Can we teach LLMs to predict the future?

Andy Kaboski
akaboski@uchicago.edu
University of Chicago
Chicago, IL, USA

Gabriel López-Asiaín
gla@uchicago.edu
University of Chicago
Chicago, IL, USA

Karsten Kropp
kroppk@uchicago.edu
University of Chicago
Chicago, IL, USA

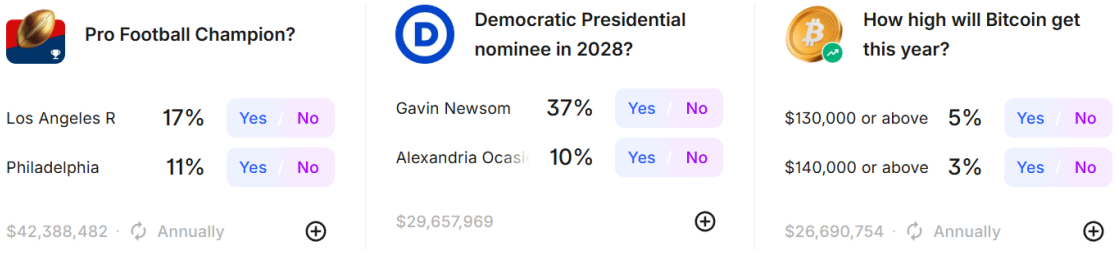

Figure 1: Examples of real-money forecasting markets on Kalshi.

Abstract

The application of Large Language Models (LLMs) to probabilistic forecasting offers the potential to automate expert-level decision-making; however, current benchmarks often suffer from data leakage and rapid obsolescence. This study explores the feasibility of teaching LLMs to predict future events through a novel retrospective evaluation framework. We introduce an automated pipeline to generate high-quality, verifiable binary forecasting questions grounded in historical events, strictly enforcing temporal causality to prevent information leakage. To measure performance on this synthetic dataset, we employ an autonomous, time-sensitive news retrieval system and fine-tune Llama-3-8B models on real prediction-market data to weigh evidence and calibrate probability estimates. We evaluate performance across three conditions: zero-shot baselines, domain-specific fine-tuning, and general-purpose fine-tuning. Our results demonstrate that fine-tuning significantly reduces Brier scores compared to zero-shot approaches, indicating improved calibration. Notably, we find no significant performance difference between domain-specific and general-purpose models, suggesting that fine-tuning enhances generalized reasoning capabilities rather than relying on domain-specific memorization. This work establishes a robust methodology for training and benchmarking forecasting agents without waiting for real-world event resolution.

Keywords

Forecasting, Large Language Models, Prediction Markets, Artificial Intelligence, Decision Making

1 Introduction

1.1 Background

Forecasting has been a central discipline of science for decades, applied to fields ranging from weather and stock markets to geopolitics and sports. It allows us to transform uncertainty into actionable probabilities that are essential for decision-making. Reliable forecasting systems enable risk mitigation and resource allocation, allowing society to navigate the volatility of the future with stability.

Traditionally, forecasting has been divided into two approaches: statistical forecasting, which relies on tools from statistical modeling that are very useful in predicting the behavior of certain systems like weather patterns, energy demand, or inventory cycles; and judgmental forecasting, where experts combine their intuition with information to make predictions about complex, unprecedented events.

With the rise of Large Language Models (LLMs) and the popularization of prediction markets like Kalshi or Polymarket, the opportunity to realize automatic forecasting for general-purpose topics has emerged. The idea is that an LLM possesses the knowledge and data of an expert but applies this capability across many different subjects simultaneously.

Additionally, it must be noted that forecasting serves as an excellent benchmarking tool for new LLMs. The problem with current benchmarks is that it is very difficult to prevent overfitting, making it challenging to develop a standard that remains valid over the years. The natural time barrier allows for a safe and contamination-proof evaluation environment; Since future events do not exist in any training dataset, models are forced to rely on genuine reasoning and analysis rather than memorization to find the answer.

1.2 Problem Statement

The goal is to develop a system that leverages LLMs to make better predictions than human experts. However, realizing this is not trivial. Since LLMs are trained for next-token prediction rather than probabilistic reasoning, models are often really imprecise and over-confident and fine-tuning is likely needed to improve accuracy. However, this presents a series of specific challenges.

