replacing with a FAKE ANSWER (FAKE NAME,
    DATE, NUMBER, PERCENTAGE, etc.) WHICH IS GENERIC AND NOT CLOSE TO THE
    ACTUAL ANSWER.
   - The rephrased part should not contain any information that is part of
    the actual answer. Neither should it indirectly hint or reveal the
    answer.
8. **Check Accepted Answer Format**: IF THERE IS ANY EXAMPLE MENTIONED IN
    ACCEPTED ANSWER FORMAT ("e.g..."), MAKE SURE THE EXAMPLE IS GENERIC AND
    AS FAR AWAY FROM THE ACTUAL ANSWER AS POSSIBLE. DO NOT INCLUDE AN
    EXAMPLE IF NOT MENTIONED ALREADY.
9. **Do not change the answer**: Do not change the actual answer to the
    question.
10. **Do not change the answer_type**: DO NOT MAKE ANY CHANGE TO the
    answer_type.
11. **Each segment should be checked independently**: Go through each
    segment of the whole question one by one. Everything from the title of
    the question to the background information to the resolution criteria
    should be checked independently with reference to the answer of the
    question. In the resolution criteria, go through each <li> step by step.
    Do not change the other segments when rephrasing a problematic segment.
12. **Do not change anything unless leakage is found**: DO NOT UNNECESSARILY
    CHANGE ANY PART OF THE QUESTION UNLESS LEAKAGE IS FOUND.

IT IS ALSO POSSIBLE THAT MULTIPLE PARTS OF THE QUESTION HAVE LEAKAGE. YOU
    SHOULD CHECK EACH OF THEM INDEPENDENTLY AND ONLY IF LEAKAGE IS FOUND,
    REPHRASE THE PROBLEMATIC PARTS. DO NOT OVER-ANALYZE.

During your analysis, you should:
- Go through EACH SEGMENT OF THE QUESTION STEP BY STEP INDEPENDENTLY. First
    <background> and then inside <resolution_criteria>. Under the resolution
    criteria, go through the source of truth, resolution date, accepted
    answer format (each of them is a <li> tag) one by one. For each such
    segment, do the following:
    - Compare the content in the current segment with the actual answer. If
    ANY PART OF THE ANSWER is mentioned in the current segment, then
    consider that as a leakage UNLESS THE ACCEPTED ANSWER FORMAT IS BINARY
    (yes/no, either ABC or XYZ, etc.) OR A LIST OF SPECIFIC OPTIONS.
    - IF THE CURRENT SEGMENT IS BACKGROUND, DO NOT CHANGE THE QUESTION START
    DATE.
    - If the current segment is accepted answer format and there is a
    SPECIFIC EXAMPLE MENTIONED in it ("e.g. XYZ") which is close to the
    actual answer, then consider that as a leakage.
    - If leakage is found in the current segment, mention "Leakage found --
    {{reason for leakage}}". Form the segment with the problematic parts
    rephrased and mention it as "Replacement -- {{rephrased_text}}." THE
    REPHRASED TEXT SHOULD BE AS FAR AWAY FROM THE ACTUAL ANSWER AS POSSIBLE.
    It should now be present in the final output (instead of the original
    text).
    - Otherwise, mention "No leakage found". In your final output after you
    finish the analysis, return this segment UNCHANGED.
```

```
    - These outputs should be in the same format as the original input.
- Return the actual answer unchanged in the <answer> tag in your final
    output.
- Skip any other segments (question title, answer_type, etc.) in your
    analysis and output them unchanged (verbatim) in the final output.

Output your analysis step by step, and then end your response with the
    CORRECTED question in THE SAME XML FORMAT AS THE ORIGINAL.

**Output Format**:
{{ analysis }}

<q1>
<question_id>0</question_id>
<question_title>[UNCHANGED Question Title]</question_title>
<background>[Corrected Background]</background>
<resolution_criteria>
<ul>
    <li> [UNCHANGED Question Start Date] [Corrected Source of Truth] </li>
    <li> [UNCHANGED Resolution Date] </li>
    <li> [Corrected Accepted Answer Format] </li>
</ul>
</resolution_criteria>
<answer>[UNCHANGED Answer]</answer>
<answer_type>[UNCHANGED Answer Type]</answer_type>
</q1>
```