

LatentDemandBench: A Contamination-Resistant Diagnostic for Pre-Recognition Product–Need Ranking in Synthetic Economies

Avinash Goutham Aluguvelly
The University of Texas at Austin
algo@utexas.edu

Abstract

An existing product may meet an unmet need before the market recognises the connection. Testing whether a language model can identify such opportunities is difficult when historical outcomes may already be in its training data. We introduce LatentDemandBench, a contamination-resistant benchmark that generates fictional economies with known realisation probabilities and noisy documentary evidence. Solvers rank product–need pairings by their chance of realisation within a fixed horizon. Market commentary is generated separately from realisation, allowing us to evaluate ranking both among discussed pairings and where market attention supplies no ordering. On 30 matched economies, two frontier models beat a mention-count control but trail a supervised logistic ranker using fourteen features recovered from the same dossiers by a template-aware parser. The stronger model, GPT-5.6-TERRA, scores 0.669 in top- K -imputed AUC, with a paired reference-minus-model gap of +0.047 [+0.025, +0.074]. Aggregating three completions under the same final top- K budget yields 0.659, leaving a reference gap of +0.058 [+0.031, +0.095]. The deficit is not uniform: an exploratory analysis finds it smaller on unmentioned pairings than on those already discussed. We release the generator, harness and artifacts to support evaluation on fresh economies.

1 Introduction

Graphics processors built for rendering found another use in training neural networks: AlexNet was trained on two GPUs in 2012 (Krizhevsky et al., 2012). The forecasting question is whether that fit between an existing capability and a new need could have been identified before it became widely recognised. Asking a model trained on later accounts cannot cleanly test that judgement: it may be recalling the outcome rather than inferring it from the evidence available at the time. We operationalise a narrower prerequisite: composing

noisy capability evidence, applying a conjunctive readiness gate, and ranking product–need pairs in fictional economies.¹

Recent work measures this contamination. It estimates a model’s propensity to recall a firm-date outcome and shows that its apparent forecasting skill concentrates exactly where that propensity is high (Gao et al., 2025), standardises the measurement of look-ahead bias in point-in-time financial models (Benhenda, 2026), and shows that reported gains from LLM trading strategies shrink under longer horizons and wider cross-sections (Li et al., 2026a). Chronologically restricted point-in-time models exclude future training data by construction (Kelly et al., 2026), while inference-time methods seek to mitigate look-ahead bias without retraining (Li et al., 2026b). For a temporally unrestricted frontier model evaluated retrospectively, however, leakage cannot be fully excluded.

We take the other route. If the world is generated rather than observed, the ground truth is a known quantity instead of an inferred one, no entity in it has ever been written about, and the market’s recognition of an opportunity can be generated separately from whether the opportunity is real. That separation is what makes the task well posed. In a synthetic economy we can ask, and score independently, the four questions that financial practice runs together: will this pairing happen, is it worth anything, does the originator keep the value, and has anyone noticed.

Our main reference is a supervised logistic model over fourteen features recoverable from the dossiers, fitted on economies disjoint from evaluation. It has labelled training data and, in its parsed-text form, a parser written for the document templates; §3.7 and §5 assess these advantages; this is not a matched zero-shot comparison.

¹Code and artifacts: <https://github.com/tokenledger/latentdemandbench>.

Our contributions:

- A generator for synthetic economies with exact realisation, materiality, capture and recognition probabilities, rendered into a document corpus in five lexical surfaces, with a lossy observation layer so that nothing a solver reads is the latent state (§3).
- An evaluation protocol that holds every ranker to the same top- K contract, with training-size and input-recovery checks for the supervised reference and its parsed-text pipeline (§3.7).
- Results on two frontier models and one open-weight model, including a coupled problem-size stress test, an assist ablation that localises the bottleneck at low capability, a counterfactual twin design that separates detecting an intervention from ranking well, and an untruncated probability-quality analysis (§5).

We are explicit about what the design cannot show. Every entity is fictional, so the benchmark cannot test whether apparent forecasting ability on real markets reflects pretrained recall. It can show only that the task is hard where entity-level and outcome-level recall are unavailable. Nor is that property permanent for a fixed instance: once released, the seeds in this paper could enter a later training corpus, which is why the generator is the artifact and evaluation on fresh seeds is the recommended protocol.

2 Related Work

Financial NLP benchmarks. Financial benchmarks have grown from single tasks (Xu and Cohen, 2018; Chen et al., 2021, 2022) to suites spanning extraction, question answering, risk and trading (Shah et al., 2022; Xie et al., 2023, 2024; Peng et al., 2025), alongside domain-pretrained models (Wu et al., 2023) and document-grounded question answering (Islam et al., 2023). Almost all of them score either comprehension of a text that already contains the answer, or return prediction, where the label is noisy and the signal is thin (Lopez-Lira and Tang, 2023). Neither isolates the stage we care about, which is whether a capability implies an application before financial discourse takes notice. The recognition gap is a first-class quantity here rather than a confound, and it is measurable because we generate commentary separately from outcomes.

Contamination and look-ahead bias. Contamination is now measured rather than assumed (Sainz et al., 2023; Golchin and Surdeanu, 2024), and the standard remedies are to keep the test set moving (White et al., 2025; Karger et al., 2025) or to build items from knowledge postdating training (Wu et al., 2025). Finance has its own variant: parametric look-ahead bias, where the outcome sits in the weights rather than in the prompt (Gao et al., 2025; Benhenda, 2026). Input-side controls such as point-in-time corpora and disciplined backtesting (Li et al., 2026a) do not reach it. Generated worlds do, at the cost of external validity, which we treat as the central limitation rather than as a footnote.

The closest recent work. Three lines come near this design without meeting it. Matlin et al. (2026) generate financial question-answer items semi-synthetically, under expert control, from a curated corpus of authoritative documents. The items describe the real economy and their answers are grounded in text that exists, so prior exposure to the sources cannot be ruled out; our entities exist nowhere. Ong et al. (2025) ask models to generate forward counterfactuals, the opportunities and risks a financial headline implies, and score those generations against a rubric, because a real headline has no single realised future to check against. We fix each pairing’s realisation probability by construction, so a forecast is scored against a known generating distribution rather than judged for plausibility. Kelly et al. (2026) restrict the model instead of the world, training on chronologically filtered text and releasing dated checkpoints. That removes leakage for a model one trains oneself; a generated world removes it for any model, including one whose training corpus is unknown.

Two recent synthetic finance benchmarks share the diagnostic goal in other modalities, over private-equity spreadsheets (Ravnik et al., 2026) and over financial time series with known generating mechanisms (Sun et al., 2026). Both control one artefact type; we generate an economy as a document corpus.

Temporal graphs and technology forecasting. The formal analogue of our task is link forecasting on a temporal graph (García-Durán et al., 2018; Jin et al., 2020; Han et al., 2021; Liu et al., 2022), and in the scientific domain link prediction over a growing concept network has been used to forecast which research directions become active (Krenn et al., 2023). We inherit the framing, a path from

an entity through its capabilities to a need, and add two things those settings lack: exact edge probabilities rather than realised edges only, and a separately generated observer whose attention is not the label.

Counterfactual and synthetic evaluation.

Counterfactual variants of familiar tasks separate reasoning from recall (Wu et al., 2024), symbolic perturbations of benchmark items expose brittleness that aggregate accuracy hides (Mirzadeh et al., 2025), and causal benchmarks probe whether models track interventions rather than correlations (Kiciman et al., 2023; Jin et al., 2023). Our twin worlds are in this line: two economies identical under common random numbers except for a single complement’s availability date. General forecasting benchmarks (Zou et al., 2022; Halawi et al., 2024; Karger et al., 2025) pursue the same goal from the opposite direction, resolving real questions after the freeze; the price is that no ground-truth probability exists, only the outcome.

Finance itself supplies the motivation for treating recognition as a separate object. Information diffuses slowly across economically linked firms (Cohen and Frazzini, 2008), attention is a binding constraint on how quickly news is incorporated (Hirshleifer et al., 2009), and price can move on attention alone (Huberman and Regev, 2001). A benchmark that scores a model against what the market already says measures diffusion, not discovery.

3 The Benchmark

3.1 What the generated quantities stand for

The generator is an abstraction of pre-consensus opportunity screening, the work done in equity research, venture investing and corporate strategy before a theme is priced. Each latent quantity has a counterpart there. A capability profile is a firm’s technology, measured with error by outsiders; a need is an emerging industry demand, described by the bottleneck holding it back; complements are the approvals, infrastructure and supply conditions that gate deployment regardless of technical fit, and in the default configuration one blocks by having no announced availability rather than by being dated late, since every dated complement is drawn inside the horizon; materiality is the addressable earnings impact if the need is served; capture is whether the originating firm keeps that value or loses it to entrants; and recognition is analyst, media and investor attention, generated from firm size

and from what has already shipped, because that is how coverage accrues (Cohen and Frazzini, 2008; Hirshleifer et al., 2009).

We are not claiming these mechanisms reproduce a real economy. The claim is narrower: the decision has this structure, and the generator makes each part of it separable with an exact answer.