Firstly, fine-tuning requires a large, balanced source of data. Platforms like Polymarket provide extensive data; nonetheless, this data covers topics unequally, leaving many general-purpose subjects underrepresented and potentially biasing the model.

Secondly, testing the system on real events would be highly impractical, as one would have to wait months or years for a resolution. To develop a system useful for long-term predictions, we need a method to evaluate performance immediately without waiting for the natural time barrier.

Finally, it is yet to be known if fine-tuning improves the model's general forecasting intuition or just allows it to understand a specific domain better. Distinguishing between these two types of learning is essential to determining wether the model is truly learning to reason or merely memorizing domain facts.

1.3 Objectives and Contributions

The primary objective of this study is to develop an automated forecasting system that leverages Large Language Models (LLMs) to achieve or exceed human-expert performance in predicting future events. This research seeks to demonstrate that fine-tuning models on structured forecasting data significantly improves their calibration and accuracy compared to zero-shot baselines. Furthermore, it examines whether training on domain-specific data improves forecasting accuracy over training on broader data.

We contribute to the field by (1) introducing a novel pipeline that leverages LLMs and web search to synthesize high-quality, verifiable forecasting datasets; (2) establishing a robust fine-tuning framework that aligns language models with probabilistic forecasting tasks; and (3) implementing an autonomous retrieval architecture that aggregates and synthesizes relevant news context to ground predictions.

2 Literature Review

Recent work has explored whether language models can predict real-world events. Zou et al. [7] introduced the Autocast benchmark, establishing a standardized framework for evaluating neural networks on world event prediction and demonstrating that models struggle with complex geopolitical questions. Jin et al. [3] developed ForecastQA, a question-answering challenge for temporal event forecasting, finding that existing models perform poorly when required to reason about future outcomes from textual evidence. Building on these benchmarks, Yan et al. [5] proposed Autocast++, which improved prediction accuracy through zero-shot ranking-based context retrieval without requiring task-specific fine-tuning.

Direct comparisons between LLMs and human forecasters have yielded mixed results. Schoenegger and Park [4] evaluated LLMs in a real-world forecasting tournament and found that models underperformed relative to skilled human forecasters. Abolghasemi et al. [1] reached similar conclusions, noting that while LLMs can process vast information quickly, they lack the calibrated uncertainty that expert humans develop through experience. More recently, Yang et al. [6] introduced Prophet Arena to systematically benchmark predictive intelligence, revealing significant variance in forecasting ability across different model architectures.

This research builds directly on Halawi et al. [2], who demonstrated that retrieval-augmented, fine-tuned LLMs can approach human-level forecasting on competitive prediction platforms. Their approach of combining structured fine-tuning with news retrieval forms the methodological foundation of our system.

3 Methodology

3.1 Research Design

This study uses an experimental approach to evaluate the predictive capabilities of Large Language Models (LLMs) on future events. To measure performance immediately without waiting for real-world resolution, we employ a retrospective strategy that tests models on resolved historical events.

To effectively prevent data leakage, every question q_i is associated with two timestamps: $t^{(i)}_{\text{ask}}$, the simulated date the question is posed, and $t^{(i)}_{\text{resolve}}$, the date the event outcome is determined. Let T_{cutoff} denote the knowledge cutoff date of the base LLM.

Our experimental process follows four main steps:

(1) **Data Generation and Splitting:** We construct our experimental datasets by curating historical questions from prediction market platforms to form the training set, $\mathcal{D}_{\text{train}}$, while utilizing our synthetic pipeline to generate the test set, $\mathcal{D}_{\text{test}}$. To ensure the validity of our retrospective evaluation and prevent data leakage, we strictly enforce the following two conditions:
   - **Forward-Looking Test Set:** For all $q_j \in \mathcal{D}_{\text{test}}$, the question date must be posterior to the model's training cutoff: $t^{(j)}_{\text{ask}} > T_{\text{cutoff}}$ ensuring the model does not already know the answer.
   - **No Leakage from Future:** To prevent the model from learning future outcomes during training, all training events must resolve before any test question is asked. That is, for every training sample $q_i \in \mathcal{D}_{\text{train}}$ and every test sample $q_j \in \mathcal{D}_{\text{test}}$, we require:

$$\max_{i \in \mathcal{D}_{\text{train}}} (t^{(i)}_{\text{resolve}}) < \min_{j \in \mathcal{D}_{\text{test}}} (t^{(j)}_{\text{ask}}) \tag{1}$$

(2) **Retrieval:** For each question q_i, we retrieve news articles strictly from the time window $t < t^{(i)}_{\text{ask}}$ to provide the model with historically accurate context.

