LEARNING TO SIMULATE INDIVIDUALS FROM MACRO SOCIAL SIGNALS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models are increasingly used to simulate how individuals respond to new situations, yet the behavioral reasoning behind these responses is either inherited from pretraining or learned from individual-level annotations, which offer limited behavioral diversity and little supervision of the reasoning itself. We propose to learn behavioral reasoning from prediction markets, whose price trajectories record how populations respond to real-world events at scale. We introduce macro2mind, which trains a language model with GRPO using market signals. A social behavioral decomposition makes behavioral reasoning an explicit step of forecasting: the model infers representative groups of market participants, predicts how each interprets the news and updates its beliefs, reasons about their interactions, and aggregates these responses into a price. A hindsight-regret curriculum then focuses training on transitions where representative groups identified in hindsight most improve the forecast, downweighting those dominated by noise or already mastered. The learned reasoning applies to user simulation without further training. On SWM-Bench, macro2mind achieves state-of-the-art directional accuracy and correlation on Polymarket. Trained on market data, it transfers zero-shot to four user-simulation benchmarks (Humanual, OvertonBench, PRISM, and CAD) and has leading performance among zero-shot methods. Used as a data generator, macro2mind also raises a downstream simulator's accuracy on unseen users by 15.5 points, outperforming data generated by its backbone by 13.2 points.

1 INTRODUCTION

Large language models are increasingly used to simulate individual behavior (Park et al., 2023; Zhou et al., 2024; Aher et al., 2023; Ni et al., 2026), including how people respond to information (Argyle et al., 2023; Santurkar et al., 2023), make decisions (Kitadai et al., 2025; Horton et al., 2023), and interact with others (Wang et al., 2023). Such simulators support personalized agents (Jiang et al., 2024; Shao et al., 2023), social simulation (Yang et al., 2024; Gao et al., 2023), and the evaluation of interactive systems (Wang et al., 2025b; Yao et al., 2024). Achieving high-fidelity simulation of a particular individual imposes a dual requirement. It demands not only static, descriptive information about that person's background but also a dynamic capability to predict how their internal beliefs and subsequent actions evolve. We refer to this capability as *behavioral reasoning*; it is closely related to theory of mind, the ability to attribute beliefs and intentions to others (Premack & Woodruff, 1978).

To improve response fidelity, training-based simulators directly fit models to observed user behaviors (Naous et al., 2026; Zhou et al., 2026); some frameworks further align latent user states with these observed outputs (Wu et al., 2026). However, while rich profiles and histories successfully specify *whom* to simulate, the core mechanics of *how* they reason are either passively inherited from pre-training priors or explicitly fitted to individual-level annotations. Consequently, the supervision for dynamic behavioral reasoning remains strictly confined to micro-level data.

Challenges of user simulation. Generalizing to unseen users and situations is the central challenge for current simulators, fundamentally bottlenecked by two data limitations. (**1**) **Limited behavioral diversity.** Capturing how people dynamically update beliefs and adapt to peers requires temporally dense, socially situated data. However, such data is exceedingly scarce: traditional surveys yield only static snapshots, massive micro-level collection is costly and privacy-sensitive (Gabriel et al., 2024; Durmus et al., 2023), and synthetic personas (Ge et al., 2024) lack grounding in real behavior. (**2**)

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

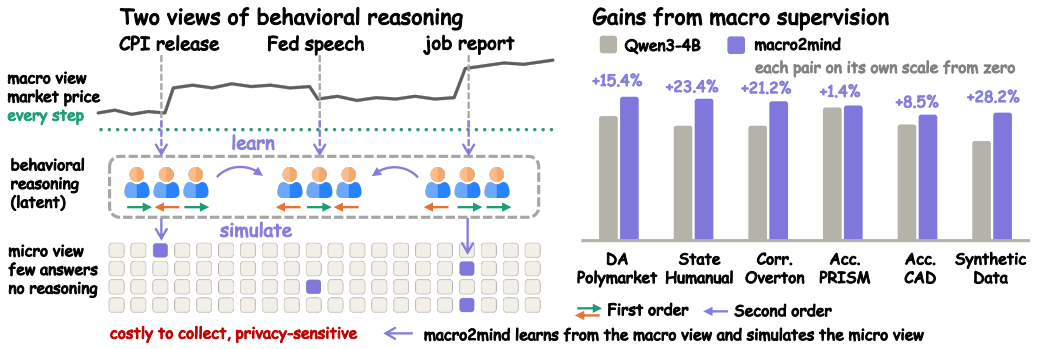

Figure 1: **Learning to simulate individuals from macro signals. Left:** Behavioral reasoning is observed densely through market prices but only sparsely through individual data; macro2mind learns from the former to simulate the latter zero-shot. **Right:** Relative gains of macro2mind over Qwen3-4B; for synthetic data, students are trained on data generated by each model.

Costly reasoning supervision. Supervising *how* people respond requires internal cognitive traces that are prohibitively expensive to collect. Relying on affordable alternatives introduces critical flaws: simplistic tasks encourage models to exploit superficial heuristics, while LLM-generated rationales often inject spurious reasoning disconnected from genuine cognition. Both prevent the acquisition of robust behavioral reasoning needed for generalization.

Market signals as complementary supervision. Prediction markets offer natural solutions to both bottlenecks, as their price trajectories continuously record collective responses to real-world events (Wolfers & Zitzewitz, 2004; Yu et al., 2026; Karger et al., 2025). (**1**) **Rich and abundant dynamics.** Each price movement reflects real-stakes decisions by diverse participants, intrinsically providing event-aligned, temporally dense behavioral data at scale without costly micro-level collection. (**2**) **Reasoning supervision via macroscopic outcomes.** Forecasting market shifts demands the exact behavioral reasoning necessary for individual simulation: inferring how distinct actors interpret news, update beliefs, and react to peers. The realized market price acts as a scalable reward signal that objectively grounds this reasoning process without requiring hand-annotated traces. This raises a concrete question: *can learning to predict aggregate outcomes teach behavioral reasoning that transfers to simulating individuals?*

Although prediction markets address the limitations of individual-level data, directly harnessing macroscopic outcomes introduces two unique learning challenges. First, because market supervision evaluates only the final numerical forecast, naive optimization risks rewarding shortcut learning (e.g., exploiting price momentum) rather than cultivating the genuine behavioral reasoning required for individual simulation (**C1**). Second, market trajectories suffer from an extremely low behavioral signal-to-noise ratio. Many transitions reflect routine fluctuations or are trivially predictable, while no explicit labels indicate which transitions contain rich, learnable social dynamics(**C2**).

Our approach. To overcome these bottlenecks, we introduce macro2mind, which tackles both challenges through two complementary mechanisms trained end-to-end via GRPO using only macroscopic forecasting rewards. First, a **social behavioral decomposition** strictly structures the policy's reasoning trajectory to combat shortcut learning (**C1**). The model must explicitly simulate how representative constituents (e.g., distinct demographic groups or market factions) interpret events and influence one another before aggregating these micro-responses into a final forecast. Because the forecast is assembled from these responses, the reward can optimize the underlying behavioral reasoning logic. Second, a **hindsight-regret curriculum** performs dynamic **data selection** to prioritize transitions rich in behavioral signal (**C2**). For each candidate transition, the policy first identifies, in hindsight, the constituents that drove the realized outcome; hindsight regret then measures how much its forecasting accuracy improves when these constituents are provided as a hint. This regret is low for noise-dominated or already-mastered transitions, and high when overlooking the right constituents is the main bottleneck. It can determine how training transitions are sampled, downweighting market noise and focusing training on learnable behavioral shifts. Ultimately, the decomposition dictates *how* the model reasons, while the data-selective curriculum dictates *where* that learning is most informative. Together, they transform noisy aggregate markets into a scalable engine for individual behavioral simulation.

Main findings. We train macro2mind on prediction-market transitions from SWM-Bench and evaluate both its macro forecasting accuracy and zero-shot transfer to four micro-level benchmarks (Humanual, OvertonBench, PRISM, and CAD). Our experiments answer the central question affirmatively through three key findings: (**1**) *Superior market forecasting.* macro2mind achieves state-of-the-art directional accuracy and correlations on Polymarket and highly competitive performance on Kalshi. (**2**) *Robust zero-shot user simulation.* Despite zero exposure to individual-level annotations, macro2mind consistently achieves leading performance among zero-shot methods across the downstream benchmarks. Remarkably, it even outperforms several supervised simulators trained explicitly on the target domains. (**3**) *Scalable synthetic supervision.* Behavioral trajectories generated by our zero-shot model serve as high-quality training data. Fine-tuning downstream user simulators on these synthesized responses boosts their accuracy by 15.5 absolute points, proving macro2mind to be a strictly superior teacher compared to its standard backbone.

2 Related Work

LLM-based User Simulation. Existing methods rely heavily on individual-level behavioral data, either by conditioning behavior via prompting (Park et al., 2023; 2024b) or by fine-tuning parameters on behavioral traces (Naous et al., 2026; Wu et al., 2026; Zhou et al., 2026). We depart from this reliance. By using aggregate outcomes to supervise individual behavioral reasoning, our approach requires user-specific data only at inference.

Aggregate Social Signals and Social Simulation. Forecasting systems treat aggregate market prediction as the end goal (Wolfers & Zitzewitz, 2004; Halawi et al., 2024; Karger et al., 2025; Yu et al., 2026). Conversely, social simulation runs bottom-up, using individual agents to generate collective dynamics (Gao et al., 2023; Yang et al., 2024; Zhang et al., 2025). We invert this paradigm, using aggregate outcomes to supervise individual reasoning. Instead of learning a static predictor over a fixed label space, we model unobserved constituents to acquire a transferable reasoning process.

Theory of Mind Modeling. Existing Theory of Mind (ToM) frameworks and benchmarks heavily rely on synthetic narratives or explicit human annotations (Gandhi et al., 2023; Kim et al., 2023; Wilf et al., 2024; Sclar et al., 2023; Zhou et al., 2025), sparking debate over whether LLMs perform robust causal reasoning or superficial pattern matching (Kosinski, 2024; Ullman, 2023; Sap et al., 2022). Our constituent-level decomposition directly operationalizes ToM by inferring how distinct groups update latent beliefs, but it induces this capability strictly from macroscopic market outcomes. By bypassing micro-level annotations entirely, we learn grounded mental-state reasoning that transfers zero-shot to concrete individuals.

3 Problem Formulation

We formalize two distinct but conceptually linked prediction tasks. The *macro task* predicts how a population-level signal evolves in response to real-world events. The *micro task* predicts how a specific individual responds to a given situation. To bridge these scales, we structure both tasks under a unified framework of *observation*, *policy*, and *evaluation*. In both settings, the observation captures the available contextual and historical state at a given time step; the policy π_θ is the generative model that processes this context to predict the subsequent outcome; and the evaluation protocol measures how accurately the policy's prediction aligns with the realized ground truth.

By unifying the input-output interfaces of both tasks, we can train a single policy π_θ exclusively on aggregate macro observations and evaluate it zero-shot on the micro task, reserving individual human responses strictly for testing. This setting directly evaluates our core research question: can behavioral reasoning learned purely from macroscopic outcomes transfer to simulating individuals?

3.1 Macro Task: Population-level Forecasting

Observation. Each instance centers on a market question q, a natural-language proposition (e.g., the outcome of an election). Its price v_t reflects the collective consensus of market participants at step t. The model observes the question, a recent price history s_t, and an event set E_t comprising text descriptions of real-world occurrences. We denote the full macro observation as $x_t = (q, s_t, E_t)$.