3.2 The generated economy

A universe is a latent economy in $\mathbb{R}^8$. Capabilities are sparse attribute vectors. A product owns one to three of them and its profile is their element-wise maximum, so a product’s capability is composed, never stated. A need specifies a required attribute profile, a sufficiency threshold, an economic materiality, and one to two complements with availability dates. For a product-need pairing, let m be the cosine alignment between the product profile and the need requirement, τ the need’s threshold, and a the logistic slope. Then

$$p_{\text{true}} = \sigma(a(m - \tau)) \cdot g,$$

where the gate g is 1 when every complement the need depends on is available inside the horizon and 0.03 otherwise. In the default generator every dated complement is inside the horizon, so $g = 0.03$ exactly when a required complement has no announced availability (NEVER). The residual is an internal activation propensity used in the initial Bernoulli draw and in commentary generation; it is not a positive probability of a blocked pairing arriving. An arrival must also have a non-empty feasible window. If a required complement is NEVER, that window is empty, the realised arrival is NEVER, and both the scored horizon probability p^* and the binary label are exactly zero, irrespective of the residual gate. For an unblocked pairing, p^* is the probability of arriving inside the horizon given that the pairing is not already live at the cutoff, obtained by spreading p_{true} uniformly over the feasible arrival window and conditioning on the pre-cutoff part not having happened. The two differ by up to 0.19 in a default universe. Realisation is drawn from a random stream private to that pairing, so an intervention on one pairing cannot perturb any other. Value capture is drawn separately from the company’s supply, distribution and defensibility, which makes it possible for a pairing to be realised while its originator keeps nothing. For realisation, the scoring targets are the conditional probability p^* and realised label y ; p_{true} feeds the internal

draw and commentary, not a metric. Table 17 distinguishes these quantities from the probability conditional on the dossier.

Market commentary is generated last and deliberately badly. A pairing is discussed because its company is large or because it has already been deployed, with only a weak dependence on the internal activation propensity. Commentary is therefore lagging and size-biased in the way real coverage is, and a ranker that reproduces it cannot thereby predict realisation.

3.3 What a solver sees

Everything a solver reads is a lossy rendering of that economy, specified in full in Table 8. Capability attributions are dropped and spuriously added, attribute readings carry instrument error, complement dates are misreported, and every figure is printed at two decimals, which is also the precision the structured references consume, so no reference reasons over numbers a reader cannot see. The corpus is a document set of roughly 3,000 words: capability characterisations, product teardowns, problem reports, readiness notes, commentary, and a list of pairings already deployed before the cutoff. Deployed pairings are disclosed and excluded from candidacy.

Figure 1 shows part of one dossier. The product teardown names capabilities, the characterisations measure them, and the problem report states what the need requires; the solver has to connect these records.

The same hidden economy renders into five surfaces: finance, opaque identifiers, space opera, anime-tech and cyberpunk. Sentence templates and statement counts are identical across surfaces and only the lexicon changes. Corpus length stays within 10.4% of the finance rendering in words, and within 0.3% for the three narrative surfaces; bare identifiers make the opaque rendering the long one. Prompt vocabulary is matched to the surface, so an opaque run does not reintroduce financial framing through the instructions.

3.4 The task

At the cutoff quarter, a solver returns two things. First, a ranking of at most $K=40$ pairings by probability of commercial realisation within the horizon, with a calibrated probability and the evidence path it composed for each. Second, answers for a probe set of sixteen pairings fixed in advance and stratified by (blocked, mentioned), giving for each

[TP-C7] Characterisation of Vexith-34. Measured attribute profile: coherence 0.14, latency 0.49, density 0.23, purity 0.62, stability 1.13. Observed downstream effect: Vexent response.

[PT-P6] Morarn-48 unit (Moroth Consortium). Teardown attributes capability: Vexith-34, Lunip, Kelax, Rethock. Issuer position: supply capacity 0.65, distribution reach 1.00, defensibility 0.56.

[PR-N0] Lunade shortfall. Governing bottleneck: Nalost constraint. Relief requires an attribute profile of thermal 0.41, yield 0.93, latency 0.43, density 0.17, throughput 0.69, stability 0.84, with a sufficiency threshold near 0.44 on profile alignment. Deployment additionally depends on: Lunade substrate, Vorarn substrate. Addressable value if served: 7.43 units.

[SR-K3] Lunade substrate: guided to quarter 21.

Figure 1: Dossier excerpt (seed 0, finance surface, cut-off quarter 20). A solver scores P6 with N0 by pooling the four attributed capability profiles, aligning the result with the printed requirement, and checking the threshold and complement dates. Rethock (C1) is a spurious attribution: the latent product has three capabilities. Latency and density are noise-added requirement coordinates. The observed alignment is 0.785 against the printed threshold of 0.44, and both reported complements arrive inside the horizon. The realised label is positive. No document states the link.

a probability, a materiality, a capture probability, whether the pairing is already discussed, and the ids of any complements still blocking it. The probe set is stratified because scoring auxiliary questions on whatever the model chose to rank is trivially gamed: a top-40 by realisation probability contains almost no blocked pairings, so answering “no blockers” every time scored a perfect 1.0.

A default universe has about 173 candidate pairings, of which 19.4% are realised within the horizon, 20.6% are blocked by a complement as reported, and 39.6% attract no commentary at all. That last subset is where the benchmark’s central question lives.

3.5 Metrics

The main outcomes are AUC_K and Gap-AUC_K . AUC_K assigns descending scores to returned pairings and ties every unreturned candidate at score 0 before computing AUC over the realised labels in each universe. Gap-AUC_K uses the same scores but restricts evaluation to pairings with no commentary. The other ranking, value, probability and probe metrics are defined in Appendix B, with results in Tables 3, 6 and 13.

Confidence intervals are percentile bootstrap over universes, and every model-to-reference comparison is paired on the universes the model actually ran on, because between-universe difficulty is the dominant variance component. The primary comparisons are each model against the negative control and against the supervised reference, on the default finance universes. Every other arm is an exploratory diagnostic and its intervals are unadjusted for multiplicity.

3.6 Scoring protocol

Every ranker is held to one contract: an ordered list of at most K pairings, $K=40$ by default and changed only in the coupled problem-size stress test. Only valid, unique entries survive. Realisation metrics use rank position, and everything outside the shortlist scores zero. Auxiliary metrics use either this validated shortlist or the fixed probe set, as specified in Appendix B.

3.7 Rankers

Every ranker is held to that same contract. Scoring a reference on all 173 candidates while a model returns at most 40 makes AUC partly a measure of truncation: under the shared protocol the oracle scores 0.830 rather than 0.909. It is Bayes-optimal in expectation under the generating process, not a hard bound on any finite draw of realised outcomes. The tables use the short labels below.

The *Mention-count* control ranks by mention count; it is an attention heuristic, not a recognition or market-consensus estimate. *Simulator-form* applies the simulator’s own functional form, which no document states, and is therefore privileged. *Expected value* ranks by probability times materiality times capture, and is privileged in the same way. *Oracle* ranks by exact p^* . It is Bayes-optimal in expectation under the generating process, not a hard ceiling on any finite draw, and it is not an achievable target.

The reference that carries the argument is none of these. The *Supervised* reference is a logistic model over fourteen observable features and an intercept, listed in full with their document sources in Table 11: overlap between the composed profile and the requirement, covered and unmet requirement mass, shared attribute count, the stated threshold and the margin against it, whether every complement lands inside the horizon, the shortest and longest complement wait, counts of complements and capabilities, stated materiality,

a formula-derived capture proxy, and the mention count. How they combine for realisation is what the fit has to discover. One feature is not innocent: the formula-derived capture proxy applies the simulator’s capture formula to three printed issuer-position numbers, so it is partially privileged. It also contributes almost nothing: removing it changes AUC_K by -0.0002 , and removing it together with commentary volume by -0.0026 . Every row labelled “Supervised” here uses the same fit: seeds 1000–1099 (100 economies, 17,257 labelled pairings). Default-finance baselines are evaluated on seeds 0–99 and model arms on 0–29; all model-to-reference contrasts use that matched subset. Fits on fewer universes appear in Figure 3 and Table 12. Knowing the generative form buys little over learning it: the simulator-form ranker beats it by $+0.006$ $[-0.002, +0.013]$ AUC_K over 100 universes.

Parsed-text pipeline. The supervised reference receives structured inputs rather than prose. To check whether its inputs are recoverable from the documents, the *parsed-text pipeline* reads the corpus and resolves entities through the same bracket tags available to a solver. A deterministic parser written for the templates recovers all 10 field types at 1.000 fidelity (Table 14). The resulting feature matrix is identical to the structured one; fitting the same logistic model reproduces 0.724 AUC_K , including on the opaque surface (0.724). This establishes recoverability, not an independent performance result. The pipeline remains supervised and template-aware, and says nothing about the difficulty of extracting equivalent features from real financial prose.

4 Experimental Setup

Universes derive from fixed integer stream identifiers, never Python’s salted hash, so a corpus is byte-identical across processes and machines. Each result artifact records source provenance. Main-arm generation hashes were captured before execution and checked at completion. The reported twin-arm generation hashes were captured only at write time and cannot establish source stability during generation. The released runners enforce both checks. Offline rescoring retains generation provenance separately from the scoring revision, and table generation checks the latter. Model calls that fail for account or quota reasons are recorded as failures and never scored as chance-level predictions, and an artifact with any unresolved call is

refused.

Frontier arms use direct, tool-free OpenAI Responses API calls with strict JSON-schema output, reasoning effort `medium`, and a 48,000-token output budget. The full system prompt, task template, surface substitutions and response schema are in Appendix E. An agentic CLI harness was available and is excluded from every reported row: read-only sandboxing prevents writes, not reads, and a run with filesystem access can read the generator, the response cache and other surfaces’ outputs.

GPT-5.6-LUNA, GPT-5.6-TERRA and GPT-5.6-SOL are the requested model identifiers sent to the API and recorded in the artifacts, not pseudonyms or provider-returned immutable revision IDs. Each ran at reasoning effort `medium`. The open-weight arm is QWEN/QWEN3-32B at bfloat16 on two H100s under vLLM 0.10.1.1, tensor-parallel 2, temperature 0, seed 0, guided JSON decoding, `max_model_len` 32768 and a 24000-token output budget, with the reasoning trace disabled.