(3) **Fine-Tuning:** We fine-tune the model on $\mathcal{D}_{\text{train}}$ to optimize its ability to weigh evidence and calibrate probability estimates.

(4) **Evaluation:** We assess the fine-tuned model on $\mathcal{D}_{\text{test}}$ against a zero-shot baseline. To investigate learning mechanisms, we compare two strategies: training on a broad, general-purpose dataset versus training on a single, narrow domain. This comparison isolates whether the model acquires a generalized forecasting intuition or merely memorizes domain-specific patterns.

3.2 Market data collection

To build our training dataset $\mathcal{D}_{train}$, we process the YuehHanChen dataset from HuggingFace which is an aggregation of forecasting questions across sites like Metaculus, Good Judgment Open, INFER, Polymarket and Manifold. This raw dataset contains nearly 50,000 questions and over 7 million user forecasts with questions spanning from 2015 to 2024. This dataset provided a robust amount of information of human-generated predictions and outcomes.

The dataset comprises 50,000 global questions (approx. 33k binary, 10k multiple-choice, and 7k numerical), each containing essential metadata including background context, resolution criteria,

and timestamps (publication, closing, resolution). This temporal structure is critical for our retrospective evaluation, enabling the accurate simulation of historical information states.

To prepare this data for the training process, we implemented a preprocessing pipeline which consisted of four steps. First, we filtered questions to ensure that we only had binary structure, as these provided clear values of 0 or 1 for ground truth labels for both training and evaluation. We also chose to exclude questions that were ill-defined, overly personal or focused on very niche topics in order to maintain data quality. Second, we used a LLM-based classifier that partitioned each of the questions into the eight predefined domains which were: Science and Tech, Healthcare and Biology, Economics and Business, Politics and Governance, Education and Research, Arts and Recreation, Security and Defenses, and Sports, dropping the question if it did not fit in one of these categories. This approach allowed us to train the separate fine-tuned models on specific domains.

Third, we extracted human forecast data for each of the questions to establish performance benchmarks. Because there were multiple collaborative predictions on the website, we aggregated these predictions to filter our synthetic data. Finally, we enforced a strict temporal segregation on the data to prevent data leakage. Because our base LLM's knowledge cutoff extends to early 2025, we made sure that all of the training questions satisfied $t_{\text{ask}}^{(j)} > T_{\text{cutoff}}$ to make sure that there was a temporal boundary between D_{train} and D_{test} making sure that the model's evaluation tests are geniune predictions and forecasting ability, not the model simply memorizing outcomes.

3.3 Retrieval System

To provide forecasters with relevant contextual information, we implement a four-stage LLM-powered news retrieval pipeline. Given a forecasting question and the simulated reference date (t_{ask}), the process operates as follows: (1) the system employs an LLM to generate optimized search keywords tailored to the question's domain (e.g., "Apple iPhone early release rumors 2024"); (2) a retrieval agent queries Google News using both the original question and generated keywords, strictly collecting articles published in the window prior to t_{ask}; (3) a rating agent independently evaluates each retrieved article on a 1–5 relevance scale, filtering out low-quality content; and (4) a synthesis agent combines all high-rated articles (score ≥ 3) into a coherent summary of verifiable facts without editorial speculation. Our approach closely follows the retrieval architecture of Halawi et al. [2], but extends it by generating multiple diverse search keywords per question and utilizing explicit numerical scoring for precise threshold-based filtering. Figure 2 shows the workflow of the system.

3.4 Fine-Tuning Framework

We opted to apply fine-tuning to the reasoning step in the pipeline, in which the model, after receiving a compiled summary of selected relevant news articles, then uses this summary to generate a reasoning thought process, followed by a prediction in the form of a probability between 0.0 and 1.0.