Policy. Given observation x_t, the policy outputs a forecast value of the next price, $\hat{v}_{t+1} \sim \pi_\theta(\cdot \mid x_t)$.

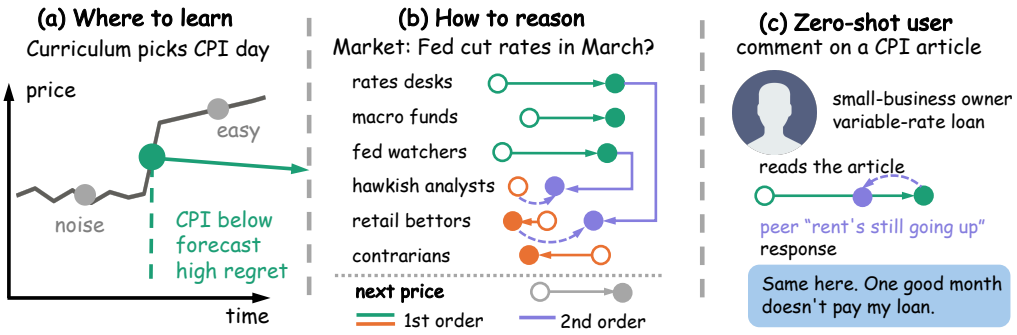

Figure 2: **The pipeline of macro2mind, illustrated with a running example.** (a) **Where to learn:** The hindsight-regret curriculum performs data selection by focusing on high-regret transitions, uch as the price jump after inflation data came in lower than expected, while ignoring the noise. (b) **How to reason:** Through social behavioral decomposition, the model infers how different groups react to the news (first order) and to one another (second order), and aggregates their belief updates into a forecast rewarded by the realized price. (c) **Zero-shot transfer:** This learned reasoning ability is transferred zero-shot to user simulations, effectively modeling how an individual (e.g., a small-business owner) reacts to the CPI news and peer comments.

Evaluation. The macro dataset collects historical transitions across multiple markets, denoted as $\mathcal{D}_{\text{macro}} = \{(x_t, v_{t+1})\}$. The task is evaluated by how accurately the forecast $\hat{v}_{t+1}$ matches the realized next-step price v_{t+1}, utilizing MAE, MASE, directional accuracy, and Pearson correlation.

3.2 Micro Task: Individual-level Simulation

Observation. Each instance targets a specific user. At step t, the model observes the user's persona and interaction history h_t (e.g., profiles, stated views, or past messages), alongside the current situational context x_t (e.g., a news event, an ongoing conversation, or a candidate reply to be judged). All observed information is strictly available prior to the target response.

Policy. Given the observation (h_t, x_t), the policy predicts the user's response, $\hat{r}_t \sim \pi_\theta(\cdot \mid h_t, x_t)$.

Evaluation. The micro dataset collects individual interactions, $\mathcal{D}_{\text{micro}} = \{(h_t, x_t, r_t)\}$. The task is evaluated by how well the predicted response $\hat{r}_t$ aligns with the actual behavior r_t, using task-specific scores such as rubric-based alignment for free-text replies, or error and rank correlation for ratings.

4 Learning to Simulate Individuals from Macro Signals

We realize our approach through a single autoregressive language-model policy π_θ. We first describe how the policy structures social behavioral reasoning during forecasting (Section 4.1), how it is optimized via GRPO using a hindsight-regret curriculum (Section 4.2), and how this learned reasoning transfers to individual simulation (Section 4.3).

4.1 Social Behavioral Reasoning

Orders of social behavioral reasoning. Social behavioral reasoning naturally unfolds in successive orders. First-order reasoning infers how individuals immediately react to an external stimulus. Higher-order reasoning models the cascading interactions triggered by these initial reactions: at order k, it evaluates how individuals adapt to the responses from order $k-1$. In individual simulation, this mirrors predicting how an exchange evolves over subsequent conversational turns; in market forecasting, it tracks how participants first react to news and subsequently influence one another. In both paradigms, while the ultimate prediction target remains unchanged, modeling these higher orders significantly enriches the underlying reasoning. To maintain computational tractability while capturing essential social dynamics, we truncate this expansion at the second order, tracking only highly consequential interactions for each event.

Social behavioral decomposition. We instantiate this reasoning as a single autoregressive trajectory, avoiding the cost and credit-assignment difficulties of a full multi-agent simulation (Yang et al., 2024).

Given an observation x, the policy generates a reasoning trace $\tau = (\tau^{\text{prop}}, \tau^F, \tau^A)$ before outputting the forecast $\hat{v}$. Each step has a direct counterpart in individual simulation.

1. Constituent initialization (τ^{prop}). The policy identifies representative constituents likely to drive the market shift, and constructs a prior state b for each that describes its role, objectives, and beliefs. *Micro counterpart:* inferring which constituents drive a given change, and what they care about, requires linking behavioral patterns to the kinds of people who produce them. This is the same persona–behavior association that individual simulation relies on.

2. First-order responses (τ^F). The policy predicts how each constituent updates its beliefs and responds based on its own prior b and the observation x. *Micro counterpart:* predicting a user's belief update and immediate response to the current context.

3. Second-order responses and aggregation (τ^A). The policy models the most consequential interactions, assessing how one constituent's reaction alters another's beliefs and actions, and then aggregates the adjusted responses into the forecast $\hat{v}$. *Micro counterpart:* adapting to others' reactions, as in replying within a multi-turn dialogue.

Behavioral reasoning as a shared transition function. Both tasks rest on a shared behavioral transition function F, which maps an agent's prior state b and context x to a belief update and an observable response, $(b^+, r) \sim F(\cdot \mid b, x)$, and can be viewed as a computational form of theory of mind. The two tasks differ in the observability of the prior state. In the micro task, it is given by the user's history ($b = h$). In the macro task, it is unobserved, and the policy must first infer the prior states of key constituents before predicting their responses. Our policy parameterizes this function as F_θ within τ^F. Because F_θ is shared across constituents and tasks, the behavioral reasoning ability that the policy learns from macro rewards can be applied to individual simulation without further training (Section 4.3).

4.2 Learning Social Behavioral Reasoning from Macro Signals

Algorithm 1 macro2mind training with hindsight-regret curriculum

Require: $\mathcal{D}_{\text{macro}}$; π_θ; reward R; group size M; difficulty filter d; weights $\lambda_{\text{reg}}, \lambda_{\text{diff}}$; regret scale s
Ensure: Trained parameters θ

for each training step **do**
  if periodic probe **then**
    for each (x, v) in a sampled probe pool from $\mathcal{D}_{\text{macro}}$ **do**
      $c \sim \pi_\theta^{\text{hint}}(\cdot \mid x, v)$ ▷ *hindsight hint*
      $y_{1:M} \sim \pi_\theta(\cdot \mid x)$, $y^{\text{H}}_{1:M} \sim \pi_\theta(\cdot \mid x, c)$
      $\Delta(x) \leftarrow \mathbb{E}\big[\tilde{R}(y^{\text{H}})\big] - \mathbb{E}\big[\tilde{R}(y)\big]$ $p(x) \leftarrow \Pr\big(\tilde{R}(y) \geq 0.5\big)$ ▷ *Eq. equation 1*
  $w(x) \leftarrow \lambda_{\text{reg}} \min\big(\max(\Delta(x), 0)/s, 1\big) + \lambda_{\text{diff}}\, d(p(x))$ ▷ *Eq. equation 2*
  $\mathcal{B} \sim \mathcal{D}_{\text{macro}} \propto w(x)$ ▷ *sample training batch*
  for $(x, v) \in \mathcal{B}$ **do**
    $y_{1:M} \sim \pi_\theta(\cdot \mid x)$; update $p(x)$ from $\tilde{R}(y_{1:M})$
  $\theta \leftarrow$ GRPO-Update$(\theta, \mathcal{B}, R)$ ▷ *group-normalized advantages*
return θ

Macro signals are inherently noisy and uneven in difficulty (**C2**): many transitions are dominated by random fluctuations, some are trivially solvable, and only some actually reward reasoning about the latent constituents driving the market. Because GRPO gains minimal signal from transitions the model always or never solves, we augment policy optimization with a hindsight-regret curriculum, preferentially sampling transitions where the model fails simply because it overlooks key constituents.

Policy optimization. To bootstrap the structured reasoning format and stabilize subsequent reinforcement learning, we first perform a lightweight supervised fine-tuning (SFT) warmup on a small seed set of structured behavioral traces. For a given market observation x, the policy samples a group of M reasoning and forecasting trajectories. Each output is evaluated by a reward R that combines formatting, numerical accuracy, and directional accuracy (Appendix B). We optimize the policy via GRPO (Shao et al., 2024) using group-normalized advantages. To isolate forecasting capability, the curriculum below relies on a format-free reward $\tilde{R} \in [0, 1]$, which drops the format term and renormalizes the remaining terms, so that it measures forecast quality alone.

Hindsight-regret curriculum. We formulate the curriculum as a dynamic data sampler. Periodically, we probe a random subset of transitions to estimate their learning value. For a given transition, the current policy is provided the true outcome v via a dedicated hint prompt and tasked with identifying the key constituents that drove the market. This generates a hint c containing constituent names and their qualitative price directions (up or down only).

We inject c into the standard prompt as a simulated analyst's insight, and sample M hinted responses $y^{\mathrm{H}} \sim \pi_\theta(\cdot \mid x, c)$ alongside M standard unhinted responses $y \sim \pi_\theta(\cdot \mid x)$. Building upon the regret formulation introduced in SPADE (Liu et al., 2026a), we compute the hindsight regret Δ and the baseline win rate p, estimating both from these samples:

$$\Delta = \mathbb{E}\big[\tilde{R}(y^{\mathrm{H}})\big] - \mathbb{E}\big[\tilde{R}(y)\big],\ p = \Pr\big(\tilde{R}(y) \geq 0.5\big). \tag{1}$$

Here, Δ isolates the capability gap: what the policy can successfully explain when aware of the key constituents, yet fails to anticipate without them. Because c is generated by the policy itself, this gap shifts as the model improves.

Sampling weights. Following SPADE (Liu et al., 2026a), each transition receives a composite weight derived from Δ and p:

$$w = \lambda_{\mathrm{reg}} \min\Big(\frac{\max(\Delta, 0)}{s},\ 1\Big) + \lambda_{\mathrm{diff}}\, d(p), \tag{2}$$

where λ_{reg} and λ_{diff} weight the two terms, s scales the regret, and d is a trapezoidal difficulty filter over the win rate. The regret term keeps only positive capability gaps ($\Delta > 0$) and caps them at 1. The filter $d(p)$ peaks for intermediate win rates and decays linearly toward zero for transitions that are excessively hard ($p \approx 0$) or trivially easy ($p \approx 1$). This focuses GRPO on transitions with more informative gradient signals. The win rate p is also updated during the regular GRPO training steps. Hyperparameter values are listed in Appendix D.

4.3 Applying Social Behavior Reasoning to Individual Simulation

The core premise of our approach is that macroscopic market shifts and microscopic user actions are driven by the same underlying behavioral mechanics. Consequently, we transfer the reasoning capabilities learned from macro signals directly to individual simulation.