We evaluate GPT-5.6-LUNA and GPT-5.6-TERRA on 30 universes per arm, against baselines over 100, in sixteen arms each: five surfaces, three assists, two horizons, two threshold slopes, two economy sizes, a three-replicate repeat and a twin-world run. We add one open-weight model, QWEN3-32B, run on the finance surface as a single batch job on H100:2 GPUs (14 minutes wall clock) with guided JSON decoding against the same schema, temperature 0 and without a reasoning configuration. A fourth model, GPT-5.6-SOL, used a different harness revision. Appendix C reports its results with a matched-model comparison across harness revisions.

5 Results

Where commentary is silent. On the zero-mention subset the mention-count control is tied at 0.500 by definition. The oracle scores 0.818 and the supervised reference 0.698, against 0.830 and 0.724 over all candidates. Since 39.6% of candidates attract no commentary, this shows that the generated evidence supports ranking where existing attention supplies no ordering; it does not establish performance against a real market.

Single completions exceed a negative control and trail a supervised reference. Both frontier models rank above the mention-count control (+0.072 [+0.051, +0.093] and +0.134 [+0.103, +0.167] AUC_K , paired) and below the supervised reference in

Ranker	n	AUC_K	Gap- AUC_K
Random	100	0.506 [0.496, 0.516]	0.509 [0.493, 0.526]
QWEN3-32B	30	0.511 [0.500, 0.523]	0.509 [0.490, 0.530]
Mention-count	100	0.538 [0.529, 0.548]	0.500 [0.500, 0.500]
GPT-5.6-LUNA	30	0.607 [0.589, 0.625]	0.601 [0.569, 0.633]
Expected value [†]	100	0.669 [0.654, 0.685]	0.642 [0.619, 0.664]
GPT-5.6-TERRA	30	0.669 [0.645, 0.695]	0.695 [0.661, 0.728]
Supervised	100	0.724 [0.709, 0.738]	0.698 [0.680, 0.718]
Simulator-form [†]	100	0.729 [0.716, 0.744]	0.715 [0.694, 0.735]
Oracle [†]	100	0.830 [0.817, 0.844]	0.818 [0.799, 0.837]

Table 1: Ranking on the finance surface, base rate 0.194, ordered by AUC_K . Baselines run on all 100 universes, model arms on the first 30. Brackets are 95% bootstrap CIs; rows marked [†] read quantities the prompt does not state. Gap- AUC_K restricts to pairings the market never mentions. The random control uses an RNG namespace independent of universe generation. The full metric set is Table 3; paired model-to-baseline comparisons on matched universes are Table 2.

single-completion runs (-0.110 [-0.135, -0.085] and -0.047 [-0.074, -0.025]). For scale, the simulator-form ranker reaches 0.729 against the latent oracle’s 0.830. Holding the scoring formula, candidate sets and top- K contract fixed, replacing the rendered inputs with latent inputs raises the reference to 0.825: an observation-layer loss of 0.096. Replacing that latent-input reference with the exact conditional-probability oracle contributes a further 0.005, for a total gap of 0.101. This is a paired decomposition along the specified sequence, not an estimate of a Bayes-optimal dossier reader. Table 1 gives the headline ranking metrics, with the full metric set and the paired contrasts in Appendix A. Two properties bound how much that reference is being helped. Its sample complexity is low: resampling the training set, five universes suffice for every one of 20 draws to beat all three single-completion model rows on the matched universes (paired median +0.033 against the strongest), while one universe suffices for only 26% of draws. Five universes also beat the replicated GPT-5.6-TERRA row in 100% of draws. Figure 3 and Table 12 give the distributions. And its advantage is not the one feature a solver is told to discount, since refitting without the mention count yields 0.721 AUC_K . It remains a supervised system reading engineered features, which the parsed-text pipeline of §3.7 is there to answer: that pipeline reads the documents and scores 0.724, so both models also fall short of a mechanical reader of the same rendered evidence under different supervision and template assumptions.

Open-weight output compliance. QWEN3-32B scores 0.511 AUC_K , indistinguishable from random (-0.008 [-0.030, +0.014]) and significantly below the commentary ranker (-0.024 [-0.045, -0.003]). Of 34.5 returned pairings on average, only 22.6 survive validation (Table 7). Its shortfall thus mixes ranking with output compliance, and its no-reasoning setting was imposed for schema compatibility. This is a result about one constrained configuration, not the model family.

The stronger model’s deficit appears on mentioned pairings, not on silent ones. We do not detect a GPT-5.6-TERRA deficit on zero-mention pairings (-0.007 [-0.038, +0.024]), while on the discussed subset there is one (-0.062 [-0.100, -0.031]), and a direct contrast estimates its deficit to be +0.055 [+0.009, +0.100] smaller on the silent pairings. That is a relative localisation, not equivalence with the reference where the market is quiet. It is also specific to this endpoint: GPT-5.6-LUNA trails on both subsets with no detectable asymmetry (+0.014 [-0.035, +0.062]), as does QWEN3-32B (+0.018 [-0.017, +0.052]).

Replication does not remove the fixed-budget deficit. Averaging three completions per universe, by averaging the rank-position scores each completion assigns to a pairing and imputing zero where a completion omitted it, then retaining only the final top K , yields GPT-5.6-TERRA 0.659 and GPT-5.6-LUNA 0.628 (paired changes of -0.011 [-0.025, +0.002] and +0.021 [+0.006, +0.038]). Their shortfalls against the reference remain +0.058 [+0.031, +0.095] and +0.089 [+0.066, +0.112], respectively. Average-score ties are preserved; candidate order resolves ties at the cutoff. The union of several shortlists must not be scored as an expanded output budget: that would confound replication with coverage. These three-completion results do not eliminate the reference advantage; they do not establish what a different aggregation rule or a larger sampling budget could achieve. Every other arm is single-completion, apart from the twin arms below.

Coupled problem-size stress test. The three configurations contain 8/10/25, 12/15/40 and 18/22/90 products/needs/ K . Averaged over the 30 model universes, they yield 76/173/379 candidate pairings, 1,676/2,927/5,384 corpus words, and top- K coverage 0.33/0.23/0.24. Entity counts, candidates, context length and K therefore co-vary; this stress test does not identify a causal effect of any

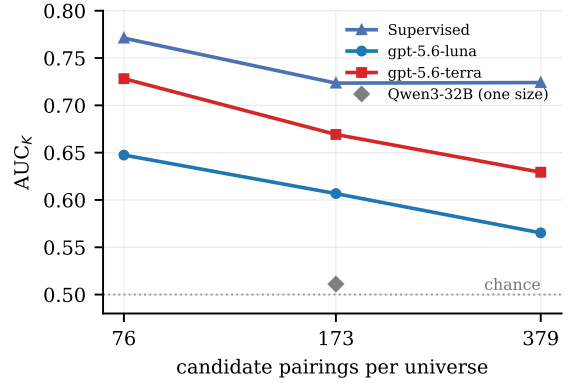


Figure 2: Coupled problem-size stress test. Each point jointly changes products, needs, candidate count, corpus length and K ; it does not isolate a causal size effect. QWEN3-32B ran at one configuration, so it is a marker, not a trend.

one component. Reference-minus-model deficits for GPT-5.6-TERRA are +0.052 [+0.025, +0.081]/+0.047 [+0.025, +0.074]/+0.091 [+0.069, +0.115], and for GPT-5.6-LUNA +0.133 [+0.101, +0.165]/+0.110 [+0.085, +0.135]/+0.155 [+0.133, +0.179] (Figure 2). At the largest configuration GPT-5.6-LUNA scores 0.565, while the reference averages 0.7241 over 100 universes. At $K = 90$, the models average 73.0/83.6 valid entries, respectively, so the comparison also includes output-compliance costs. Varying horizon and threshold slope produces no point-estimate ordering reversal (Table 16).

No lexical-surface effect is detectable. Across five renderings of the same economies, as finance, opaque identifiers, space opera, anime-tech and cyberpunk, we detect no AUC_K difference larger than 0.019 for either model (Table 4). Opaque-minus-finance point estimates are positive (+0.019 [-0.007, +0.045] and +0.008 [-0.011, +0.025], paired), but both intervals include small decreases. This does not establish invariance, since we did not prespecify an equivalence margin, and it is not a test of linguistic robustness, since templates and statement counts are fixed and only the vocabulary changes. What it tests is dependence on domain vocabulary, and we find no evidence of a material one within these fixed templates.

The composition assist helps the lower-capability model more. Supplying pre-computed profile alignments raises GPT-5.6-LUNA by +0.064 [+0.042, +0.088] and leaves GPT-5.6-TERRA where it was (+0.006 [-0.010, +0.021]; Table 5). Those are two separate intervals, so we test the difference

directly: the assisted gain is $+0.059$ $_{[+0.035, +0.083]}$ larger for the weaker model, and that interval excludes zero. This localises a difference between these two arms; it does not establish a general capability law. Supplying resolved complement readiness moves neither model ($+0.003$ $_{[-0.013, +0.021]}$ and -0.004 $_{[-0.025, +0.018]}$) and we find no evidence its effect differs between them ($+0.007$ $_{[-0.023, +0.038]}$). Supplying both helps both models but far more the weaker one ($+0.081$ $_{[+0.059, +0.104]}$ against $+0.013$ $_{[+0.001, +0.026]}$, difference $+0.068$ $_{[+0.043, +0.092]}$). The stronger model retains a shortfall after receiving both intermediate quantities, so neither assist closes it.

Twin worlds are underpowered at this budget.

In twin worlds differing only in one complement’s availability, with mention histories held to pre-intervention values and per-pairing random streams so nothing else moves, every uninformed ranker sits at exactly 0.500 by construction, so any movement is attributable to the intervention alone. The reference reaches 0.590 and the oracle 0.645, a score limited by the top- K contract because a pairing outside the list in both worlds cannot move. GPT-5.6-LUNA scores 0.603 $_{[0.517, 0.695]}$ and GPT-5.6-TERRA 0.562 $_{[0.459, 0.659]}$ on 15 twins each, too few to read an ordering into (Table 15).