To gather training data, we used scratchpad prompts that ask a model to fill in various blanks, thereby guiding its reasoning. We gave questions and news summaries, inserted into these scratchpad prompts along with detailed instructions, to various base, non-fine-tuned LLMs, using multiple models (Llama 3.3-70B and gpt-oss-120b) and multiple scratchpad prompts for variety. Because these models were run using cloud servers, we were far less constrained by the need to conserve storage and compute resources. Thus, our goal was merely to generate as many possible examples of "successful" reasoning which led to a model making a stronger and more accurate prediction than the human-based market. The use of multiple models and multiple scratchpad prompts allowed multiple reasoning examples to be generated from a single question, potentially allowing multiple examples to be worked into the training dataset.

The vast majority of the training data was ultimately removed from the dataset due to an inability to outperform the human benchmark. However, a total of 4,356 predictions, in some cases multiple from the same question, managed to outperform humans, and were added to the training dataset. This data was then split by domain, creating eight additional smaller training datasets, each a subset of this larger training dataset. These eight datasets are designed to work in tandem - a new question will be categorized into one of the eight domains using an LLM, and that domain's respective model will then be used for reasoning.

3.5 Synthetic data Generation

To construct our dataset $\mathcal{D}_{\text{test}}$, we employ a three-module pipeline that leverages a retrospective generation approach. This system systematically creates verifiable binary forecasting questions by grounding them in historical events while simulating past uncertainty. For each generated question, the pipeline explicitly defines the simulated interaction time t_{ask} and the resolution time t_{resolve}, strictly enforcing the causality constraint $t_{\text{ask}} < t_{\text{resolve}} \leq t_{\text{now}}$.

The pipeline consists of three specialized agents with distinct roles (Figure 3):

The Architect (Taxonomy Expander) ensures the dataset $\mathcal{D}$ covers a diverse range of topics. It breaks broad user-defined domains (e.g., "Politics") into a granular taxonomy of subcategories, assigning a complexity score (1-10) to each. To prevent domain imbalance, the question generation budget is allocated proportionally to these complexity scores, ensuring that challenging or niche subjects are adequately represented in the training data.

The Historian (Question Generator) executes a two-stage process, maintaining a context window to guarantee diversity among questions in each subcategory. First, it drafts a binary question and defines the resolution date t_{resolve} from the perspective of t_{ask}, without knowledge of the actual outcome. Second, it performs a web search to determine the ground truth result at t_{resolve}, ensuring the label is historically accurate.

The Critic (Independent Validator) provides adversarial quality assurance. It conducts an independent web search to validate three criteria: (1) the question is unambiguous and strictly binary; (2) there is no temporal leakage (i.e., the answer was genuinely unknowable at t_{ask}); and (3) the ground truth label is correct. A question is only accepted into $\mathcal{D}$ if the Critic and Historian independently converge on the same result.