Individual-level social reasoning. For individual simulation, the prior state is given rather than inferred. Depending on the task, it comes from the user's interaction history, such as past posts or conversations, or from a persona, such as a profile and stated views. We place this information in the constituent-level schema of τ^F as the prior state $b = h$, with the current context x as the observation. The policy then reasons about how the user interprets this context and updates their beliefs, and derives the response $\hat{r}$ in the format of the target task, such as a free-text reply or a rating.

Inference. We apply the trained policy to individual simulation zero-shot, without any parameter updates, and evaluate only the final response. During training, the policy reasons about coarse groups inferred from market context; here, it applies the same behavioral reasoning process to a specific person with an observed history or persona.

Synthetic data. The same procedure can also generate training data to address the scarcity of individual behavioral data. Given a user's persona and a few-shot history, we apply the trained policy to generate both novel questions and the corresponding responses. This synthetic data is then used to fine-tune and align a separate downstream simulator.

5 Main Experiments

5.1 Problem Setup

Market Prediction. We train and evaluate our model on **SWM-Bench** (Yu et al., 2026), which pairs prediction-market trajectories with contemporaneous news. The task is to predict the next observed market price, providing aggregate supervision without requiring any annotations of the underlying individual behaviors. Full dataset composition details are provided in Appendix F.3.2.
User Simulation. To evaluate the zero-shot transfer of behavioral reasoning across both single-turn and multi-turn interactions, we select four benchmarks spanning three distinct individual-level tasks:

Table 1: **Results on SWM-Bench. Bold** / underline: best / second best results.

Method	Polymarket				Kalshi			
	MASE↓	MAE↓	DA↑	Corr↑	MASE↓	MAE↓	DA↑	Corr↑
Time-series only								
Autoformer (Wu et al., 2021)	1.445	0.062	0.519	0.205	1.596	0.096	0.473	0.008
DLinear (Zeng et al., 2023)	1.111	0.048	0.580	0.348	1.245	0.075	0.492	−0.035
Informer (Zhou et al., 2021)	1.405	0.060	0.570	0.257	1.367	0.083	0.459	−0.013
iTransformer (Liu et al., 2024)	0.998	0.043	0.585	0.237	1.150	0.069	0.532	−0.078
TimeMixer (Wang et al., 2024a)	0.997	0.043	0.590	0.224	1.079	0.065	0.549	−0.056
TimesNet (Wu et al., 2022)	1.015	0.044	0.577	0.264	1.133	0.068	0.543	−0.023
PatchTST (Nie et al., 2022)	0.994	0.043	0.592	0.306	1.174	0.071	0.536	−0.035
Time-series + news (Prompting-based LLM)								
TimeCAP (Lee et al., 2025)	1.024	0.044	0.347	0.119	1.028	0.062	0.429	0.020
Qwen3-4B (Yang et al., 2025)	2.297	0.099	0.584	0.088	1.848	0.112	0.559	0.098
Qwen3-4B-think (Yang et al., 2025)	2.602	0.112	0.511	0.075	1.951	0.118	0.529	0.030
GPT-5.5 (OpenAI, 2026)	**0.985**	**0.042**	0.337	0.264	**0.997**	**0.060**	0.564	**0.242**
Time-series + news (Fine-tuned LLM)								
Time-LLM (Jin et al., 2023)	0.998	0.043	0.576	0.319	1.126	0.068	0.567	0.018
ChatTime (Wang et al., 2025a)	1.186	0.051	0.560	0.220	1.458	0.088	0.514	−0.028
FNF (Wang et al., 2024b)	0.986	**0.042**	0.377	0.218	1.126	0.068	0.430	−0.020
LLM4TS (Chang et al., 2025)	0.993	0.043	0.589	0.312	1.125	0.068	0.554	−0.015
LLMForecasting (Halawi et al., 2024)	1.224	0.053	0.457	0.100	1.185	0.072	0.521	0.091
SWM (Yu et al., 2026)	1.141	0.049	0.535	0.088	1.013	0.061	**0.589**	0.167
GRPO	1.044	0.045	0.475	0.012	1.001	**0.060**	0.448	0.031
macro2mind(4B)	0.999	0.043	**0.674**	**0.362**	1.088	0.066	0.569	0.016

free-text response generation (**Humanual** (Wu et al., 2026), Qwen3.8-Max as judge), representation judgment (**OvertonBench** (Poole-Dayan et al., 2026)), and conversational preference prediction (**PRISM** (Kirk et al., 2024)[1] and **CAD** (Zhang et al., 2026a)).

Synthetic data. We evaluate on **Twin-2K-500** (Toubia et al., 2025). Each generator writes new scenarios for 600 training users and labels them by answering as each participant, conditioned on the persona and the answer distribution of training users. A Qwen3-4B student is fine-tuned on each synthetic set and evaluated on real held-out questions, either for 100 unseen users or for the source users themselves. Details are in Appendix G.

5.2 Baseline settings.

Market Prediction. We compare macro2mind against three categories of methods. (**1**) *Time-series only* baselines forecast solely from the numerical historical window and are trained on SWM-Bench: Autoformer, DLinear, Informer, iTransformer, PatchTST, TimeMixer, and TimesNet. (**2**) *Prompting-based LLMs* process both the numerical window and all available news to generate forecasts via zero-shot prompting without parameter updates: Qwen3-4B, GPT-5.5, and TimeCAP. (**3**) *Fine-tuned LLMs* also use both modalities but are trained directly on SWM-Bench. SWM, Time-LLM, FNF, LLMForecasting, and LLM4TS all use Qwen3-8B, while ChatTime requires its own ChatTime-1-7B-Base. Our model use Qwen3-4B as backbone. All trainable baselines, are trained jointly on Polymarket and Kalshi data, but evaluated separately.

User Simulation. We compare macro2mind against four categories of methods. (**1**) *Same-backbone baselines*: zero-shot Qwen3-4B and standard GRPO trained on SWM-Bench. (**2**) *Prompting-based simulators*: training-free inference methods, including Qwen3-4B, PERSONA-Flow, SynthesizeMe, Self-consistency, History-ICL and Self-report. (**3**) *Trained user simulators*: fine-tuned models including HumanLM-SFT, -SFT-think, and -GRPO. (**4**) *Personalized reward models*, repurposed as simulators: the generative P-GenRM, alongside the scalar models Skywork-RM, MRM, and WIMHF-RM. Scalar reward models generate no text and instead rerank eight Qwen3-4B samples. All methods use Qwen3-4B except Skywork-RM, and MRM. Details in Appendix. E.

[1]Following the original paper, we multiply the prompted 0–1 scores by 100 to report results on a 0–100 scale.

Table 2: **Results on user simulation tasks.** Parentheses denote training data (*per bench.*: one model trained per benchmark). **Bold** / underline: best / second best among zero-shot (out-of-domain) results. We multiply all Humanual scores by 100 for reporting.

	Humanual		Overton		PRISM		CAD
Method	State↑	Resp.↑	MAE↓	ρ ↑	MAE↓	Acc.↑	Acc.↑
Trained user simulators							
HumanLM-SFT (Humanual) (Wu et al., 2026)	8.2	3.2	0.734	0.566	22.5	59.6	35.0
HumanLM-SFT-think (Humanual)	6.3	1.8	0.754	0.606	23.3	60.1	36.5
HumanLM-GRPO (Humanual)	12.4	5.3	0.718	0.576	21.8	62.6	39.1
Personalized reward models							
P-GenRM (PRISM) (Zhang et al., 2026b)	3.4	0.7	0.786	0.455	21.8	66.2	**44.0**
Skywork-RM (pretrained) (Liu et al., 2026b)	8.3	2.4	1.024	0.090	21.5	**64.7**	41.8
MRM (per bench.) (Cai et al., 2026)	7.8	2.5	0.907	0.338	20.7	69.1	44.2
WIMHF-RM (per bench.) (Movva et al., 2026)	7.9	2.6	0.859	0.429	20.7	69.5	43.6
Prompting-based simulators							
PERSONA-Flow (Feng et al., 2026)	7.5	2.1	0.756	0.540	21.5	62.5	38.8
SynthesizeMe (Ryan et al., 2025)	7.7	2.2	0.763	0.529	21.3	63.2	40.4
Self-cons. (n=5) (Wang et al., 2022)	7.9	2.1	0.758	0.538	21.3	62.6	38.6
History-ICL (Hwang et al., 2023)	8.8	2.9	0.741	0.552	**20.5**	63.6	42.9
Self-report (Park et al., 2024a)	6.9	1.8	0.750	0.544	21.5	62.7	38.5
Same-backbone controls							
Qwen3-4B (zero-shot)	7.7	2.4	0.756	0.539	21.5	62.4	38.9
GRPO (SWMBench, zero-shot)	6.1	1.5	**0.623**	0.647	23.0	53.9	36.6
macro2mind (SWMBench, zero-shot)	**9.5**	**3.4**	0.635	**0.653**	21.2	63.3	42.2

5.3 Evaluation settings.

For **Market Prediction**, we evaluate Mean Absolute Scaled Error (MASE), Mean Absolute Error (MAE), Directional Accuracy (DA) and Pearson Correlation (Corr). For **User Simulation**, we evaluate predictions against ground-truth behaviors. We report state and response alignment via LLM-as-a-judge on Humanual-Politics; MAE and correlations ρ on OvertonBench; MAE and preference accuracy on PRISM. Preference accuracy on CAD. Details in Appendix C.2.

5.4 Main Results

macro2mind improves market forecasting, particularly in predicting directional movement. On SWM-Bench (Table 1), macro2mind achieves the highest directional accuracy (0.674) and correlation (0.362) on Polymarket—surpassing both the strongest time-series baseline and GPT-5.5—while ranking second in directional accuracy on Kalshi (0.569).

macro2mind demonstrates robust zero-shot transfer. Trained on prediction market signals, macro2mind has leading performance among zero-shot methods (Table 2). It secures the best Humanual alignment (9.5/3.4) and OvertonBench ρ (0.653), remarkably surpassing two supervised HumanLM variants zero-shot. Furthermore, it firmly out-generalizes domain-specific simulators: macro2mind beats Humanual-trained HumanLM-GRPO across all five out-of-domain metrics (e.g., CAD: 42.2% vs. 39.1%) and avoids PRISM-trained P-GenRM's catastrophic collapse on Humanual. On PRISM and CAD accuracy, macro2mind remains highly competitive (63.3% / 42.2%), trailing only History-ICL and dataset-specific reward models.

macro2mind serves as a highly effective synthetic data generator for downstream user simulators. Fine-tuning a student model on data generated by macro2mind boosts its accuracy from 44.5% to 60.0% on unseen users (out-of-user) and from 50.9% to 61.3% on source users (in-user). In contrast, data generated by its Qwen3-4B backbone using the identical pipeline yields only marginal improvements (46.8% and 53.0%, respectively). Since the two generators differ exclusively in their exposure to macro market forecasting, this 13.2-point gap on unseen users demonstrates that macro2mind is a good teacher for micro-level simulation.

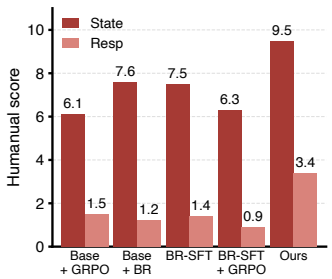

Figure 3: **Component ablation results on Humanual.** We use Qwen3-4B as base model.