The four elicited questions come apart. On the probe set (Table 6) models recover the latent value almost perfectly through its noisy printed version (materiality ρ of 0.964 and 0.965) and judge recognition well at higher capability (0.756 rising to 0.958), while remaining weak on the prerequisite question under the generator’s undated-only blocking regime (F1 0.560 and 0.572) and on who captures the value (MAE 0.122 and 0.080). Probability quality is measured here too, since nothing is truncated: Brier against the realised outcome puts both models near 0.127 against 0.093 for the reference, with the oracle’s 0.068 the irreducible outcome variance rather than zero. Table 13 gives that and the latent- p^* error. A model can be right about what an opportunity is worth and wrong about whether it is reachable, the conflation the benchmark was built to separate. Evidence paths cite dossier-attributed capabilities for 0.911/0.803 of top-ranked GPT-5.6-LUNA/GPT-5.6-TERRA pairs; false-capability citations occur in 0.003/0.000.

6 Discussion and conclusion

The parsed-text pipeline, using the supervised reference, outranks both frontier models on the base task. Under the same final output budget, three-completion GPT-5.6-TERRA scores 0.659, leaving a paired reference gap of $+0.058$ $_{[+0.031, +0.095]}$. The deficit is not uniform: an exploratory analysis detects a GPT-5.6-TERRA shortfall on discussed pairings but not on unmentioned ones. That distinction does not establish equivalence on the silent subset, and the supervised comparison does not isolate the cause of the overall gap. LatentDemandBench makes these distinctions testable by separating evidence of an opportunity from commentary about it. Its results concern ranking in noisy, generated economies, not broad commercial judgement. We release the generator, harness and artifacts (Appendix D).

Limitations

The economy is synthetic, and the reasoning it requires is profile alignment under noise plus a conjunctive availability gate. Strong performance here is not evidence of the analogical insight that motivates the problem, and the ordering of models on this benchmark should not be read as an ordering on real financial judgement.

Because all entities are fictional by construction, the benchmark cannot test the hypothesis that motivates it, namely whether apparent forecasting ability on real markets reflects pretrained recall. It establishes only that the task is hard where recall is unavailable. Testing the recall hypothesis requires a complementary real-market study.

No human baseline was collected. Two worked examples were hand-verified against the rendered dossier, confirming that the chain is recoverable from the text and that the reference’s arithmetic is correct, but that is a correctness check on the task rather than a measurement of human performance. Consequently we make no claim about models relative to people; the comparison is to a statistical reference reading the same evidence. Tooling for a human study is included in the release.

The supervised reference has labelled training data, which a solver reading one dossier does not. We bound that advantage rather than remove it. Five universes of labels suffice against every single-completion model on the base finance task in every resampled draw, and 100% of those draws beat the replicated arm. The commentary feature is not what carries it, and the parsed-text pipeline shows the inputs are recoverable from the documents. Refitting on 50 random single universes gives a median of 0.652 with a 5th to 95th percentile range of 0.482 to 0.697. Only 26% of single-universe fits beat all three model rows on matched universes, and a minority land below chance, because 173 labelled rows against fourteen features leave some fits degenerate. These results support the five-universe comparison, not a general claim about single-universe training. The reported reference was trained on about 17,257 labelled pairings; the models received no labelled training examples. The comparison therefore measures supervised, template-aware performance rather than a matched training regime. We did not run the information-matched alternatives suggested by this asymmetry: an in-context arm given the same five labelled universes, or an observable-only zero-shot

symbolic or calculator ranker with neither labels nor simulator-form knowledge. Those arms are required before attributing the remaining gap to arithmetic consistency, extraction or supervision.

The prompt tells a solver that a complement arriving after the horizon blocks realisation. In the evaluated latent economies that case is never generated, because every dated complement is drawn inside the horizon and blocking happens through undated ones; an after-horizon date can reach the observed dossier only through the misreport jitter. The instruction is therefore accurate about the scoring rule and misleading about the distribution, which slightly understates how easily the gate can be read. Accordingly, blocker F1 and the resolved-readiness assist are diagnostics for this undated-only regime, not clean estimates of reasoning about the general availability gate or late arrivals.

The parser exploits known templates. It shows that the evidence in these documents supports the ranking; it says nothing about extraction difficulty on real filings. Because recovery is exact, the parsed feature matrix is identical to the structured one and the two systems score identically; the pipeline is a demonstration that the inputs are recoverable, not an independent measurement.

Two further caveats on the analyses in §5. The mentioned-versus-silent localisation is exploratory and unadjusted for multiplicity: the direct within-model contrast puts GPT-5.6-TERRA’s deficit $+0.055$ [+0.009, +0.100] smaller on silent pairings, but the result is specific to one endpoint and does not establish equivalence with the reference there. And the open-weight arm’s near-chance score mixes ranking ability with compliance: it returns 22.6 valid pairings of a requested 40, so a model that merely filled the list would score higher without reasoning better.

Three models are a limited sample, including one open-weight model. The open-weight arm was run without a reasoning configuration; a free-form decode with a JSON extractor would measure something different and was not evaluated. It covers only the finance surface. Other proprietary backends are implemented but were not run to completion for budget reasons. The twin arms contain only 15 twins per model, and most arms are single-completion. The aggregation rule tested here retains a reference deficit; other aggregation rules and larger sampling budgets remain untested. The main-table ordering reflects point estimates.

Ethical Considerations

The benchmark contains no real companies, products, people or market data, and no personal or proprietary information. It measures a forecasting capability that has obvious commercial application, and we note that a system which ranked well here would still be untested on real markets, where the label is noisy, the observation process is adversarial and the cost of a confident wrong answer is borne by someone. We report a model’s failures on this task as failures on this task.

Acknowledgments

We thank the reviewers for their comments and suggestions. We used Modal’s cloud GPU infrastructure to run the QWEN3-32B experiments.

References

- Mostapha Benhenda. 2026. Look-ahead-bench: a standardized benchmark of look-ahead bias in point-in-time LLMs for finance. *arXiv preprint arXiv:2601.13770*.
- Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2021. FinQA: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.
- Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. 2022. ConFinQA: Exploring the chain of numerical reasoning in conversational finance question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.
- Lauren Cohen and Andrea Frazzini. 2008. Economic links and predictable returns. *The Journal of Finance*, 63(4):1977–2011.
- Zhenyu Gao, Wenxi Jiang, and Yutong Yan. 2025. Detecting lookahead bias in LLM forecasts. *arXiv preprint arXiv:2512.23847*.
- Alberto García-Durán, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.
- Shahriar Golchin and Mihai Surdeanu. 2024. Time travel in LLMs: Tracing data contamination in large language models. In *International Conference on Learning Representations*.
- Danny Halawi, Fred Zhang, Chen Yueh-Han, and Jacob Steinhardt. 2024. Approaching human-level forecasting with language models. In *Advances in Neural Information Processing Systems* 37.
- Zhen Han, Peng Chen, Yunpu Ma, and Volker Tresp. 2021. Explainable subgraph reasoning for forecasting on temporal knowledge graphs. In *International Conference on Learning Representations*.
- David Hirshleifer, Sonya Seongyeon Lim, and Siew Hong Teoh. 2009. Driven to distraction: Extraneous events and underreaction to earnings news. *The Journal of Finance*, 64(5):2289–2325.
- Gur Huberman and Tomer Regev. 2001. Contagious speculation and a cure for cancer: A nonevent that made stock prices soar. *The Journal of Finance*, 56(1):387–396.
- Pranab Islam, Anand Kannappan, Douwe Kiela, Rebecca Qian, Nino Scherrer, and Bertie Vidgen. 2023. FinanceBench: A new benchmark for financial question answering. *arXiv preprint arXiv:2311.11944*.
- Woojeong Jin, Meng Qu, Xisen Jin, and Xiang Ren. 2020. Recurrent event network: Autoregressive structure inference over temporal knowledge graphs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.
- Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. 2023. CLadder: Assessing causal reasoning in language models. In *Advances in Neural Information Processing Systems* 36.
- Ezra Karger, Houtan Bastani, Chen Yueh-Han, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E. Tetlock. 2025. ForecastBench: A dynamic benchmark of AI forecasting capabilities. In *International Conference on Learning Representations*.
- Bryan Kelly, Semyon Malamud, Johannes Schwab, and Teng Andrea Xu. 2026. Scaling point-in-time language models. *arXiv preprint arXiv:2607.11889*.
- Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. 2023. Causal reasoning and large language models: Opening a new frontier for causality. *arXiv preprint arXiv:2305.00050*.
- Mario Krenn, Lorenzo Buffoni, Bruno Coutinho, Sagi Eppel, Jacob Gates Foster, Andrew Gritsevskiy, Harlin Lee, Yichao Lu, João P. Moutinho, Nima Sanjabi, Rishi Sonthalia, Ngoc Mai Tran, Francisco Valente, Yangxinyu Xie, Rose Yu, and Michael Kopp. 2023. Forecasting the future of artificial intelligence with machine learning-based link prediction in an exponentially growing knowledge network. *Nature Machine Intelligence*, 5:1326–1335.

- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* 25.
- Weixian Waylon Li, Hyeonjun Kim, Mihai Cucuringu, and Tiejun Ma. 2026a. Can LLM-based financial investing strategies outperform the market in long run? In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.
- Weixian Waylon Li, Mengyu Wang, and Tiejun Ma. 2026b. Summoning the oracle to slay it: Mitigating look-ahead bias in financial backtesting with large language models. *arXiv preprint arXiv:2605.24564*.
- Yushan Liu, Yunpu Ma, Marcel Hildebrandt, Mitchell Joblin, and Volker Tresp. 2022. TLogic: Temporal logical rules for explainable link forecasting on temporal knowledge graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*.
- Alejandro Lopez-Lira and Yuehua Tang. 2023. Can ChatGPT forecast stock price movements? return predictability and large language models. *arXiv preprint arXiv:2304.07619*.
- Glenn Matlin, Akhil Theerthala, Anant Gupta, Anirudh JM, Rayan Castilla, Yi Mei Ng, and Sudheer Chava. 2026. FinForge: Semi-synthetic financial benchmark generation. *arXiv preprint arXiv:2601.06747*.
- Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2025. GSM-Symbolic: Understanding the limitations of mathematical reasoning in large language models. In *International Conference on Learning Representations*.
- Keane Ong, Rui Mao, Deeksha Varshney, Paul Pu Liang, Erik Cambria, and Gianmarco Mengaldo. 2025. Deriving strategic market insights with large language models: A benchmark for forward counterfactual generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*.
- Xueqing Peng, Lingfei Qian, Yan Wang, Ruoyu Xiang, Yueru He, Yang Ren, Mingyang Jiang, Jimin Huang, Qianqian Xie, Sophia Ananiadou, et al. 2025. MultiFinBen: Benchmarking large language models for multilingual and multimodal financial application. *arXiv preprint arXiv:2506.14028*.
- Jan Ravník, Matjaž Ličen, Felix Bühmann, Bithiah Yuan, Felix Stinson, and Tanvi Singh. 2026. FinSheet-Bench: From simple lookups to complex reasoning, where LLMs break on financial spreadsheets. *arXiv preprint arXiv:2603.07316*.
- Oscar Sainz, Jon Ander Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. 2023. NLP evaluation in trouble: On the need to measure LLM data contamination for each benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.
- Raj Sanjay Shah, Kunal Chawla, Dheeraj Eidnani, Agam Shah, Wendi Du, Sudheer Chava, Natraj Raman, Charese Smiley, Jiaao Chen, and Diyi Yang. 2022. When FLUE meets FLANG: Benchmarks and large pretrained language model for financial domain. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.
- Jiaze Sun, Kelvin J. L. Koa, Ruiyang Ni, Yize Liu, Haonan Chen, and Ke-Wei Huang. 2026. FinStressTS: A parametric synthetic benchmark for time-series forecasting in finance. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.
- Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Ben Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Siddhartha Naidu, Chinmay Hegde, Yann LeCun, Tom Goldstein, Willie Neiswanger, and Micah Goldblum. 2025. LiveBench: A challenging, contamination-limited LLM benchmark. In *International Conference on Learning Representations*.
- Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhajan Kambadur, David Rosenberg, and Gideon Mann. 2023. BloombergGPT: A large language model for finance. *arXiv preprint arXiv:2303.17564*.
- Xiaobao Wu, Liangming Pan, Yuxi Xie, Ruiwen Zhou, Shuai Zhao, Yubo Ma, Mingzhe Du, Rui Mao, Anh Tuan Luu, and William Yang Wang. 2025. AntiLeak-Bench: Preventing data contamination by automatically constructing benchmarks with updated real-world knowledge. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*.
- Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2024. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yanzhao He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, et al. 2024. The FinBen: An holistic financial benchmark for large language models. *arXiv preprint arXiv:2402.12659*.
- Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. 2023. PIXIU: A comprehensive benchmark, instruction dataset and large language model for finance. In *Advances in Neural Information Processing Systems 36, Datasets and Benchmarks Track*.
- Yumo Xu and Shay B. Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

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

A Full result tables

Every table below is generated from the released artifacts by the same script that generates the numbers quoted in the text.

B Scoring protocol in full

Every ranker is held to one contract. A ranker returns an ordered list of at most K pairings, $K=40$ by default and changed only in the coupled problem-size stress test. The first K valid, unique entries survive: duplicates keep their first position, and nonexistent or already-deployed pairings receive no score. Realisation metrics use rank position; value metrics instead use the separate probability-weighted orders, and Probe AUC uses the probe answers. Everything outside the shortlist scores 0, or a constant 0.02 where a probability is required. AUC is computed per universe with average ranks within ties, so a constant scorer sits at exactly 0.500, then macro-averaged over defined values. AUC subsets lacking either label class are undefined, not assigned chance.

Ranking and value. Recall@10 is the share of all realised pairings found in the first ten positions. nDCG@10 uses binary realisation gains, discounts position i by $1/\log_2(i+1)$, and normalises by the ideal top-ten gain. nDCG_{EV} uses realised materiality as gain instead. Its ordering is predicted probability times materiality, restricted to the validated shortlist; capture-AUC orders that shortlist by probability times capture and uses the realised-and-captured label. Rank-only controls have no probability-weighted value order, so these two metrics are undefined for them, not zero. Mentioned AUC_K restricts the main ranking to pairs with positive mention counts, complementing Gap-AUC_K. Consensus ρ is Spearman correlation between the ranking scores and mention counts. Ties in top-ten ordering use candidate order; Spearman uses average ranks.

Probability quality. The all-candidate *selection-and-probability error* is mean squared error against p^* after assigning probability 0.02 to unreturned candidates. It mixes selection with probability estimation: even the oracle has non-zero error because it returns only K pairs. We therefore also score the separately elicited probabilities on the fixed probes, without shortlist truncation. Brier score is squared error against the realised binary outcome and measures calibration and resolution together.

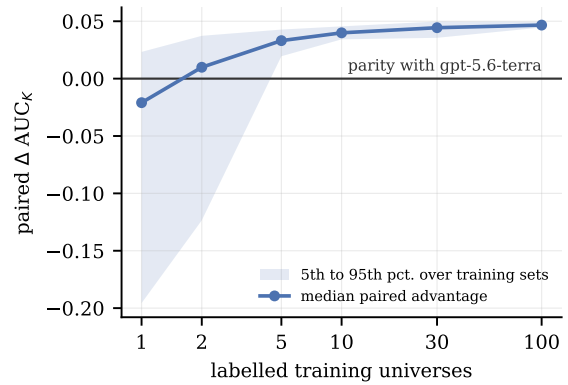


Figure 3: Paired advantage of the supervised reference over GPT-5.6-TERRA, on the 30 universes both were run on, against the number of labelled training universes. Points are the median across resampled training sets and the band the 5th to 95th percentile across those sets, uncertainty an evaluation bootstrap does not carry. One universe is unstable and usually loses (26% of draws beat all three single-completion rows); every one of the 20 five-universe fits wins.

Probe p^* MSE measures recovery of the hidden generating probability, not calibration conditional on the dossier: the best probability given a lossy dossier need not equal p^* . These two probe errors exclude missing answers. Probe AUC instead assigns zero to missing probabilities; single-class probe sets have undefined AUC. Probe coverage is 0.994/1.000 for GPT-5.6-LUNA/GPT-5.6-TERRA.

Elicited quantities and evidence paths. Materiality is scored by Spearman correlation with latent materiality, using average ranks for ties. Missing answers are excluded; constant predictions or targets give undefined correlation. Capture MAE compares the reported capture probability with the simulator’s probability. Recognition accuracy compares the reported flag with whether any commentary exists. Blocker F1 compares the reported complement ids with those the dossier shows as unavailable inside the horizon; two empty sets score 1. Missing probes contribute capture error 1, recognition accuracy 0 and blocker F1 0, including an entirely empty answer set. Each quantity is computed per universe, then averaged over defined values. The two path metrics use the validated ranked predictions, not the probes: they report the fraction of paths citing at least one dossier-attributed capability, and the fraction citing at least one capability not attributed to that product. They check attribution, not the full reasoning chain.

Model — reference	LUNA	TERRA	QWEN3-32B
Mention-count	+0.072 [+0.051, +0.093]	+0.134 [+0.103, +0.167]	-0.024 [-0.045, -0.003]
Supervised	-0.110 [-0.135, -0.085]	-0.047 [-0.074, -0.025]	-0.205 [-0.232, -0.181]
Simulator-form	-0.104 [-0.126, -0.081]	-0.041 [-0.057, -0.026]	-0.199 [-0.221, -0.178]
Oracle	-0.210 [-0.239, -0.181]	-0.147 [-0.179, -0.120]	-0.305 [-0.334, -0.278]

Table 2: Paired differences in AUC_K (model — reference) on the 30 universes each model was run on, with 95% bootstrap CIs over universes. Positive favours the model. Pairing removes between-universe difficulty, which is the dominant variance component.

Ranker	n	R@10	nDCG@10	nDCG _{EV}	AUC_K	Capture AUC	Gap- AUC_K	ρ_{cons}	Sel.+prob. MSE
Random	100	0.068	0.220	—	0.506 [0.496, 0.516]	—	0.509 [0.493, 0.526]	0.018	—
QWEN3-32B	30	0.065	0.241	0.199	0.511 [0.500, 0.523]	0.518	0.509 [0.490, 0.530]	0.047	0.106
Mention-count	100	0.086	0.286	—	0.538 [0.529, 0.548]	—	0.500 [0.500, 0.500]	0.755	—
GPT-5.6-LUNA	30	0.117	0.395	0.350	0.607 [0.589, 0.625]	0.589	0.601 [0.569, 0.633]	0.036	0.077
Expected value [†]	100	0.148	0.488	0.522	0.669 [0.654, 0.685]	0.679	0.642 [0.619, 0.664]	0.058	—
GPT-5.6-TERRA	30	0.141	0.458	0.391	0.669 [0.645, 0.695]	0.656	0.695 [0.661, 0.728]	0.047	0.070
Supervised	100	0.187	0.600	0.514	0.724 [0.709, 0.738]	0.702	0.698 [0.680, 0.718]	0.132	0.051
Simulator-form [†]	100	0.199	0.644	0.516	0.729 [0.716, 0.744]	0.708	0.715 [0.694, 0.735]	0.072	0.064
Oracle [†]	100	0.281	0.851	0.679	0.830 [0.817, 0.844]	0.796	0.818 [0.799, 0.837]	0.106	0.012

Table 3: Full metric set on the finance surface, base rate 0.194, ordered by AUC_K . Baselines run on all 100 universes and model arms on the first 30; n is per row. Brackets are 95% bootstrap CIs over universes. Gap- AUC_K is AUC_K restricted to pairings the market never mentions. Rows marked [†] are privileged: they read quantities the prompt does not state, so they bound the signal in the evidence rather than achievable performance. The random control uses an RNG namespace independent of universe generation. Dashes denote undefined quantities, not zero: rank-only controls have no probability-weighted value order or probability error, and the expected-value baseline does not emit a realisation probability. Sel.+prob. MSE is squared error against p^* over all candidates, which mixes truncation with probability error; the untruncated probability-quality figures are in Table 6.