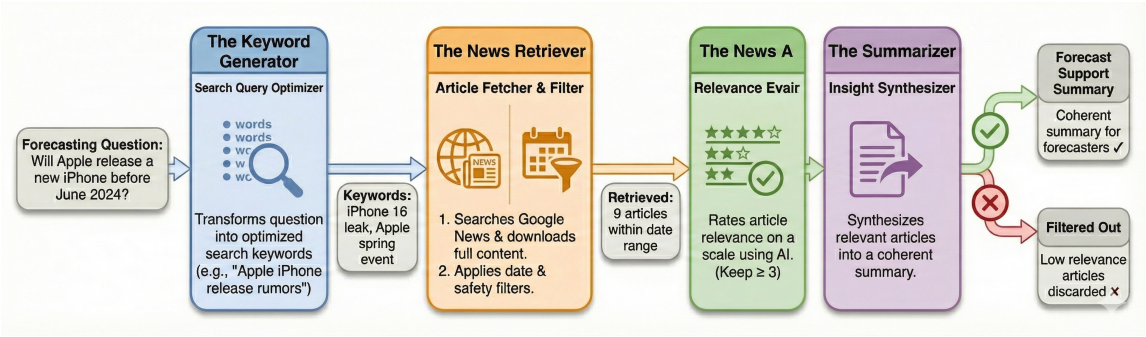

Figure 2: News summary generation pipeline

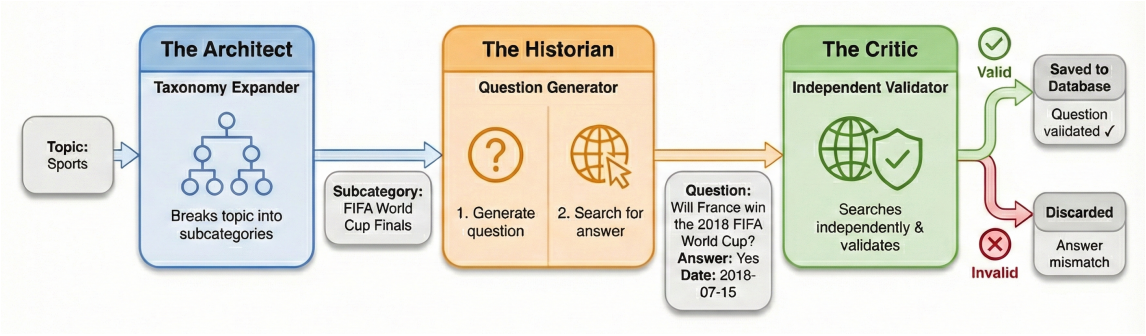

Figure 3: Data Generation Pipeline architecture

3.6 Evaluation Methodology

In order to evaluate the performance of the various trained models, we employed the Brier Score to be the primary metric for evaluation as so:

$$BS = \frac{1}{N}\sum_{i=1}^{N}(p_i - o_i)^2$$

Where p_i denoted the model's predicted probability for event i, and o_i is the binary outcome (0 or 1), and N is the total number of predictions. The Brier Score evaluation metric ranges from 0 (perfect calibration), to 1 (worst possible), where a lower score denotes greater predictive performance. We chose this metric to use for evaluation as Brier Score is the standard evaluation measure on the competitive forecasting platforms like Metaculus and Polymarket. Additionally as a strict scoring rule, the Brier Score incentivizes honest probabilistic forecasting.

The evaluation uses the held-out test set, D_{test}, for which we generated a set of approximately 2400 questions equally distributed among the categories *Arts & Recreation, Economics & Business, Education & Research, Healthcare & Biology, Politics & Governance, Science & Tech and Security & Defenses*. Sports

We compare the three model variants which are the zero-shot baseline model which uses the LLama3-8B-Instruct with no fine-tuning, a model which is fine-tuned on general purpose forecasting data from all domains and finally the eight domain specific models which are trained on subsets of the questions from said single domain. For the domain-specific model we utilize the classification to ensure that all questions trained and tested on are a part of the specific domain of focus.

The comparison between the three groups of models allows us to understand which model performs best for domain specific predictions as well as in a general forecasting accuracy scope. We then compare the model's brier score evaluations against the human forecaster benchmarks sourced from the dataset to further contextualize the performance of the models.

3.7 Implementation Details

Due to storage limitations on the cluster, the model used for fine-tuning is Llama3-8B-Instruct, with weights taken from HuggingFace. Hyperparameters were likewise chosen to be computationally efficient, with 3 epochs, a learning rate of 0.0002, and a batch size of 4. Each model was trained using a single NVIDIA A100 GPU and 100GB of memory. Each model took roughly 30 minutes to train, and roughly one minute per question to run.

All other LLMs were done using cloud servers, allowing hundreds of requests to be made at once, speeding up all other steps of the process. News evaluation and summarization used gpt-oss-120b, while zero-shot reasoning for acquiring fine-tuning training data was done using Llama 3-70B and gpt-oss-120b.

All code was done using Python, with libraries from HuggingFace (for fine-tuning) and OpenAI (for calling models on servers). News articles were searched for and retrieved using GNews API.

The full implementation is available on our GitHub repository (https://github.com/glopezas/forecast_kag).

4 Results

The results of running the pipeline, using different versions of the reasoning model, on the synthetically generated questions and answers are shown in Table 1.

5 Discussion

5.1 Interpretation of Results

The results show very little variation in performance between the domain-specific models, which were fine-tuned only on reasoning within their respective domains, and the general-purpose model, which was trained across all training data regardless of domain. There is similarly very little variation between domains.

It is not clear if this suggests that the added benefit of fine-tuning using domain-specific training data is counteracted by the negative effects of the decreased sample size (thousands to hundreds) of the training data itself, or if the lack of a significant difference is simply because neither the sample size nor the topics covered by the training data have any significant impact relative to the impact of having examples of any "correct" reasoning at all. Answering this question would require comparison of models trained on a specific sample size of data, regardless of breadth.