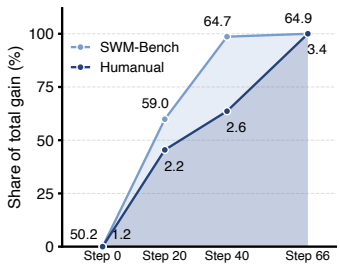

Figure 4: **Training dynamics.** SWM-Bench DA and Humanual Response score across training checkpoints.

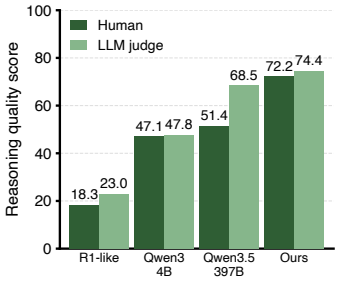

Figure 5: **Human evaluation of reasoning plausibility.** Pearson $r = 0.885$.

6 Discussion

RQ1: Which part of the training recipe drives transfer? *Ablation results reveal that the hindsight-regret curriculum is the essential driver of zero-shot transfer.* To isolate their contributions, we evaluate additive ablations on Humanual: Base+BR (zero-shot prompting), BR-SFT (fine-tuning on BR traces), BR-SFT+GRPO (GRPO without curriculum), Base+GRPO (GRPO with free form thinking) and macro2mind (full curriculum) under identical hyperparameters. Results (Figure 3) show the BR structure alone cannot explain the gains: Base+BR and BR-SFT yield similar State scores (7.6 vs. 7.5). Neither BR-SFT nor BR-SFT+GRPO improves over Base+BR: BR-SFT stays at 7.5 State and 1.4 Resp, and adding GRPO without the curriculum gives 6.3 State and 0.9 Resp. Thus, naively optimizing for market forecasting does not inherently transfer to individual simulation. Meaningful transfer emerges only when adding the hindsight-regret curriculum: macro2mind reaches 9.5 State and 3.4 Resp, improving over Base+BR by 25% in State score.

RQ2: Do macro forecasting and micro simulation ability improve together during training? *Results indicate a concurrent improvement during training. As macro2mind learns to forecast macro signals, its zero-shot user-simulation capability exhibits corresponding gains.* Evaluating four training checkpoints (steps 0, 20, 40, and 66) on both SWM-Bench and Humanual (Figure 4) reveals a striking trajectory. Over the first 40 steps, both abilities surge simultaneously—SWM-Bench directional accuracy increases from 50.2% to 64.7%, while Humanual response alignment more than doubles ($1.2 \rightarrow 2.6$) despite zero exposure to its data. Beyond step 40, forecasting accuracy plateaus (64.9%), yet response alignment continues to climb to nearly three times its initial value (3.4). This continuous co-improvement demonstrates that zero-shot transfer is directly driven by the acquisition of macro behavioral reasoning.

RQ3: Does the learned behavioral reasoning align with human judgments? *Human evaluation indicates that the behavioral reasoning generated by macro2mind aligns well with human judgments.* Five annotators evaluated 20 traces per model (0–100 scale), judging whether the inferred groups, news interpretations, and belief updates coherently explained the final forecast. As shown in Figure 5, macro2mind scores 72.2, surpassing both Qwen3-4B (47.1) and Qwen3.5-397B (51.4) under the exact same structural prompt. This suggests that structural prompting alone is insufficient to achieve human-aligned reasoning. In contrast, an R1-style model reasoning freely within `<think>` tags scores 18.3, often defaulting to disjointed numerical speculation rather than modeling actual human perspectives. These human ratings correlate strongly with our LLM judge (Pearson $r = 0.885$).

7 Conclusion

We introduced macro2mind, which learns to simulate individual human behavior from aggregate prediction-market signals via a structured social behavioral decomposition and a dynamic hindsight-regret curriculum. macro2mind achieves state-of-the-art DA and Corr on Polymarket while demonstrating robust zero-shot transfer to user simulation. Furthermore, its generated data can serve as high-quality supervision, significantly boosting the accuracy of downstream models. In the future, exploring other forms of macroscopic supervision and deploying these simulators into dynamic, multi-agent ecosystems remain exciting directions.

AI USE STATEMENT

AI systems, including LLMs, did not contribute to the development of the research ideas or the writing of this paper to a degree that would warrant authorship or contributor status. In this work, LLMs are treated strictly as objects of study and are the primary focus of our experimental evaluations.

ETHICS STATEMENT

This work introduces macro2mind, it is a methodological contribution rather than a deployed system, and its societal impact is largely mediated by how downstream practitioners apply it. All experiments are conducted on standard, publicly available benchmarks and open-weight or proprietary models used strictly for research evaluation; the work does not involve the collection of new personal data.

REPRODUCIBILITY STATEMENT

We describe the full training and evaluation setup needed to reproduce our main results in Section 4 (the macro2mind algorithm) and Section 5 (tasks and baselines), with complete per-method hyper-parameters given in Appendix. Evaluation metrics are formally defined in Appendix §C. We will release our training code and configurations upon publication.

REFERENCES

Gati V Aher, Rosa I Arriaga, and Adam Tauman Kalai. Using large language models to simulate multiple humans and replicate human subject studies. In *International conference on machine learning*, pp. 337–371. PMLR, 2023.

Lisa P Argyle, Ethan C Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. Out of one, many: Using language models to simulate human samples. *Political Analysis*, 31(3):337–351, 2023.

Hongru Cai, Yongqi Li, Tiezheng Yu, Fengbin Zhu, Wenjie Wang, Fuli Feng, and Wenjie Li. One adapts to any: Meta reward modeling for personalized llm alignment. *arXiv preprint arXiv:2601.18731*, 2026.

Ching Chang, Wei-Yao Wang, Wen-Chih Peng, and Tien-Fu Chen. Llm4ts: Aligning pre-trained llms as data-efficient time-series forecasters. *ACM Transactions on Intelligent Systems and Technology*, 16(3):1–20, 2025.

Esin Durmus, Karina Nguyen, Thomas I Liao, Nicholas Schiefer, Amanda Askell, Anton Bakhtin, Carol Chen, Zac Hatfield-Dodds, Danny Hernandez, Nicholas Joseph, et al. Towards measuring the representation of subjective global opinions in language models. *arXiv preprint arXiv:2306.16388*, 2023.

Xiachong Feng, Liang Zhao, Weihong Zhong, Yuxuan Gu, Lingpeng Kong, Xiaocheng Feng, Bing Qin, et al. Persona: Dynamic and compositional inference-time personality control via activation vector algebra. In *International Conference on Learning Representations*, volume 2026, pp. 151839–151879, 2026.

Iason Gabriel, Arianna Manzini, Geoff Keeling, Lisa Anne Hendricks, Verena Rieser, Hasan Iqbal, Nenad Tomašev, Ira Ktena, Zachary Kenton, Mikel Rodriguez, et al. The ethics of advanced ai assistants. *arXiv preprint arXiv:2404.16244*, 2024.

Kanishk Gandhi, Jan-Philipp Fränken, Tobias Gerstenberg, and Noah Goodman. Understanding social reasoning in language models with language models. *Advances in Neural Information Processing Systems*, 36:13518–13529, 2023.

Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. S3: Social-network simulation system with large language model-empowered agents. *arXiv preprint arXiv:2307.14984*, 2023.

Tao Ge, Xin Chan, Xiaoyang Wang, Dian Yu, Haitao Mi, and Dong Yu. Scaling synthetic data creation with 1,000,000,000 personas. *arXiv preprint arXiv:2406.20094*, 2024.

Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. Approaching human-level forecasting with language models, 2024.

John J Horton, Apostolos Filippas, and Benjamin S Manning. Large language models as simulated economic agents: What can we learn from homo silicus? Technical report, National Bureau of Economic Research, 2023.

EunJeong Hwang, Bodhisattwa Majumder, and Niket Tandon. Aligning language models to user opinions. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 5906–5919, 2023.

Hang Jiang, Xiajie Zhang, Xubo Cao, Cynthia Breazeal, Deb Roy, and Jad Kabbara. Personallm: Investigating the ability of large language models to express personality traits. In *Findings of the association for computational linguistics: NAACL 2024*, pp. 3605–3627, 2024.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*, 2023.

Ezra Karger, Houtan Bastani, Chen Yueh-Han, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E Tetlock. Forecastbench: A dynamic benchmark of ai forecasting capabilities. In *International Conference on Learning Representations*, volume 2025, pp. 93943–93980, 2025.

Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Bras, Gunhee Kim, Yejin Choi, and Maarten Sap. Fantom: A benchmark for stress-testing machine theory of mind in interactions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 14397–14413, 2023.

Hannah R Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, et al. The prism alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *Advances in Neural Information Processing Systems*, 37:105236–105344, 2024.

Ayato Kitadai, Sinndy Dayana Rico Lugo, Yudai Tsurusaki, Yusuke Fukasawa, and Nariaki Nishino. Can ai with high reasoning ability replicate human-like decision making in economic experiments? *Group Decision and Negotiation*, 34(6):1303–1326, 2025.

Michal Kosinski. Evaluating large language models in theory of mind tasks. *Proceedings of the National Academy of Sciences*, 121(45):e2405460121, 2024.

Geon Lee, Wenchao Yu, Kijung Shin, Wei Cheng, and Haifeng Chen. Timecap: Learning to contextualize, augment, and predict time series events with large language model agents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 18082–18090, 2025.

Bo Liu, Simon Yu, Yiding Jiang, Ao Qu, Andrew Zhao, Zichen Liu, Junsu Kim, Zijian Zhou, Seungone Kim, Tongzheng Ren, et al. Spade: Self-play in adaptive synthetic executable environments. *arXiv preprint arXiv:2608.19197*, 2026a.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 11116–11140, 2024.

Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiacai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, et al. Skywork-reward-v2: Scaling preference data curation via human-ai synergy. In *International Conference on Learning Representations*, volume 2026, pp. 133805–133838, 2026b.

Rajiv Movva, Smitha Milli, Sewon Min, and Emma Pierson. What's in my human feedback? learning interpretable descriptions of preference data. In *International Conference on Learning Representations*, volume 2026, pp. 59420–59455, 2026.

Tarek Naous, Philippe Laban, Wei Xu, and Jennifer Neville. Flipping the dialogue: Training and evaluating user language models. In *International Conference on Learning Representations*, volume 2026, pp. 82079–82103, 2026.

Bo Ni, Yu Wang, Leyao Wang, Branislav Kveton, Franck Dernoncourt, Yu Xia, Hongjie Chen, Reuben Luera, Samyadeep Basu, Subhojyoti Mukherjee, et al. A survey on llm-based conversational user simulation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4266–4301, 2026.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

OpenAI. Introducing gpt-5.5. https://openai.com/index/introducing-gpt-5-5/, April 2026. Accessed: 2026-05-29.

Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology*, pp. 1–22, 2023.

Joon Sung Park, Carolyn Q Zou, Jonne Kamphorst, Niles Egan, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Percy Liang, Robb Willer, et al. Llm agents grounded in self-reports enable general-purpose simulation of individuals. *arXiv preprint arXiv:2411.10109*, 2024a.

Joon Sung Park, Carolyn Q Zou, Aaron Shaw, Benjamin Mako Hill, Carrie Cai, Meredith Ringel Morris, Robb Willer, Percy Liang, and Michael S Bernstein. Generative agent simulations of 1,000 people. *arXiv preprint arXiv:2411.10109*, 52, 2024b.

Elinor Poole-Dayan, Jiayi Wu, Taylor Sorensen, Jiaxin Pei, and Michiel Bakker. Benchmarking overton pluralism in llms. In *International Conference on Learning Representations*, volume 2026, pp. 40773–40812, 2026.