Surface	LUNA	TERRA	Supervised
Finance	0.607	0.669	0.724
Opaque identifiers	0.626	0.677	0.724
Space opera	0.619	0.677	0.724
Anime-tech	0.618	0.661	0.724
Cyberpunk	0.622	0.668	0.724
Spread (max — min)	0.019	0.016	—

Table 4: The same hidden economies under five lexical surfaces: sentence templates and statement counts are identical and only the vocabulary changes. The supervised reference reads the recovered observables and is surface-invariant by construction, so it fixes the scale; model movement measures vocabulary sensitivity under fixed templates.

C A fourth model arm

GPT-5.6-SOL ran on the finance surface under a different harness revision and is excluded from every table. It scores 0.663 and is indistinguishable from GPT-5.6-TERRA on matched universes (-0.009 [-0.030, +0.013]); the same GPT-5.6-TERRA arm run under both revisions differs by -0.002 [-0.008, +0.003]. This matched-model comparison measures sensitivity to the harness revision; it does not make the fourth arm directly comparable to the main-table runs.

D Reproduction

Code and artifacts: <https://github.com/tokenledger/latentdemandbench>. The benchmark snapshot is commit [a36091a5957d](#), with prompt/response contract digest 99e2ea63074a3a11 (Appendix E).

The baseline table, reference twin results, sensitivity baselines, training-set resampling, parsed-text pipeline and probe analyses can be rebuilt without model calls. Model scores can be recomputed from archived responses. The provenance checks described in §4 hash the benchmark library and root runners; offline rescoring retains the generation record separately. The large-economy arms use a schema permitting 90 predictions. The table generator rejects scoring-hash mismatches, unresolved model calls and backends with filesystem access. Regression tests regenerate the tables and prose macros from the artifacts and compare them with the committed files. The random control and all paired comparisons use NumPy seed namespace 20260810, separate from universe generation.

For the frontier arms, artifacts retain the requested model identifier, reasoning effort and tool isolation, but not an API-returned immutable revision, request timestamp or wall time. Their

Evidence supplied	GPT-5.6-LUNA	Δ	GPT-5.6-TERRA	Δ
None (base task)	0.607 [0.589, 0.625]	—	0.669 [0.645, 0.695]	—
Composed alignments	0.671 [0.651, 0.691]	+0.064 [+0.042, +0.088]	0.675 [0.651, 0.698]	+0.006 [-0.010, +0.021]
Resolved readiness	0.610 [0.590, 0.631]	+0.003 [-0.013, +0.021]	0.665 [0.643, 0.687]	-0.004 [-0.025, +0.018]
Both	0.688 [0.667, 0.707]	+0.081 [+0.059, +0.104]	0.683 [0.657, 0.708]	+0.013 [+0.001, +0.026]

Table 5: Localising the bottleneck. Each row supplies one intermediate quantity the base task requires the model to compute: composed product profiles with their alignment to each need, resolved complement readiness, or both. Δ is paired against the same model’s base-task run.

Elicited quantity	LUNA	TERRA
Probe p^* MSE $\downarrow$	0.052 [0.042, 0.063]	0.044 [0.032, 0.056]
Materiality ρ $\uparrow$	0.964 [0.944, 0.981]	0.965 [0.944, 0.982]
Recognition acc. $\uparrow$	0.756 [0.700, 0.812]	0.958 [0.933, 0.981]
Blocker F1 $\uparrow$	0.560 [0.442, 0.674]	0.572 [0.485, 0.659]
Capture MAE $\downarrow$	0.122 [0.091, 0.156]	0.080 [0.066, 0.095]
Probe AUC $\uparrow$ ($n = 28$)	0.745 [0.677, 0.812]	0.792 [0.717, 0.864]
Path capability cited $\uparrow$	0.911 [0.843, 0.966]	0.803 [0.697, 0.895]
Path capability false $\downarrow$	0.003 [0.000, 0.007]	0.000 [0.000, 0.000]

Table 6: Finance-surface probe and path metrics. The first six rows use the fixed 16-pairing probe set; the last two use the validated ranked predictions, not probes. For Probe AUC, n counts universes whose probe set contains both label classes. The probe set is not truncated, so squared error against p^* here is free of truncation; omitted probe answers are excluded from that mean rather than penalised, which affects only GPT-5.6-LUNA (0.6% of probes). Materiality correlation also excludes missing answers. The references score 0.029 (supervised) and 0.044 (simulator-form).

temperature: 0.0 metadata does not describe a sent parameter: the Responses API request omits temperature. The open-weight arm instead records its complete vLLM decoding configuration, including temperature 0, seed 0 and wall time; its single offline batch on H100:2 GPUs took 14 minutes for 30 universes with no failed calls.

Quantity	LUNA	TERRA	QWEN3-32B
Requested K	40.00	40.00	40.00
Entries returned	39.13	39.50	34.53
Valid ranked entries	36.97	39.17	22.63
Duplicates dropped	0.07	0.00	8.53
Pairing does not exist	0.00	0.00	1.83
Disclosed as deployed	2.10	0.33	1.53
Fraction of K filled	0.92	0.98	0.57
Probe coverage	0.99	1.00	1.00

Table 7: What each endpoint returned against the top- K contract, recomputed from the archived responses rather than read off a stored count. Entries naming a pairing that does not exist, or one the dossier disclosed as already deployed, are dropped, as are repeats. The open-weight arm fills little more than half the list it was asked for, mostly through self-duplication, so its near-chance score is in part a list-completion result; the frontier arms fill 92% and 98%.

Quantity	Distribution or formula as coded	What the solver observes
<i>Defaults</i>		
Latent dimension	8 (module constant D)	attribute names only: 8 named channels per surface
Timeline length	40 quarters	not stated in the dossier
Never-activates sentinel	10^9	rendered as 'no announced availability'
Economy size	n_products=12, n_needs=15, n_caps=10, n_complements=8	12 teardowns, 15 problem reports, 10 characterisations, 8 readiness notes
Cutoff / horizon	cutoff=20, horizon=8	both stated verbatim in the prompt
Logistic slope	9.0	never stated; must be inferred
Ranking cap	TOP_K = 40	stated in the task text and enforced in scoring
<i>Capabilities</i>		
Capability attribute vector	$k \sim \mathcal{U}\{2, 3\}$ nonzero coordinates (rng.integers(2,4)), positions sampled without replacement from the 8; each nonzero $\sim \mathcal{U}(0.4, 1.0)$; all other coordinates exactly 0	$\text{round}_2(\max(0, v_c + \mathcal{N}(0, 0.18^2)))$ with values < 0.12 zeroed; printed as 'attr 0.xx' pairs
Capability / effect names	pseudo-words: onset x coda from two 20-item lists, plus a '-NN' suffix with prob. 0.4, $NN \sim \mathcal{U}\{2..89\}$; resampled until unique within the universe	the name string itself (bare id under the opaque surface)
<i>Products</i>		
Capability ownership	$ caps(p) \sim \mathcal{U}\{1, 2, 3\}$ (rng.integers(1,4)), sampled without replacement from the 10 capabilities	noisy claimed list: each owned capability kept with prob. $1 - 0.15$; each NON-owned capability added with prob. $0.10/10 \times 3 = 0.03$; if the list empties, the first owned capability is restored
Product profile	$\pi_p = \max_{c \in caps(p)} v_c$ (element-wise maximum)	never printed; the solver must compose it from the teardown plus the characterisations
Company size	$\exp(\mathcal{N}(0, 1^2))$ (lognormal, $\sigma = 1$)	NOT rendered anywhere; only its effect via mention counts is visible
Supply / distribution / moat	each i.i.d. $\mathcal{U}(0.1, 1.0)$	$\text{round}_2(\text{clip}(x + \mathcal{N}(0, 0.08^2), 0, 1))$, printed on a 0-1 scale in the teardown
<i>Needs</i>		
Requirement vector	$k \sim \mathcal{U}\{2, 3, 4\}$ nonzero coordinates (rng.integers(2,5)); each nonzero $\sim \mathcal{U}(0.4, 1.0)$	$\text{round}_2(\max(0, r_n + \mathcal{N}(0, 0.18^2)))$, values < 0.12 zeroed
Sufficiency threshold	$\mathcal{U}(0.45, 0.72)$	$\text{round}_2(\text{clip}(\tau_n + \mathcal{N}(0, 0.04^2), 0.2, 0.9))$, printed as 'sufficiency threshold near x.xx'
Materiality	$\exp(\mathcal{N}(0, 0.8^2))$ (lognormal, $\sigma = 0.8$)	$\text{round}_2(\max(0.01, M_n(1 + \mathcal{N}(0, 0.10^2))))$ - MULTIPLICATIVE 10% error, printed as 'value if served'
Complements per need	$ K(n) \sim \mathcal{U}\{1, 2\}$ (rng.integers(1,3)), sampled without replacement from the 8 complements	each required complement is dropped from the claimed list with prob. 0.15 (drop_p); if the list empties, the first is restored

Table 8: The generator and observation layer as coded, at default settings. Line references are in the released source. Two entries are worth the reader’s attention: dated complements are always drawn inside the horizon, so the availability gate fails through undated complements rather than late ones, and the scored probability target is the conditional arrival probability rather than the internal activation propensity.