What is, however, noticeable, is that the fine-tuned models significantly outperform the base model using only the scratchpad prompt. The mere presence of some kind of fine-tuning on "correct" reasoning seems to decrease (and therefore improve) the Brier scores significantly. If probabilities were selected uniformly between 0 and 1 at random, the Brier scores would average 0.333. This is therefore a substantial increase, comparing the fine-tuned results' improvements over the base model's results' improvements against random chance.

5.2 Comparison with Existing Work

Halawi et al. [2] did not use domain-specific models, but found significant variance in Brier scores between domains. The disparity between these findings could come from many possible sources, but one strong possibility is our use of synthetic questions and answers. Although they tried to prevent models from answering questions and making predictions using their training data, this may not have been fully successful. The use of questions which can only be answered through retrieved news articles rather than the model's original training data may explain the comparative lack of variation in our results compared to theirs, as the lack of such a resource could even out results.

Brier scores on the fine-tuned models were higher as a whole for our results than for theirs. This is expected, as our own prediction pipeline contains a number of shortcuts designed to reduce compute time and the number of LLM calls, most notably retrieving fewer news articles and only running the reasoning model once, rather than taking the average across multiple attempts.

5.3 Limitations

Some limitations exist within the scope of the project including the temporal effects and data coverage, the retrieval systems inherent constraints, synthetic data artifacts and the model of use.

Temporal effects and data coverage: Our test set focused on questions that are resolved within the months that the question is asked also primarily within the model's knowledge cutoff period. This means that we do not validate performance on predictions with long horizons, say 5 - 10+ years, where long term uncertainties and compounding errors are more significant. Additionally there exists a natural topic imbalance in the questions being used both from market data and the synthetic pipeline. There exists a heavy concentration of questions in the domain of politics, economics and sports, while more niche domains like scientific discovery or research and specific regional events are underrepresented in the questions and prediction efforts. This most likely biases our model to these domains where the performance may be higher as these topics are represented at a high quantity.

Retrieval system constraints: Our retrieve system relies on Google News, which has information and publications which are dominated by English sources as well as having limited coverage pre-2010. This linguistics and temporal bias could impair performance on non-English sources or questions that have a longer historical context.

Synthetic data artifacts: The questions generated by the LLM may have subtle patterns or artifacts that exist in the LLM-generated questions that differs from the sentence and phrasing structure of true human-generated questions. This would potentially inflate performance on these synthetic questions as these questions may not accurately reflect true human-generated questions and induce a systematic bias in the distribution of questions and the model's performance on synthetic vs market data derived questions.

Model architecture: Utilization of the LLama3-8B-Intruct model may exhibit computational constraints when it comes to the model's efforts of forecasting. Larger models may handle predictions in a different way or generate a different world model's used for their predictions, as we are not sure how this approach generalize to different kinds of models, of larger or smaller scale.

6 Conclusion

This paper set out to investigate the ability of LLMs to be taught to make predictions about future events effectively. By asking the model to give a probabilistic answer combined with proper retrieval, prompting, and fine-tuning, combined with the generation of synthetic questions beyond the model's train date but which have already resolved in the present, we were able to successfully create a forecasting system and evaluate the system's performance without contamination from retrospective information that would not be available before the event's real life resolution.

Based on our results, we have found that by fine-tuning the LLM used for the main reasoning phase, we can significantly improve predictive performance, significantly outperforming the untrained zero-shot baseline version of the model. This suggests that LLMs are capable of internalizing important lines of reasoning used in making successful predictions, such as weighing different pieces of evidence, and accounting for uncertainty in a probabilistic fashion. However, the performance difference between LLMs trained on smaller subsets of data specific to a given domain, and LLMs trained on a larger number of examples of reasoning across a larger range of topics, was found to be minimal and insignificant, suggesting

Topic	Domain Model Score	General Model Score	Untrained Model Score
Arts & Recreation	0.205	0.214	0.284
Economics & Business	0.235	0.234	0.286
Education & Research	0.228	0.218	0.271
Healthcare & Biology	0.206	0.201	0.266
Politics & Governance	0.232	0.228	0.287
Science & Tech	0.232	0.203	0.289
Security & Defenses	0.216	0.232	0.273
Sports	0.211	0.205	0.265

Table 1: Brier scores by topic for domain-specific, general, abd models

that the exact nature of the fine-tuning training data is far less important than the presence of the data itself.