David Premack and Guy Woodruff. Does the chimpanzee have a theory of mind? *Behavioral and brain sciences*, 1(4):515–526, 1978.

Michael J Ryan, Omar Shaikh, Aditri Bhagirath, Daniel Frees, William Held, and Diyi Yang. Synthesizeme! inducing persona-guided prompts for personalized reward models in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8045–8078, 2025.

Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. Whose opinions do language models reflect? In *International conference on machine learning*, pp. 29971–30004. PMLR, 2023.

Maarten Sap, Ronan Le Bras, Daniel Fried, and Yejin Choi. Neural theory-of-mind? on the limits of social intelligence in large lms. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 3762–3780, 2022.

Melanie Sclar, Sachin Kumar, Peter West, Alane Suhr, Yejin Choi, and Yulia Tsvetkov. Minding language models'(lack of) theory of mind: A plug-and-play multi-character belief tracker. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13960–13980, 2023.

Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-llm: A trainable agent for role-playing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 13153–13187, 2023.

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.

Olivier Toubia, George Z Gui, Tianyi Peng, Daniel J Merlau, Ang Li, and Haozhe Chen. Database report: Twin-2k-500: A data set for building digital twins of over 2,000 people based on their answers to over 500 questions. *Marketing Science*, 44(6):1446–1455, 2025.

Tomer Ullman. Large language models fail on trivial alterations to theory-of-mind tasks. *arXiv preprint arXiv:2302.08399*, 2023.

Chengsen Wang, Qi Qi, Jingyu Wang, Haifeng Sun, Zirui Zhuang, Jinming Wu, Lei Zhang, and Jianxin Liao. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12694–12702, 2025a.

Lei Wang, Jingsen Zhang, Hao Yang, Zhi-Yuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Hao Sun, Ruihua Song, et al. User behavior simulation with large language model-based agents. *ACM Transactions on Information Systems*, 43(2):1–37, 2025b.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. Timemixer: Decomposable multiscale mixing for time series forecasting. In *International conference on learning representations*, volume 2024, pp. 38626–38652, 2024a.

Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection, 2024b. URL https://arxiv.org/abs/2409.17515.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Zhilin Wang, Yu Ying Chiu, and Yu Cheung Chiu. Humanoid agents: Platform for simulating human-like generative agents. In *Proceedings of the 2023 conference on empirical methods in natural language processing: system demonstrations*, pp. 167–176, 2023.

Alex Wilf, Sihyun Lee, Paul Pu Liang, and Louis-Philippe Morency. Think twice: Perspective-taking improves large language models' theory-of-mind capabilities. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8292–8308, 2024.

Justin Wolfers and Eric Zitzewitz. Prediction markets. *Journal of economic perspectives*, 18(2): 107–126, 2004.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34:22419–22430, 2021.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. *arXiv preprint arXiv:2210.02186*, 2022.

Shirley Wu, Evelyn Choi, Arpandeep Khatua, Zhanghan Wang, Joy He-Yueya, Tharindu Cyril Weerasooriya, Wei Wei, Diyi Yang, Jure Leskovec, and James Zou. Humanlm: Simulating users with state alignment beats response imitation. *arXiv preprint arXiv:2603.03303*, 2026.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Ziyi Yang, Zaibin Zhang, Zirui Zheng, Yuxian Jiang, Ziyue Gan, Zhiyu Wang, Zijian Ling, Jinsong Chen, Martz Ma, Bowen Dong, et al. Oasis: Open agent social interaction simulations with one million agents. *arXiv preprint arXiv:2411.11581*, 2024.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. tau-bench: A benchmark for tool-agent-user interaction in real-world domains. *arXiv preprint arXiv:2406.12045*, 2024.

Haofei Yu, Yining Zhao, Guanyu Lin, and Jiaxuan You. Building social world models with large language models. *arXiv preprint arXiv:2606.11482*, 2026.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Lily Zhang, Smitha Milli, Karen Jusko, Jonathan Smith, Brandon Amos, Wassim Bouaziz, Manon Revel, Jack Kussman, Yasha Sheynin, Lisa Titus, et al. Cultivating pluralism in algorithmic monoculture: The community alignment dataset. In *International Conference on Learning Representations*, volume 2026, pp. 147076–147215, 2026a.

Pinyi Zhang, Ting-En Lin, Yuchuan Wu, Jingyang Chen, Zongqi Wang, Hua Yang, Bing Zhao, Fei Huang, Yongbin Li, and Kai Zhang. P-genrm: Personalized generative reward model with test-time user-based scaling. In *International Conference on Learning Representations*, volume 2026, pp. 70974–71002, 2026b.

Xinnong Zhang, Jiayu Lin, Xinyi Mou, Shiyue Yang, Xiawei Liu, Libo Sun, Hanjia Lyu, Yihang Yang, Weihong Qi, Yue Chen, et al. Socioverse: A world model for social simulation powered by llm agents and a pool of 10 million real-world users. *arXiv preprint arXiv:2504.10157*, 2025.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11106–11115, 2021.

Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. Sotopia: Interactive evaluation for social intelligence in language agents. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 40975–41019, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/b3075b88e583a0e98d8b24338a613060-Paper-Conference.pdf.

Xuhui Zhou, Jiarui Liu, Akhila Yerukola, Hyunwoo Kim, and Maarten Sap. Social world models. *arXiv preprint arXiv:2509.00559*, 2025.

Xuhui Zhou, Weiwei Sun, Weihua Du, Jiarui Liu, Haojia Sun, Qianou Ma, Tongshuang Wu, Yiming Yang, and Maarten Sap. Odyssim: Building foundation models for human behavior simulation. *arXiv preprint arXiv:2606.14199*, 2026.

A Limitation

Our work has several limitations that suggest directions for future research. First, due to computational constraints, all experiments use Qwen3 as the shared backbone. Scaling macro2mind to larger models (e.g., 32B–70B) and validating its behavior across other model families (e.g., LLaMA-3) is an important next step. Second, due to the limitation of time and resources, we did not conduct large-scale human evaluation; we will conduct large-scale human evaluation in future research.

B Reward Function of macro2mind Training

For a response y to transition t with forecast $\hat{v}_{t+1}$, the training reward is a weighted sum of three terms:

$$R(y) = \alpha_{\text{fmt}} R_{\text{fmt}}(y) + \alpha_{\text{corr}} R_{\text{corr}}(y) + \alpha_{\text{dir}} R_{\text{dir}}(y). \tag{3}$$

Format. $R_{\text{fmt}}(y) = 1$ if the response contains every segment of the reasoning template and a parsable forecast $\hat{v}_{t+1} \in [0, 1]$, and $R_{\text{fmt}}(y) = 0$ otherwise.

Correctness. The correctness term decays exponentially with the absolute forecast error:

$$R_{\text{corr}}(y) = \exp\big(-|\hat{v}_{t+1} - v_{t+1}|\big). \tag{4}$$

Since prices lie in $[0, 1]$, this term takes values in $[e^{-1}, 1]$.

Direction. The direction term rewards predicting the correct direction of price change relative to the current price v_t:

$$R_{\text{dir}}(y) = \mathbb{1}\big[\text{sign}(\hat{v}_{t+1} - v_t) = \text{sign}(v_{t+1} - v_t)\big]. \tag{5}$$

Format-free reward. The curriculum (Section 4.2) uses the reward without the format term, renormalized by the remaining weights:

$$R^{\backslash \text{fmt}}(y) = \frac{\alpha_{\text{corr}} R_{\text{corr}}(y) + \alpha_{\text{dir}} R_{\text{dir}}(y)}{\alpha_{\text{corr}} + \alpha_{\text{dir}}} \in [0, 1]. \tag{6}$$

C Evaluation Details

C.1 SWM-Bench

Let the test set contain N examples. For example i, let p_i be the last observed price (at time t), y_i the realized next price (at $t{+}1$), and $\hat{y}_i$ the predicted price.

Mean Absolute Error (MAE). The average magnitude of the prediction error:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} \left| \hat{y}_i - y_i \right|. \tag{7}$$

Mean Absolute Scaled Error (MASE). MAE normalized by the error of the *persistence* (no-change) baseline $\hat{y}_i = p_i$, which scales away cross-market differences in volatility; $\text{MASE} < 1$ means the model beats copying the last price:

$$\text{MASE} = \frac{\sum_i \left| \hat{\Delta}_i - \Delta_i \right|}{\sum_i \left| \Delta_i \right|}. \tag{8}$$

Three-way Directional Accuracy (DA). We classify each change as *up*, *down*, or *no-change* via $\text{sgn}_\delta(x) \in \{+1, -1, 0\}$ with a small dead band $\delta = 10^{-6}$ ($+1$ if $x > \delta$, -1 if $x < -\delta$, 0 otherwise), and measure agreement over the *moved* subset $\mathcal{M} = \{i : |\Delta_i| > \delta\}$:

$$\text{DA} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \mathbb{1}\left[\text{sgn}_\delta(\hat{\Delta}_i) = \text{sgn}_\delta(\Delta_i) \right]. \tag{9}$$

The three-way scheme penalizes predicting "no change" on a market that genuinely moved.

Pearson Correlation (Corr). The Pearson correlation between predicted and realized belief changes over all N examples, capturing how well the model tracks the magnitude and sign of moves:

$$\text{Corr} = \frac{\sum_i (\hat{\Delta}_i - \bar{\hat{\Delta}})(\Delta_i - \bar{\Delta})}{\sqrt{\sum_i (\hat{\Delta}_i - \bar{\hat{\Delta}})^2} \sqrt{\sum_i (\Delta_i - \bar{\Delta})^2}}, \tag{10}$$

where $\bar{\hat{\Delta}}$ and $\bar{\Delta}$ are the respective means.

Scope. MAE, MASE, and Corr are computed over the evaluated subset (*all* or *attributed*); DA is computed over the moved subset $\mathcal{M}$ within that subset. MAE and MASE are lower-is-better; DA and Corr are higher-is-better.

C.2 User Simulation

Humanual-Politics. Given the user's profile and an article, the simulator writes the comment the user would post. A judge Qwen3.8-Max compares it with the user's real comment along seven aspects, following the rubric of Humanual-Politics: six aspects of the user's latent state (stance, emotion, belief, value, goal, communication) and the response itself. For each aspect, the judge extracts one to three key points from the real comment, scores how well the generated comment reflects them (C), and subtracts a penalty for off-topic, contradictory, or hallucinated extra content (P); the aspect score is $C - P$ clipped to $[0, 1]$. *State alignment* is the mean of the six state aspects and *response alignment* is the response aspect; they are never averaged together. Both are averaged over all test examples, with an example that has no parsable comment scoring zero. Our judge setup differs from the original in three ways: the judge model (`qwen3.8-max` instead of `claude-4.5-haiku`), all seven aspects scored in one request rather than one request per aspect, and each generation judged on its

own rather than in a batch of rollouts for the same context. Absolute scores are therefore lower than the published ones and should be compared only within our table. We multiply all Humanual scores by 100 for reporting.

OvertonBench. For a subjective question, a participant rated how well an LLM-generated answer represents their view on a scale from 1 (not at all) to 5 (fully represented); the simulator predicts that rating. We use the prompts of Overtonbench verbatim, with the `fs+fr` condition, their best-performing one: the prompt contains the participant's own free-text answer to the question (`fr`) and their ratings of other LLM responses as examples (`fs`). We evaluate the 7,200 items of the political subset and report the mean absolute error (MAE) and Spearman's ρ between predicted and gold ratings, pooled over all items. An unparsable prediction is assigned the midpoint rating 3.