Quantity	Distribution or formula as coded	What the solver observes
<i>Complements</i>		
Availability date	$r \sim \mathcal{U}(0, 1)$: $r < 0.45 \Rightarrow \rho_k \sim \mathcal{U}\{0, \dots, t_c - 1\}$; $0.45 \leq r < 0.85 \Rightarrow \rho_k \sim \mathcal{U}\{t_c, \dots, t_c + H - 1\}$; $r \geq 0.85 \Rightarrow \rho_k = \text{NEVER}$	with prob. 0.12 (and only if dated) the printed date is $\max(0, \rho_k + j)$, $j \sim \mathcal{U}\{-4..4\}$; rendered as ‘available since q’, ‘guided to q’, or ‘no announced availability’
<i>Pairs</i>		
Alignment	$\cos(\pi_p, r_n)$ (0 if either norm is 0)	not printed; recomputable only from the noisy, rounded vectors
Technical term	$1/(1 + e^{-9.0(m_{pn} - \tau_n)})$	not printed
Commercial gate	$g = 1$ if $\rho_k \leq t_c + H$ for EVERY $k \in K(n)$, else $g = 0.03$	inferable only from the printed (possibly misreported, possibly dropped) complement dates
Internal activation propensity	$p^{true} = \sigma(a(m - \tau)) \cdot g$	not printed
Activation draw	$u_{act} \sim \mathcal{U}(0, 1)$ from the private stream <code>default_rng([seed, 10000p+n])</code> ; activated iff $u_{act} < p^{true}$	only realised-before-cutoff pairings are disclosed
Activation quarter	$lo = \max(t_c - 8, \max_{k \in K(n)} \rho_k)$; $t = lo + \lfloor u_{arr}(t_c + H + 1 - lo) \rfloor$ if $lo \leq t_c + H$, else NEVER; NEVER if not activated	disclosed only if $t \leq t_c$ (REALIZED DEPLOYMENTS block)
Exact conditional target	$lo_0 = \max(t_c - 8, \max_k \rho_k)$; if $lo_0 > t_c + H$ then $p^* = 0$; else with $span = t_c + H - lo_0 + 1$, $n_b = \max(0, t_c - lo_0 + 1)$, $p_b = p^{true} n_b / span$, $p_a = p^{true} (span - n_b) / span$: $p^* = p_a / (1 - p_b)$ (0 if $p_b \geq 1$)	not observable; it is the latent-probability target, not the realised label
Label	$y = 1$ iff $t_{pn} \neq \text{NEVER}$ and $t_c < t_{pn} \leq t_c + H$	the quantity to be predicted
Candidate set	every pair with $t_{pn} > t_c$; pairs with $t_{pn} \leq t_c$ are excluded	the excluded ones are listed explicitly as already deployed
<i>Capture</i>		
Capture probability	$\text{clip}(0.15 + 0.55 u_p d_p + 0.30 m_p, 0, 1)$	only the noisy rounded u, d, m are printed; the coefficients are not stated
Realised capture	$\text{captured} = (t_{pn} \neq \text{NEVER}) \wedge (u_{cap} < p_p^{cap})$, u_{cap} the third draw of the pair stream	scored via the capture probe
<i>Commentary</i>		
Per-quarter mention rate	$base = 0.02 + 0.10 \tanh(s_p) + 0.05 p_{pn}^{true}$; $\lambda(q) = \min(base + 0.45 \cdot \mathbb{I}[t_{pn} \leq q], 0.9)$	only the total count is printed
Mention count	$\{q \in [\max(0, t_c - 8), t_c] : U_q < \lambda(q)\}$, one Bernoulli per quarter, private stream <code>default_rng([seed, 999, 10000p+n])</code>	printed as ‘discussed by N sources’ only when $N > 0$; zero-mention pairings are simply absent

Table 9: Generator and observation layer (continued from Table 8): complements, pairs, capture, commentary.

Quantity	Distribution or formula as coded	What the solver observes
<i>Observation</i>		
Capability-attribution drop rate	0.15, i.i.d. per owned capability (and per required complement)	the claimed list only
Spurious-attribution rate	nominal 0.10, but applied as $false_p / n_{caps} \times 3 = 0.03$ per non-owned capability	the claimed list only
Instrument noise	additive $\mathcal{N}(0, 0.18^2)$ per coordinate on capability AND need vectors, clipped at 0, then coordinates < 0.12 set to 0	the printed profile
Threshold noise	$\mathcal{N}(0, 0.04^2)$, clipped to $[0.2, 0.9]$	printed threshold
Materiality noise	multiplicative $\times (1 + \mathcal{N}(0, 0.10^2))$, floored at 0.01	printed value if served
Company-factor noise	$\mathcal{N}(0, 0.08^2)$ on supply/distribution/moat, clipped to $[0, 1]$	printed 0-1 scores
Complement-date misreport rate	0.12; the jitter $j \sim \mathcal{U}\{-4, \dots, 4\}$ is drawn unconditionally and applied as $\max(0, \rho_k + j)$ only when misreported AND dated	printed availability line
Printed precision	2 decimals, applied to the STORED Observed values as well as the rendered text	every number in the dossier
Corpus sections	7 blocks: header, characterisations, teardowns, problem reports, readiness notes, already-deployed list, commentary	the whole dossier
<i>Task</i>		
Probe set	candidates bucketed by (observed-blocked, mentioned>0); the first 4 of each cell in sorted order	the 16 pairings are listed verbatim in the prompt

Table 10: Generator and observation layer (continued from Table 8): observation, task.

Feature	Document source	Transformation
Profile-requirement overlap	composed profile, need requirement	dot product
Covered requirement mass	composed profile, need requirement	elementwise min, summed
Unmet requirement mass	need requirement	sum where profile is zero
Shared attribute count	both vectors	count of shared nonzeros
Stated sufficiency threshold	problem report	as printed
Margin against threshold	the two above	overlap minus threshold
All complements inside horizon	readiness notes	conjunction
Shortest complement wait	readiness notes	min, relative to cutoff
Longest complement wait	readiness notes	max, relative to cutoff
Required complement count	problem report	count
Attributed capability count	product teardown	count
Stated materiality	problem report	as printed
Derived capture proxy [‡]	product teardown	simulator capture formula
Commentary volume	financial narratives	mention count
Intercept	–	constant

Table 11: Every input to the supervised reference, with the part of the dossier it comes from. Fourteen features and an intercept; the fit standardises them, discovers the weights by gradient descent with ℓ_2 regularisation, and is never shown the simulator’s realisation formula. The single exception is marked [‡]: the formula-derived capture proxy applies the simulator’s capture formula to three printed issuer-position numbers rather than reading one printed value, and is therefore partially privileged. Removing it changes the reference by -0.0002 AUC_K , and removing it together with commentary volume by -0.0026.

n	draws	median	5th	95th	paired Δ	beats all
1	50	0.652	0.482	0.697	-0.021	26%
2	30	0.689	0.554	0.713	+0.010	73%
5	20	0.711	0.698	0.715	+0.033	100%
10	20	0.718	0.712	0.721	+0.040	100%
30	20	0.720	0.717	0.724	+0.044	100%
100	10	0.723	0.721	0.724	+0.047	100%

Table 12: Refitting the supervised reference on random training sets from the expanded pool of seeds 1000–1999, which contains the deployed fit’s seeds 1000–1099 and remains disjoint from evaluation seeds 0–99. AUC_K columns are on the 100 evaluation universes and percentiles are across draws, which is uncertainty an evaluation bootstrap does not carry. Median Δ and the last column are the paired comparison on the 30 universes the models ran on: the median paired difference against 0.669, and the share of draws beating all three single-completion model rows (0.607, 0.669 and 0.511). One-universe training is unstable; five-universe training is reliable against those rows; 100% of five-universe draws beat the fixed-budget three-completion arm.

Ranker	Mentioned AUC_K	Brier	p^* MSE
Oracle [†]	0.836 [0.821, 0.851]	0.068 [0.059, 0.077]	0.000 [0.000, 0.000]
Simulator-form [†]	0.738 [0.723, 0.755]	0.111 [0.096, 0.128]	0.044 [0.036, 0.055]
Supervised	0.736 [0.720, 0.752]	0.093 [0.082, 0.105]	0.029 [0.024, 0.035]
GPT-5.6-TERRA	0.658 [0.625, 0.691]	0.127 [0.097, 0.161]	0.044 [0.032, 0.056]
GPT-5.6-LUNA	0.605 [0.582, 0.627]	0.128 [0.099, 0.159]	0.052 [0.042, 0.063]
QWEN3-32B	0.509 [0.496, 0.523]	0.251 [0.207, 0.295]	0.168 [0.134, 0.203]

Table 13: Left: AUC_K restricted to pairings that do draw commentary, the complement of Gap- AUC_K . Right: the two probability errors on the fixed probe set, with omitted answers excluded. Brier is against the realised outcome and is the proper outcome-based score; p^* MSE is distance to the hidden world, which the lossy dossier does not let a solver reach. The oracle’s Brier is the irreducible outcome variance, not a failure.

Recovered field	Fidelity
Capability profiles	1.000
Product attributions	1.000
Need requirements	1.000
Thresholds	1.000
Materiality	1.000
Complement dates	1.000
Required complements	1.000
Mention counts	1.000
Candidate set	1.000
Issuer-position values	1.000
End-to-end AUC_K	0.724

Table 14: What the deterministic parser recovers from the rendered corpus, and what the resulting parsed-text pipeline scores. The corpus is templated, so recovery is exact and the pipeline reproduces the supervised reference; this bounds the input asymmetry between that reference and a model reading prose, and says nothing about extraction from real filings.