Beyond the modeling results, this work highlights the value of synthetic question generation as a robust benchmarking tool. Unlike traditional evaluations that rely on fixed datasets from prediction markets—which often suffer from topic imbalance or platform-specific biases—our synthetic approach provides a flexible framework to test models across broader, user-defined categories. This independence from third-party sources allows researchers to target specific under-represented domains and scale evaluation benchmarks on demand.

These findings carry significant implications for a variety of fields. Because the model is primarily trained for general reasoning in a variety of contexts, and we found that narrowing the context made little difference, the result is a model which is capable of predicting, with reasonably high and likely improvable accuracy, future events for any topic, expressed verbally, using retrieved verbal information. This has practical relevance for applications in a variety of fields where information across domains is relevant and must be synthesized and used for prediction, such as policy analysis, risk management, strategic planning, and prediction markets.

7 Future Work

Several promising directions could extend this work such as expanded model architectures, going beyond binary predictions, expanding the training data and considering long-horizon forecasting.

Expanded model architectures: For future work we could evaluate whether our findings generalizes across various model families such as GPT models, Claude, Gemini etc., and scales with these larger parameter models. Larger models may be able to understand more complex prediction tasks as well as potentially better reasoning efforts compared to smaller models. Additionally we could also explore a mixture of experts architecture where we utilize multiple models, combining domain-specific knowledge, expanding upon the current single model approach to see if this improves on predictive accuracy.

Beyond Binary Predictions: Binary predictions in this context provide clear evaluation metrics and an ideal method of training the model on predictive accuracy. Extending this framework to handle multiple-choice questions or continuous numerical predictions, we could move beyond the current binary question dataset to see how the model performs with broader applications across question types. This would require developing or using differing evaluation metrics from the Brier Score as well as differing fine-tuning approaches.

Expanded training data: Future datasets could be expanded upon with more diverse, validated and larger datasets. Accessing historical data directly from prediction market platforms could provide more insight to training signals and directions to fine-tune our models. Additionally, synthesizing questions for underrepresented domains could help address the topic imbalance in the categories observed in the dataset and improve the overall predictive accuracy.

Long-term Forecasting: Our research focused primarily on questions that were resolved within a few months of being asked. Taking this concept further, focusing on questions that span years or decades, would require a far different approach in the prediction efforts, world model, and fine-tuning approach. Extrapolating trends or incorporating structured causal models could contribute to LLMs' capabilities of strategic planning and long-term policies.

Acknowledgments

We thank Prof. Rick Stevens for his lecture insights and the teaching assistants of the Deep Learning Systems course for their constructive feedback. We also acknowledge the Data Science Institute for providing the computing resources essential for our experiments.

References

[1] Mahdi Abolghasemi, O Ganbold, and K Rotaru. 2023. Humans vs large language models: Judgmental forecasting in an era of advanced AI. *arXiv preprint arXiv:2312.06941* (2023).

[2] Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. 2024. Approaching Human-Level Forecasting with Language Models. arXiv:2402.18563 [cs.LG] https://arxiv.org/abs/2402.18563

[3] Woojeong Jin, Rahul Khanna, Suji Kim, Dong-Ho Lee, Fred Morstatter, Aram Galstyan, and Xiang Ren. 2021. ForecastQA: A Question Answering Challenge for Event Forecasting with Temporal Text Data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*.

[4] Philipp Schoenegger and Peter S Park. 2023. Large language model prediction capabilities: Evidence from a real-world forecasting tournament. *arXiv preprint arXiv:2310.13014* (2023).

[5] Q Yan, R Seraj, J He, L Meng, and T Sylvain. 2024. Autocast++: Enhancing World Event Prediction with Zero-Shot Ranking-Based Context Retrieval. In *International Conference on Learning Representations (ICLR)*.

[6] Qingchuan Yang, Simon Mahns, Sida Li, Anri Gu, Jibang Wu, and Haifeng Xu. 2025. LLM-as-a-Prophet: Understanding Predictive Intelligence with Prophet Arena. arXiv:2510.17638 [cs.AI] https://arxiv.org/abs/2510.17638

[7] 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. In *Advances in Neural Information Processing Systems*.