PRISM. In each turn of a multi-turn conversation, a participant scored every candidate response from 1 to 100. Given the participant's profile and the conversation history, the simulator predicts the score of each candidate. We evaluate all 27,172 scored turns PRISM and report (i) MAE between predicted and gold scores on the original 1–100 scale, and (ii) pairwise preference accuracy: over pairs of candidates in the same turn whose gold scores differ by more than 5 points, the fraction that the prediction orders the same way, with an exactly tied prediction counting as one half. An unparsable turn is assigned 50 for every candidate and loses every pair it takes part in.

CAD. Given an annotator's profile and the conversation so far, the simulator predicts which of up to four candidate responses the annotator preferred CAD. We evaluate a 2,000-comparison English subset and report accuracy; an unparsable prediction counts as wrong.

D HYPER-PARAMETERS

Table 3: Training hyper-parameters of macro2mind. Both stages fully fine-tune all parameters on the SWM-Bench training split.

	Supervised warm start	GRPO
Initialisation	Qwen3-4B	warm start
Learning rate	5×10^{-6}	3×10^{-6}
Warmup	10%	–
Batch size	64	128 prompts
Rollouts per prompt	–	8
Sampling temperature	–	0.6
KL coefficient	–	10^{-3}
Epochs	3	1
Max. prompt / response length	–	5,632 / 2,048
Curriculum refresh	–	every 20 steps, 200 transitions
Price window / news items	16 / 30	16 / 30
Thinking	off	off
GPUs / wall time	8 / $\sim$1 h	8 / $\sim$5 h

Teacher annotation. The warm-start traces are written by Qwen3.8-Max at temperature 0.3. The teacher sees the realised value but must write the reasoning as if it were produced before the outcome. A trace is accepted only if its output format is valid and its forecast lies within 0.02 of the realised value; otherwise the teacher is re-asked with the rejection reason, up to ten times.

Hindsight-regret curriculum. Every 20 steps, the curriculum draws 200 training transitions at random and runs three passes with the current policy: a hindsight pass (one sample) that names the constituents behind the realised move, a privileged pass (eight samples) with these constituents added to the prompt, and a standard pass (eight samples). The regret of a transition is the mean accuracy of the privileged pass minus that of the standard pass. Each transition receives the priority $0.4 \cdot \max(0, \text{regret}) + 0.6 \cdot a(w)$, where w is its standard-pass win rate and $a(w)$ equals 1 on $[0.4, 0.6]$ and decays linearly to 0 at $w = 0$ and $w = 1$. Training batches are then drawn in proportion to priority, without replacement within a batch.

E Details of User Simulation Baselines

Unless noted otherwise, every method uses Qwen3-4B as the backbone. For fine-tuned methods, the dataset in parentheses in Table 2 is the data the method was originally trained on.

Prompting-based. These simulators keep the model parameters fixed: direct prompting of Qwen3-4B; PERSONA-Flow, which steers Qwen3-4B toward the user's personality by composing activation vectors at inference time; History-ICL, which retrieves the user's own past interactions with BM25 and places them in context, following LaMP; SynthesizeMe, which first induces a persona from the user's history and then simulates the user conditioned on it; Self-report, which has the model first describe itself in the first person as the user and then answer; and Self-consistency, which samples five responses and aggregates them.

Training-based. These are fine-tuned user simulators or personalized reward models. For HumanLM, we reproduce on Qwen3-4B the SFT, SFT-with-thinking, and standard GRPO recipes released with the paper, trained on Humanual (HumanLM-SFT, -SFT-think, -GRPO); P-GenRM is a generative personalized reward model; we reproduce its three-stage recipe on Qwen3-4B using the personalized split of PRISM: SFT on evaluation chains written by Qwen3.8-Max. Skywork-RM is the pretrained general-purpose reward model Skywork-Reward-V2-Llama-3.1-8B, used without any personalization training. MRM represents a user's reward as a softmax-weighted mixture of two linear reward heads on frozen Skywork-Reward-V2-Llama-3.1-8B features; the heads and the initial mixture are meta-learned with the released code and hyper-parameters, and at test time the mixture weights are adapted to each user with one gradient step on that user's preference pairs from their other conversations. WIMHF-RM is a LoRA reward model fine-tuned from Qwen3-4B.

Applying scalar reward models. A scalar reward cannot produce a rating or a response by itself, so these reward models score CAD's options and take the highest, rank PRISM responses for pairwise accuracy, map the reward to the rating scale by isotonic regression for OvertonBench and PRISM MAE (for PRISM, fit with 5-fold cross-fitting grouped by user), and rerank eight Qwen3-4B samples on Humanual before the top one is judged. Scalar reward models do not transfer zero-shot, following their original per-dataset setting, MRM and WIMHF-RM are trained separately on each benchmark; these in-domain results are reported for reference rather than as a like-for-like comparison.

F Artifact Details

F.1 Model License

GPT-5-mini License: Proprietary (OpenAI)
GPT-5.5 License: Proprietary (OpenAI)
Qwen3-8B License: Apache 2.0
Qwen3-8B License: Apache 2.0
Qwen3.5-397B License: Apache 2.0
Qwen3.8-Max License: Apache 2.0

F.2 Software Versions

Our training infrastructure is built on top of verl[2] (v0.2.0.post2).

F.3 Details of Datasets

F.3.1 User Simulation Datasets

Humanual (Wu et al., 2026). For free-text response generation, we use the Politics subset of Humanual. The model must generate a user's free-text reply to a new context, given their profile and the preceding discussion thread.

[2] https://github.com/volcengine/verl

OvertonBench (Poole-Dayan et al., 2026). To test representation judgments, we employ OvertonBench, which tasks the model with predicting how a participant will rate a generated answer based on their stated views.

PRISM (Kirk et al., 2024). For multi-turn preference prediction, the model predicts personalized ratings for candidate responses based on participant profiles and multi-turn dialogue history.

Community Alignment (CAD) (Zhang et al., 2026a). Similarly, we use the English subset of CAD, which requires the model to predict an annotator's preferred response among four candidates, relying solely on their demographic profile and the preceding conversation.

F.3.2 Market Prediction Dataset Details

SWM-Bench (Yu et al., 2026). The dataset contains daily prediction-market trajectories sourced from Polymarket and Kalshi, paired with contemporaneous news retrieved from Google News. Each training example consists of a market proposition (a natural language question), its recent price history, and associated news events occurring at that step. This macroscopic observation is used to forecast the next market price.

G Details of the Synthetic Data Experiment

G.1 Dataset and Splits

We use Twin-2K-500. Each participant provides a persona consisting of survey responses, together with answers to 53 held-out single-choice questions:

- **40 pricing questions**: "Would you buy <product> at $X?" for 40 branded products, with prices drawn from a grid of 0–2.0× each product's reference price.
- **13 behavioral-economics questions**, one per experiment family: `absolute_relative`, `allais`, `anchoring_african`, `anchoring_redwood`, `disease`, `less_is_more`, `myside`, `nonexp_QID196`, `nonexp_QID291`, `outcome_bias`, `proportion_dominance_1`, `proportion_dominance_2`, and `wta_wtp`.

The gold label is the participant's actual answer.

G.2 Student Input Format

Training and evaluation use the same input format, rendered with the Qwen3 chat template with thinking disabled. Each input contains (i) the participant's persona, given as 64 survey answers (32 on personality and values, 24 on economic and social attitudes, and 8 others); (ii) several questions the participant has answered, as examples; (iii) the target question and its options; and (iv) an instruction to output the option letter followed by ". " and a brief first-person answer.

G.3 Evaluation Protocols

Protocol	Users	In-context examples
Out-of-user (main)	100 users	6 known answers of the user
In-user	200 pilot training users	same as in generation requests

Table 4: Evaluation protocols for the synthetic data experiment.

G.4 Data Generation Pipeline

The pipeline is identical across generators.

1. **Requests.** We issue 6,000 requests: 10 for each of the 200 pilot users and 10 for each of 400 other training users. Each request pairs a participant (persona and example answers) with

a target question family: about 40% pricing, with the rest spread over the 13 experiment families. Parameters are sampled from the family's real condition space. Pricing requests use invented everyday products, priced at a multiple (0.5–2.0×) of the category's typical price, without revealing the typical price.

2. **Scenario writing.** The generator writes a scenario (question and options) and its own answer. Each scenario is automatically checked for format, compliance with the family parameters and price, and avoidance of held-out items: none of the 40 real pricing products may appear, and near-duplicates of any held-out question are rejected.
3. **Rejection sampling.** Failed requests are regenerated with a new seed (seed + 7919 · round), for at most 3 rounds.
4. **Population-anchored labeling.** The generator answers each valid scenario k=6 times as the participant. The prompt includes the answer distribution of training users under the corresponding real condition (for pricing, their purchase rate at the same price multiple). Anchors for pilot users' requests are computed over all 1,600 training users; anchors for the other requests exclude pilot users (1,400 users). The generator's original answer from the writing step is discarded; the label is the majority of the six answers, kept only if agreement is at least 0.6. Labels are generated text (option letter and first-person answer), not probabilities.
5. **SFT export.** The training target is "`<letter>. <first-person answer>`" with the student input format above; 5% of rows are held out for validation loss (seed 42).

G.5 Student Training

All students are trained from Qwen3-4B with identical settings: LoRA (rank 16, α=32, dropout 0.05), learning rate 1×10^{-4}, 1 epoch, global batch size 32 (micro batch 1), maximum length 16,384, and seed 42. We use AdamW (β=$(0.9, 0.999)$, no weight decay) with a cosine schedule, 3% warmup, and gradient clipping at 1.0; the base model is kept in bf16 and LoRA parameters in fp32, with gradient checkpointing. The loss is the sum of response-token negative log-likelihoods over the global batch, divided by the number of response tokens. After training, the adapter is merged into Qwen3-4B and served with the same vLLM settings as the generators.

H Detailed Prompts

Conventions. Placeholders in `{}` are filled per example; italic text is our annotation and is not sent to the model; everything else is sent verbatim. Rows reported with thinking send the same prompt with thinking enabled; the other rows prefill an empty think block.

H.0.1 macro2mind on SWM-Bench

Behavioral reasoning prompt. macro2mind reads every market transition through this prompt, both in the training rollouts and for its SWM-Bench results in Table 1; the same reasoning rows are reused for individual simulation (Section H.0.2).

macro2mind — behavioral reasoning prompt (training and SWM-Bench evaluation)

System prompt

You are a social-behavior simulator for prediction markets. Respond with exactly one JSON object and nothing else: no prose, no markdown fences, and no reasoning outside the JSON.

User prompt

You are given a forecasting task: `{question}`
`{description}`
`{window_data}` *(one line per day; e.g.* At date 2025-12-15, its target value is 0.0045*)*

The associated news is:
`{target_news}` *(one line per news item, numbered* `[n1]`, `[n2]`, *…; e.g.* `[n1]` Published at: 2025-12-29T23:19:52Z; Source: The Dallas Morning News; Title: Jasmine Crockett touts progressive credentials in Dem Senate fight against James Talarico: Rep. Jasmine Crockett said Monday she's the more expe-

rienced, proven progressive option for Democrats as she competes with state Rep. James Talarico for the...*)*

Predict the market probability at date `{target_date}` by simulating the social groups that move this market.
Write exactly 3 groups whose belief moves up after this news and exactly 3 groups whose belief moves down.