Ranker	n	Direction acc.	Δ percentile
Random	44	0.500 [0.500, 0.500]	+0.000 [+0.000, +0.000]
Mention-count	44	0.500 [0.500, 0.500]	+0.000 [+0.000, +0.000]
Expected value [†]	44	0.604 [0.580, 0.628]	+0.110 [+0.087, +0.135]
Supervised	44	0.590 [0.570, 0.610]	+0.104 [+0.086, +0.122]
Simulator-form [†]	44	0.594 [0.573, 0.615]	+0.108 [+0.089, +0.127]
Oracle [†]	44	0.645 [0.627, 0.663]	+0.147 [+0.128, +0.166]
GPT-5.6-LUNA	15	0.603 [0.517, 0.695]	+0.106 [+0.084, +0.131]
GPT-5.6-TERRA	15	0.562 [0.459, 0.659]	+0.110 [+0.086, +0.137]

Table 15: Counterfactual twin worlds: one complement’s availability is moved across the horizon and everything else is held fixed under common random numbers. Chance is 0.500. Δ percentile is the mean signed change in a pairing’s percentile rank between the two worlds. Model arms run on the twins that fit the query budget, so their intervals are wide.

Setting	GPT-5.6-LUNA	n	Reference gap	GPT-5.6-TERRA	n	Reference gap
Horizon 4	0.593	30	+0.123 [+0.090, +0.161]	0.646	30	+0.069 [+0.042, +0.099]
Horizon 12	0.614	30	+0.097 [+0.079, +0.114]	0.669	30	+0.041 [+0.020, +0.064]
Slope 6	0.600	30	+0.079 [+0.057, +0.102]	0.628	30	+0.052 [+0.027, +0.084]
Slope 14	0.618	30	+0.139 [+0.116, +0.163]	0.696	30	+0.061 [+0.034, +0.088]

Table 16: Horizon and logistic-slope sensitivity on the finance surface. Horizon is in quarters. Each row changes only the named parameter from the default ($H = 8$, slope 9, $K = 40$). Model AUC_K means and defined-universe counts are shown alongside paired Supervised-minus-model gaps, with 95% bootstrap CIs on the same model universes.

Quantity	Meaning in the generator	Available to a solver	Evaluation role
p_{true}	Internal activation propensity used for the first Bernoulli draw and weakly in commentary; includes the 0.03 residual gate.	No. It is not printed.	Simulator intermediate, never the scored probability.
p^*	Probability of arrival inside the horizon conditional on not already being live at the cutoff; exactly zero for a pairing with no feasible arrival window.	No. It is latent.	Probability target, oracle input, and latent- p^* error target.
y	Realised binary event that the sampled arrival quarter falls after the cutoff and inside the horizon.	Only pre-cutoff deployments are disclosed and excluded; future y is hidden.	Ground-truth label for ranking metrics and Brier score.
$\Pr(y=1 \mid D)$	Bayes-optimal probability conditional on the lossy rendered dossier D . It need not equal p^* because multiple latent worlds can produce the same dossier.	It is the quantity a perfectly calibrated document reader would infer.	Not computed by the benchmark; no reported row is claimed to be this posterior.

Table 17: Target map. The internal propensity, latent conditional target, realised label and dossier-conditional optimum are distinct quantities.

E Complete prompt and response contract

This appendix is generated directly from the canonical strings and schema used by the harness (contract digest 99e2ea63074a3a11). Runtime substitution fills the brace-delimited fields. The corpus field is the complete generated dossier and probes is the fixed list of pair identifiers for that universe.

E.1 System prompt template

You are an analyst working inside a synthetic economy.

You are given a dossier of evidence available at a cutoff {unit}. Your job is to predict which {pairing} pairs will become **COMMERCIALY REALIZED** within the next {horizon} {units}.

No document tells you whether a pairing will be realized. Some pairings are named in {commentary}; that records attention already paid to them, not evidence that they will be realized, and attention tracks the size of the holder and things that have already happened. Ranking by how much a pairing is discussed will not work.

What the evidence does support is a chain you must assemble yourself:

```
item -> its capabilities -> those capabilities'
measured attribute profile
need -> its bottleneck -> the attribute profile
that relieves it
```

A pairing is technically viable when the first item's combined attribute profile aligns with the need's required profile above that need's stated sufficiency threshold. Technical viability is not sufficient: a need also cannot be served until every complement it depends on is available. A complement with no announced availability, or one arriving after the horizon, blocks realization no matter how good the profile alignment.

Pairings listed as already deployed are history and are not candidates; every other pairing is. Do not rank a pairing that the dossier records as already deployed.

Return TWO things.

(1) `predictions`: your ranking, ordered by probability of commercial realization within the horizon. Rank by that probability alone – not by value, not by who captures it.

(2) `probes`: answers for the specific pairings listed under PROBES below. These are fixed in advance and are not a ranking; answer every one.

For each entry in either list, report:

- probability – that it becomes commercially realized within the horizon
- materiality – the value if served, on the scale the dossier reports
- capture – probability the holder captures that value rather than losing it to entrants, given it is realized. This depends on supply capacity, distribution reach and defensibility, not on whether the need gets served at all

recognized – whether this pairing is ALREADY discussed in {commentary}
prerequisites – the ids of any complements still blocking it, [] if none

Materiality, capture and recognition do not change your ranking order in (1); they are separate judgements about the same pairing.

Answer from the dossier text alone. Do not read files, run commands, or search.

E.2 Surface lexicon substitutions

```
{
  "finance": {
    "unit": "quarter",
    "units": "quarters",
    "commentary": "sell-side commentary",
    "pairing": "product-need"
  },
  "opaque": {
    "unit": "step",
    "units": "steps",
    "commentary": "reference counts",
    "pairing": "item-pair"
  },
  "space_opera": {
    "unit": "cycle",
    "units": "cycles",
    "commentary": "council chatter",
    "pairing": "artifact-need"
  },
  "anime_tech": {
    "unit": "arc",
    "units": "arcs",
    "commentary": "guild rumours",
    "pairing": "frame-need"
  },
  "cyberpunk": {
    "unit": "quarter",
    "units": "quarters",
    "commentary": "fixer traffic",
    "pairing": "deck-need"
  }
}
```

E.3 Task wrapper

{corpus}

===

Cutoff {unit}: {cutoff}. Horizon: the next {horizon} {units}.

Rank the {k} pairs most likely to become commercially realized in ({unit} {cutoff}, {unit} {end}). Use the [PT-*] and [PR-*] bracket tags as ids (e.g. product_id "P3", need_id "N7").

For each, give a calibrated probability and the evidence path you composed.

PROBES (answer all of these, in `probes`; they are not part of the ranking):
{probes}

E.4 Canonical response schema

The schema below is for $K = 40$. At request time, the harness sets the prediction list's `maxItems` to K , including $K = 90$ for the large-economy arms. Archived small-economy calls used a schema limit of 40 but were scored at their requested $K = 25$; their generation provenance is retained separately.

```
{
  "type": "object",
  "properties": {
    "predictions": {
```

```

"type": "array",
"maxItems": 40,
"items": {
  "type": "object",
  "properties": {
    "product_id": {
      "type": "string"
    },
    "need_id": {
      "type": "string"
    },
    "probability": {
      "type": "number",
      "minimum": 0.0,
      "maximum": 1.0
    },
    "materiality": {
      "type": "number",
      "minimum": 0.0
    },
    "capture": {
      "type": "number",
      "minimum": 0.0,
      "maximum": 1.0
    },
    "recognized": {
      "type": "boolean"
    },
    "prerequisites": {
      "type": "array",
      "items": {
        "type": "string"
      }
    },
    "evidence_path": {
      "type": "string"
    }
  },
  "required": [
    "product_id",
    "need_id",
    "probability",
    "materiality",
    "capture",
    "recognized",
    "prerequisites",
    "evidence_path"
  ],
  "additionalProperties": false
},
"probes": {
  "type": "array",
  "items": {
    "type": "object",
    "properties": {
      "product_id": {
        "type": "string"
      },
      "need_id": {
        "type": "string"
      },
      "probability": {
        "type": "number",
        "minimum": 0.0,
        "maximum": 1.0
      },
      "materiality": {
        "type": "number",
        "minimum": 0.0
      },
      "capture": {
        "type": "number",
        "minimum": 0.0,
        "maximum": 1.0
      },
      "recognized": {
        "type": "boolean"
      },
      "prerequisites": {
        "type": "array",
        "items": {
          "type": "string"
        }
      },
      "evidence_path": {
        "type": "string"
      }
    }
  }
}

```

```

    },
    "required": [
      "product_id",
      "need_id",
      "probability",
      "materiality",
      "capture",
      "recognized",
      "prerequisites",
      "evidence_path"
    ],
    "additionalProperties": false
  },
  "probes": {
    "type": "array",
    "items": {
      "type": "object",
      "properties": {
        "product_id": {
          "type": "string"
        },
        "need_id": {
          "type": "string"
        },
        "probability": {
          "type": "number",
          "minimum": 0.0,
          "maximum": 1.0
        },
        "materiality": {
          "type": "number",
          "minimum": 0.0
        },
        "capture": {
          "type": "number",
          "minimum": 0.0,
          "maximum": 1.0
        },
        "recognized": {
          "type": "boolean"
        },
        "prerequisites": {
          "type": "array",
          "items": {
            "type": "string"
          }
        },
        "evidence_path": {
          "type": "string"
        }
      }
    }
  }
}

```

E.5 Backend-specific enforcement

For the OpenAI frontier arms, the harness recursively marks every object closed and every declared property required, then sends the schema with strict structured output. It sends the rendered system template as instructions, the rendered task as input, reasoning effort medium, and a 48,000-token output limit; it does not send temperature. For the reported vLLM arm, the canonical schema is supplied to guided decoding. If the decoder cannot compile all constraints, its fallback removes `maxItems` and `additionalProperties`, and then numeric bounds if required. The scorer independently enforces the top- K cap, candidate validity and probability range.