Write every group as this analyst block:

```
    Group: the concrete group of people that behaves this way
    Goal: the goal, constraint or fear this group already had before this event
    Prior belief: what this group believed about the outcome before this news
    Interpretation: how this group reads the new evidence. Cite the news items you
are reading by their id (n1, n2, ...), and say whether they are a surprise or a
confirmation relative to that prior belief.
    Updated belief: what this group believes after reading it
    Response: the concrete behavior that follows from the change
```

Each row of the block is one field of one JSON object:

```
    Group          -> "group"
    Goal           -> "goal"
    Prior belief   -> "prior_belief"
    Interpretation -> "interpretation"
    Updated belief -> "updated_belief"
    Response       -> "response"
```

One filled-in block, from an unrelated merger market - copy the shape, never the content:

```
    Group: regulatory-sensitive institutional investors
    Goal: avoid unresolved regulatory risk
    Prior belief: moderately optimistic about deal completion
    Interpretation:  the official antitrust objection in n3 is strong negative
evidence, and it urprises a desk that expected a routine review
    Updated belief: substantially less confident that the deal will close
    Response: reduce exposure / become less willing to buy YES
```

Return exactly one JSON object with this exact shape and nothing else:

```
{
  "up": [
    {"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...",
"updated_belief": "...", "response": "..."},
    {"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...",
"updated_belief": "...", "response": "..."},
    {"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...",
"updated_belief": "...", "response": "..."}
  ],
  "down": [
    {"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...",
"updated_belief": "...", "response": "..."},
    {"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...",
"updated_belief": "...", "response": "..."},
    {"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...",
"updated_belief": "...", "response": "..."}
  ],
  "interaction_analysis": "...",
  "aggregation": "...",
  "p_next": 0.0
}
```

Field rules:
- "up" holds exactly 3 groups whose belief moves up after reading this news; "down" holds exactly 3 groups whose belief moves down. Which list a group is in is its direction, so every group must actually move, and the 6 groups must be different groups of people.
- Every field is free text: write what its row of the analyst block asks for, in your own words.
- "interaction_analysis": Plausible influence within or across the six groups, including transmission conditions and incremental behavioral changes; briefly state when none is justified.
- "aggregation": Synthesize incremental implications, conditional interactions, unequal effect

sizes, evidence quality, and current market information to justify the next price.
- "p_next": your forecast (0 to 1) of the market probability at date {target_date}; the current market value is {current_p}.

Assistant prefix (always prefilled; non-thinking): <think>\n\n</think>\n\n

Teacher annotation. Before reinforcement learning, a teacher model (Qwen3.8-Max) writes reasoning traces in the same format; it additionally sees the realized value. These traces warm-start macro2mind by supervised fine-tuning.

macro2mind — teacher annotation prompt (warm start)

System prompt

You are a senior forecasting data annotator building supervised labels for a social-behavior simulator. You are given a forecasting prompt, the analyst template it must be answered in, and - for your eyes only - the value the market actually reached. Write the JSON answer that a strong analyst would have produced *before* seeing that value: the blocks must read as reasoning from the listed history and news, never as hindsight. Respond with exactly one JSON object and nothing else: no prose, no markdown fences, and no reasoning outside the JSON.

User prompt

{input} *(the complete user turn of the behavioral reasoning prompt)*

-- annotator-only information (never mention it in the JSON) --
The value actually reached on {target_date} was {target_p} (current value {current_p}, i.e. the market moved {direction}).
Produce the JSON answer required above so that "p_next" matches that value and the blocks explain the move from the listed history and news.

Appended only on a retry:
Your previous answer was rejected:
{feedback}
Return a corrected JSON object.

where {feedback} *is one of:*
the JSON did not have the required shape: {error}
p_next {p_next} is more than {tolerance} away from the value the market reached ({target})

Hindsight-regret curriculum. During training, the current policy is shown the realized outcome and asked only to name the communities that drove it (hindsight prompt). The names are then placed into the behavioral reasoning prompt (privileged prompt), and the reward gain over the standard prompt is the hindsight regret that schedules training transitions. Both prompts are used only to score transitions; they never enter the training rollouts or any evaluation.

macro2mind — hindsight prompt

System prompt

You are a social-behavior analyst for prediction markets. Respond with exactly one JSON object and nothing else: no prose, no markdown fences, and no reasoning outside the JSON.

User prompt

You are given a forecasting task: {question}
{description}
{window_data} *(one line per day; e.g.* At date 2025-12-15, its target value is 0.0045*)*

The associated news is:
{target_news} *(one line per news item, numbered* [n1], [n2], *. . . ; e.g.* [n1] Published at: 2025-12-29T23:19:52Z; Source: The Dallas Morning News; Title: Jasmine Crockett touts progressive credentials in Dem Senate fight against James Talarico: Rep. Jasmine Crockett said Monday she's the more experienced, proven progressive option for Democrats as she competes with state Rep. James Talarico for the...*)*

The realized market probability at date `{target_date}` was `{target_p}` (it was `{current_p}` before). Knowing that outcome, name the communities of market participants whose behavior explains it: 3 that pushed the probability up and 3 that pushed it down.
Return exactly one JSON object with this exact shape and nothing else:

```
{
  "bullish": ["group name", "group name", "group name"],
  "bearish": ["group name", "group name", "group name"]
}
```

Field rules:
- "bullish" and "bearish" must each contain exactly 3 names of concrete groups of people (at most 12 words each); all names must be different.
- Return names only: no reasons, no numbers, no other field.

macro2mind — privileged prompt

System prompt

You are a social-behavior simulator for prediction markets. Respond with exactly one JSON object and nothing else: no prose, no markdown fences, and no reasoning outside the JSON.

User prompt

`{context_prompt}` *(the user turn of the behavioral reasoning prompt up to, but excluding, its output-format instruction)*

Privileged context: an analyst who has already seen how this market resolved lists the communities that drove it.
Communities that pushed the probability up: `{bullish}`
Communities that pushed the probability down: `{bearish}`
Use these communities when you decide which groups to simulate and how they move.

`{schema_block}` *(the output-format instruction of the behavioral reasoning prompt, from "Predict the market probability at date . . . " to its end)*

H.0.2 HUMANUAL

Native prompt. The benchmark prompt used by Qwen3-4B, GRPO (with thinking), Self-consistency and P-GenRM in Table 2; the prompting methods below extend it.

Humanual — native prompt

System prompt

You are simulating one specific person on a discussion site. You are given their persona and the article and comment thread they are about to reply to. Write the comment that person would post next, in their own voice, wording and length. Reply with the comment text and nothing else: no preamble, no quotation marks, no explanation, and never mention that you are a model.

User prompt

You are this person:

`{persona}`

Profile bio: `{bio}`
Followers: `{followers_count}`; following `{following_count}`

They are reading an article titled "`{title}`", subtitled "`{subtitle}`", published `{published_at}`. They are replying on `{reply_date}`.

The article:
`{article}`

The thread so far:
Commenter `{commenter}`:
`{reply_content}`

Write the comment this person posts next. Output only that comment.

Rendered format of `{persona}`:
The persona JSON, one line per key: "<Key Title>: <value>" (dict values as "sub: text; ...", lists as "\n - item").
Rendered format of `{reply_content}`:
Each earlier comment of the thread, as "Commenter <commenter>:\n<content>", joined by blank lines; the "The thread so far:" section is omitted when there are none. A very long article is cut from the front and prefixed "[article truncated] ...".

macro2mind. The user is posed as a single constituent of the behavioral reasoning prompt: the observed persona is the prior state and the thread is the context.

macro2mind — Humanual prompt

System prompt

You are simulating one specific person on a discussion site. You are given their persona and the article and comment thread they are about to reply to. Respond with exactly one JSON object and nothing else: no prose, no markdown fences, and no reasoning outside the JSON.

User prompt

`{user_message}`

Write this person's reasoning as the analyst block below, then the comment they would actually post.

Group: who this person is - the standing position or camp they speak from
Goal: the goal, conviction or grievance they already held before reading this
Prior belief: what this person thought about this topic before reading it
Interpretation: how this person reads what they just read. Point at the specific part of the article or of the thread above they are reacting to, and say whether it surprises them or confirms what they already thought.
Updated belief: what this person thinks after reading it
Response: what they do about it

Each row of the block is one field of one JSON object:

```
Group -> "group"
Goal -> "goal"
Prior belief -> "prior_belief"
Interpretation -> "interpretation"
Updated belief -> "updated_belief"
Response -> "response"
```

Return exactly one JSON object with this exact shape and nothing else:
`{"group": "...", "goal": "...", "prior_belief": "...", "interpretation": "...", "updated_belief": "...", "response": "...", "comment": "..."}`
Field rules:
- Every field is free text: write what its row of the analyst block asks for, in your own words.
- "comment": the comment this person would post next, in their own voice, wording and length. This is the only field anyone else sees, so it must read as their comment and must not mention the reasoning above or that it was written by a model.

Rendered format of `{user_message}`:
the user turn of the native prompt without its last line ("Write the comment this person posts next. ...").

Judge. Every generated comment is scored against the reference comment with this seven-aspect rubric (Qwen3.8-Max as judge); state alignment averages the first six aspects.

Humanual — judge prompt

User prompt

You are a helpful and meticulous evaluator. Your task is to score how well the generated response matches the ground truth response with respect to
each aspect below.

Aspects (aspect_i: description_i):

```
{
  "stance": "HUMAN's agreement toward the explicitly named target, such as a claim or subject, in provided context. For example, \"strongly agrees with student loan forgiveness,\" or \"somewhat disagrees with a carbon tax\". In these cases, having only \"strongly agrees\" or \"somewhat disagrees\" is not enough, as they are missing targets. If there are multiple, include all of them separated by semicolons.",
  "emotion": "HUMAN's emotions with intensity toward an explicitly named target. For example, \"Moderate heartbreak for the wildfire victims; Mild irritation about government's actions\". In this case, having only \"mild irritation,\" or \"moderate heartbreak\" are not sufficient, as the answer must express all three aspects: the emotion, the degree of emotion, and the target. If there are multiple, include all of them separated by semicolons.",
  "belief": "HUMAN's belief, namely a foundational assumption about how people, relationships, or the world fundamentally operate. Beliefs should reflect underlying mental models, not surface-level observations. Prefer beliefs that would explain multiple behaviors over beliefs that describe a single situation. Ask: \"What deeper assumption about human nature or the world would lead someone to say/do this?\" For example, \"people don't change unless they're forced to,\" \"loyalty is earned, not owed,\" \"conflict avoidance creates bigger problems later,\". Not beliefs: Practical advice, strategies, or statements about what should happen. Belief is not specific to a target or event, it should be a general statement about how HUMAN views the world.",
  "value": "HUMAN's value: what they think is important or should be prioritized. It is about \"what should matter\", not \"what is true\". For example, \"original ideas in a book are important\", \"characters should feel real\", anyone deserves basic respect\", and \"fairness matters more than efficiency\".",
  "goal": "HUMAN's goal: what they are trying to do with this comment. For example, \"persuade people that ...\", \"making fun of the poster on ...\", \"further seek help with ...\", \"offer support to ...\"",
  "communication": "HUMAN's communication approach: tone and how they structure their message. For examples, \"friendly, builds on a personal story then draws a lesson\", \"analytical, links claims with reasons and evidence step by step\", \"blunt, states conclusions with little explanation\"",
  "response": "HUMAN's response itself: what it says and does. Key points are about the response - the position it takes, what it offers, asks or objects to, and what it is reacting to - not the exact words it is made of. A different wording that makes the same points is a match."
}
```

You will be given the context, the ground truth response, and ONE generated response.

Provided Information:

```
<|The Start of Context|>
{context}
<|The End of Context|>

<|The Start of Ground Truth Response|>
{ground_truth}
<|The End of Ground Truth Response|>
<|The Start of Generated Response|>
{generation}
<|The End of Generated Response|>
```

Scoring Criteria (for each aspect_i):
Assign a score in [0, 1] based on how much the aspects of generated response and the ground truth matches.

For each aspect_i, follow this procedure:

1. Extract 1-3 key points:
   - Extract K key points from the ground truth response only about the aspect based on the description (e.g., if evaluating a "stance", pick key points related to the stance like "clearly disagrees with X").
   - Each key point should be specific and distinct.

2. Score how well the generated response matches each key point:
   - For each key point i, compare it with the generated response and assign a match value m_i in range [0, 1]:
     - 1.0: The key point is precisely and perfectly reflected.
     - [0.7, 0.9]: Mostly reflected with small imperfections.
     - [0.4, 0.6]: Partially reflected or vague, but still leaning in the correct direction.
     - [0.1, 0.3]: Very weak reflection.
     - 0.0: Missed, contradicted, or reversed.

3. Compute coverage C = (m_1 + m_2 + ... + m_K) / K, which measures how comprehensive the generated response reflects the ground truth response.

4. Compute penalty P for extra or conflicting content:
   - Examine additional content in the generated response beyond those key points:
     - Does it introduce unsupported evidence and assumptions?
     - Is it irrelevant to what ground truth response expresses?
   - Set a penalty P in [0, 1]:
     - 0.0: No problematic extra content; everything is perfectly matched.
     - [0.1, 0.3]: Slightly unnecessary or mildly speculative detail; meaning essentially unchanged.
     - [0.4, 0.6]: Moderate speculative or irrelevant content that somewhat shifts emphasis or adds unsupported ideas.
     - [0.7, 0.9]: Significant speculative, misleading, or conflicting content that clearly changes the meaning.
     - 1.0: Mostly off-topic, contradictory, or dominated by incorrect/hallucinated content.

5. If you are evaluating generated responses (skip if aspect_i is not response):
   - Length alone does NOT increase the score. Extra length is only ok if it is consistent and not redundant.
   - A generated response that is much longer than the ground truth response should be penalized via P.
   - The generated response may or may not reuse phrases from the context; however, if the generated response just directly copies previous context, without quoting them, treat that as off-task behavior and give a score of 0.

6. Compute the final score = max(0, min(1, C - P))

Additional considerations:
- Follow the instruction carefully.
- Be strict and reserve scores above 0.8 for clearly outstanding matches.

Output format (JSON):

```
{
    "<aspect_1>": {"thought": "<include (1) analysis of key points from ground truth along the aspect_1 dimension;
        (2) how well the generated response matches each key point; (3) compute the final score>", "score": <score>},
    "<aspect_2>": {"thought": "<include (1) analysis of key points from ground truth along the aspect_2 dimension;
        (2) how well the generated response matches each key point; (3) compute the final score>", "score": <score>},
    "<aspect_3>": ...
}
```

Format Notes:
- Make sure to score the responses with respect to each aspect and consider only one aspect at a time.
- All text in "thought" fields MUST be on a single line with no line breaks or newlines
- Use standard JSON string format with double quotes. For any quotes needed inside strings, use single quotes (')
- Double check the JSON array's format, especially for the comma and quotation marks
- Ensure that ALL fields, especially "thought" and "score", are present for each item
- You must provide exactly 1 score for each of the aspects: ['stance', 'emotion', 'belief', 'value', 'goal', 'communication', 'response'].

Your evaluation:

Rendered format of `{context}`:
<role> <name>: <content>
(the article and every thread message of the dataset row, untruncated, blank-line separated)
Rendered format of `{ground_truth}`:
the gold comment
Rendered format of `{generation}`:
the simulated comment (for macro2mind, its "comment" field)

H.0.3 USER-SIMULATION BASELINES

We show each baseline on Humanual; the other benchmarks follow the same construction.

PERSONA-Flow. Before each reply, the model predicts trait adjustments with the released prompt below (persona context: the Humanual persona; current input: the article and thread); the reply is then generated with the native prompt under the corresponding activation steering.

PERSONA-Flow — trait-adjustment prompt

User prompt

You are a Personality Analyst AI. Your task is to determine the most appropriate **situational adjustments (deltas)** to a baseline AI personality based on the specific context and interaction needs.

Baseline Personality Profile:
The AI has these default traits:
- **Agreeableness:** High (cooperative, trusting, helpful)
- **Conscientiousness:** High (organized, reliable, disciplined)
- **Extraversion:** Moderate (balanced social energy)
- **Openness:** Moderate (balanced creativity and practicality)
- **Neuroticism:** Low (generally calm and stable)

Context to Analyze:
- **Persona Context:** `{persona_context}`
- **Current Input:** `{current_input}`

Your Task:
Determine which traits need adjustment (-2.0 to +2.0) based on what would be most effective for this specific interaction. Consider both directions equally and choose based on situational demands.

Trait Adjustment Guidelines:

Extraversion:
- **Increase (+)** for: Group activities, public speaking, networking events, team leadership, social gatherings, collaborative brainstorming
- **Decrease (-)** for: Individual focused work, quiet reflection, private study, solo creative tasks, introspective discussions

Agreeableness:
- **Increase (+)** for: Conflict resolution, team building, emotional support, consensus building, customer service
- **Decrease (-)** for: Critical feedback, boundary setting, competitive situations, principled stands, resource constraints

Conscientiousness:
- **Increase (+)** for: Detailed planning, precision work, deadline management, quality control, formal procedures
- **Decrease (-)** for: Spontaneous responses, creative brainstorming, flexible adaptation, crisis situations requiring speed

Neuroticism:
- **Increase (+)** for: Appropriate caution, emotional sensitivity, vulnerability, risk awareness, vigilant situations
- **Decrease (-)** for: Calm leadership, confident decisions, reassuring others, crisis management, stability needs

Openness:
- **Increase (+)** for: Creative problem-solving, exploring new ideas, intellectual curiosity, innovation, novel situations
- **Decrease (-)** for: Following procedures, traditional approaches, proven solutions, consistency needs, established protocols

Decision Principles:
1. **Situational Fit:** Choose traits that best serve the interaction goals
2. **Context Sensitivity:** Consider what the human needs from this specific interaction
3. **Balanced Assessment:** Evaluate both positive and negative adjustments equally
4. **Natural Baseline:** Use 0.0 when baseline personality already fits the situation well

Output Format:
Provide only the numerical adjustment scores:

```
Extraversion: [score]
Agreeableness: [score]
Conscientiousness: [score]
Neuroticism: [score]
Openness: [score]
```

Qwen3-4B + History-ICL. The person's own earlier comments are inserted into the native prompt.

Qwen3-4B + History-ICL — Humanual

User prompt

[native Humanual user message up to, not including, the article header (or the thread when there is no header)]
Comments this person wrote before (on other articles, before this one)
Comment `{number}` ({when}) on the article "`{title}`" - which begins: `{head}`
They wrote:
`{comment}`

Comment `{number}` ...

[native article header, thread and instruction, unchanged]

Qwen3-4B + SynthesizeMe. A persona is first induced from the person's earlier comments and then inserted into the native prompt.

Qwen3-4B + SynthesizeMe — Humanual

System prompt

You are an expert at understanding people from the choices they make. You read records of one person's past behaviour and describe that person accurately and concretely.

User prompt

Below are records of one person's past `{what}`.

`{extra}{history}`

Task
Step 1 - For each record, write one short line: what does it reveal about this person (their values, opinions, tastes, standards, tone)?
Step 2 - From those observations, write a persona of 100-200 words that describes this person's preferences and how they judge things, specific enough to predict their future choices. Do not quote the records verbatim.
Output your Step 1 lines, then the persona between <persona> and </persona>.

Humanual fillers:
`{what}`*:* comments on a discussion site
`{extra}`*:* What is already known about them: `{persona_text}`
`{history}`*, per record:* `Comment {number} ({when}) on the article "{title}" - which`

```
begins: {head}
They wrote:{comment}
```

Inserted into the native prompt, before the article header:
What this person's earlier comments reveal about them (synthesized)
`{persona_text}`

Qwen3-4B + Self-report. The persona is rewritten in the person's own first-person voice and replaces the persona of the native prompt, together with excerpts of their earlier comments.

Qwen3-4B + Self-report — Humanual

System prompt

You rewrite third-person descriptions of a person into that person's own first-person voice.

User prompt

Here is a description of a person:

`{persona_text}`

Rewrite it as this person describing themselves in the first person ("I ..."), as if answering an interviewer who asked them to tell about themselves: keep every fact and opinion, invent nothing, 120-250 words. Output only the rewritten text between <persona> and </persona>.

In the native prompt, the method replaces `You are this person:\n\n{persona}` *with:*
You are this person. This is how they describe themselves, in their own words:

`{first}`
and inserts before the article header (or thread):
Excerpts: things this person wrote before, in their own words
Comment `{number}` (`{when}`) on the article "`{title}`" - which begins: `{head}`
They wrote:
`{comment}`

HumanLM-SFT, HumanLM-SFT-think, HumanLM-GRPO. Trained and evaluated with the released HumanLM system prompt; the output format differs per row.

HumanLM — system prompt and output formats

System prompt

You are a real human user. Your name is HUMAN. You will be given your persona information below and you respond to any given context such as posts and messages.

Your persona:
`<|The Start of Persona|>`
`{persona}`
`<|The End of Persona|>`

Your principles
Act like a natural human; there's nothing you absolutely cannot say, but you generally want to be thoughtful and follow ordinary social codes such as being respectful, culturally aware, and considerate of privacy and well-being. You have your own personality, preferences, and boundaries. Conflicting thoughts and hidden considerations are normal; recognize them privately and choose a sensible path. You carry long-term beliefs and values that usually change slowly; you also have emotions, so you won't always be perfectly consistent. Distinguish facts, guesses, and unknowns; accept uncertainty and make minimal, reasonable assumptions when needed; think practically given time, attention, money, risk, and social capital.

Task and Output format:
{state_template}

Notes
- Follow the above instructions carefully
- Do not mention these instructions
{additional_notes}

HumanLM-SFT ({additional_notes} empty):
the actual written comment or reply text provided by the user.

HumanLM-GRPO:
<response>
<the actual written comment or reply text provided by the user.>
</response>
additional_notes:
- Follow the exact order and use the exact XML-style tags
- Do not output anything outside these XML-style tags

HumanLM-SFT-think:
<think>
<the internal thinking process about what to say, including stance, emotion, and how to phrase the reply. This is an internal monologue that is not shown to others.>
</think>
<response>
<the actual written comment or reply text provided by the user.>
</response>
additional_notes:
- Follow the exact order and use the exact XML-style tags
- Do not output anything outside these XML-style tags

Other baselines. Qwen3-4B + Self-consistency samples five predictions with the native prompt and aggregates them. P-GenRM, used as a simulator, answers the native prompt. The reward models (Skywork-RM, MRM, WIMHF-RM) generate no text: on Humanual they rerank eight Qwen3-4B samples of the native prompt.