Learning Clinical-Trial Strategy: Offline Policy Training for Decision Agents

William Bolton [1] **Philip Torr** [1]

Abstract

Clinical development is sequential decision-making under uncertainty, where a sponsor must plan a portfolio of experiments from heterogeneous evidence. We study this setting by framing oncology clinical development as an offline decision-making problem in which an agent predicts the next six-month trial portfolio of an oncology drug program from information available at the decision date. To support this, we construct a temporal dataset that combines 31.7k heterogeneous public data records, including trial registries, regulatory reviews, sponsor filings, utilization data, and epidemiology, into 881 offline decision episodes across 45 historical programs. We compare four offline objectives: behavioral cloning, reward-weighted behavioral cloning, learned-reward training, and value-based implicit Q-learning against four frontier LLM agents that share a common date-gated retrieval scaffold across held-out drug, sponsor, drug-class, and temporal splits. Models trained offline outperform every non-fine-tuned baseline in the post-August 2025 contamination-clean holdout, obtaining 40.8–51.4% indication F1 against 17.7% for the strongest tool agent. Among offline objectives, value-based Q-learning is consistently strongest, recovering the full indication, phase and strategy decision 20.3% of the time and remains the strongest arm under autoregressive rollout. These results show that structured offline learning can teach agents to plan clinical experiments.

1. Introduction

Clinical development is a sequential decision-making problem under uncertainty. At each point in a drug program, a sponsor must decide what to plan next, whether to expand into a new indication, move to a later phase, test a combination, change geography, collect real-world evidence, or stop investing. These decisions are made with imperfect information. Sponsors combine prior trials, regulatory signals, competitor activity, epidemiology, and commercial context, but future clinical, regulatory, and market outcomes remain unknown. Furthermore, the consequences of decisions are delayed, since a trial must run for its outcome to be observed, and it may only affect an approval, revenue, or patients years in the future.

Most machine-learning work on clinical trials asks whether a predefined trial will succeed or investigates problems such as whether a patient matches a trial (Fu et al., 2022; Jin et al., 2024; Chen et al., 2025). We instead study the upstream strategy. Given the state of a drug program and its external environment at time t, what trials should be launched next?

We instantiate this problem in oncology by constructing a temporal dataset that turns heterogeneous public evidence into structured states, trial-portfolio actions, and delayed outcome signals for offline sequential decision-making. Methodologically, this work connects to offline reinforcement learning and sequence modeling from logged trajectories, where policies are learned from fixed datasets rather than through online interaction (Levine et al., 2020; Prudencio et al., 2023; Chen et al., 2021; Janner et al., 2021). Our reward-weighted objective is closest to advantage-weighted behavioral cloning, where imitation learning is reweighted by return or advantage estimates (Peng et al., 2019; Nair et al., 2020; Kostrikov et al., 2021). It also relates to LLM-agent work in which models use tools to support scientific experimentation, and simulated clinical reasoning (Huang et al., 2025; Schmidgall et al., 2024). The key difference is that our actions are timestamped trial portfolios, the evidence is date-gated, and rewards come from external outcomes observed after the decision. This work is a first step toward models that learn from the history of clinical development to support better trial strategy under uncertainty.

Contributions.

- We formulate clinical-trial strategy as an offline training problem over structured trial portfolios.
- We construct a temporal oncology dataset of 881 de-

[1]Department of Engineering Science, University of Oxford, Oxford, United Kingdom. Correspondence to: William Bolton <will.bolton@eng.ox.ac.uk>.

Accepted at ICML 2026 Workshop on Generative and Agentic AI for Biology (GenBio).

cision episodes across 45 drug programs, assembled from 31.7k public records spanning trial registries, FDA and EMA reviews, SEC filings, CMS utilisation, SEER epidemiology, and literature. Yielding 27k structured fields to predict across 1,956 launch actions.

- We compare four offline training objectives against four frontier LLM agents on a shared date-gated retrieval scaffold, across drug, sponsor, drug-class, and temporal splits.

2. Methods

2.1. Problem formulation

Clinical-development strategy is formulated as an offline decision-making problem over historical drug-program windows. For each drug program p and each six-month window t, the model observes a state $s_{p,t}$ containing only information available before t, including completed and ongoing trials, prior approvals, the competitive landscape, epidemiology, and market positioning. It then predicts an action $a_{p,t}$, defined as the set of trials launched by the sponsor during that window, with each trial represented as a 14-field structured object. Each program-window pair is also assigned an outcome-derived reward $R_{p,t}$, which summarizes downstream real-world signals associated with those decisions.

2.2. Dataset construction

We assembled a novel temporal dataset on oncology drug development programs by curating public information. Collection was seeded by per-drug configuration files containing canonical drug names, synonyms, sponsor identifiers, regulatory keys, and source-specific queries. This was necessary because across different sources the same drug can appear under different brand names, international non-proprietary names, development codes, and sponsor aliases.

The episode state, action, and reward are constructed from five primary public-source streams. ClinicalTrials.gov provides trials, eligibility, outcomes, and trial timing, with each registered trial identified by a unique `NCT` accession number. Drugs@FDA provides review PDFs that our approval linker mines for those NCT identifiers and trial acronyms. SEC EDGAR provides filings parsed for quarterly revenue. CMS Part D provides beneficiary counts, and SEER provides US cancer epidemiology. Raw fetches are stored immutably with content-hash addressing and full provenance, then normalized into trial records, document records, quarterly revenue entries, and per-trial competitive snapshots evaluated at each trial start date.

FDA approvals are linked back to supporting ClinicalTrials.gov trials using NCT identifier matches in review documents, trial-acronym matching, and sponsor-scoped fallback rules. This allows approval credit to be assigned to the decision window in which the linked pivotal trial was launched, allowing delayed approvals to be credited to the earlier launch decision. The resulting offline dataset contains 881 historical (s, a, R) triples across 45 drug programs.

The 881 windows are partitioned into eight held-out evaluation splits that test different generalization axes. Five single-drug holdouts (tagrisso, gleevec, imbruvica, keytruda, opdivo) hold out every window of one drug program to test transfer to a fully unseen drug with rich training history. A sponsor-blind split holds out all four AstraZeneca programs (tagrisso, lynparza, imfinzi, calquence) to test transfer between sponsors. The drug-class split holds out all PD-1 and PD-L1 checkpoint inhibitors to test transfer across pharmacology. Finally, two temporal splits hold out windows with a decision date after a model's knowledge cutoffs. A post-September 2024 split aligned with the Qwen-2.5 cutoff, and a post-August 2025 split aligned with the latest cutoff for GPT-5.4 and Opus 4.5, to provide a prospective assessment of model performance that cannot be contaminated by training data. Each split's remaining windows form its own training set, and together the eight test sets total 432 held-out windows.

2.3. Action representation

Every predicted and ground-truth trial conforms to a JSON Schema enforced at decode time. The schema fixes 14 required fields per trial, including the indication, phase, strategy, study design, enrollment bucket, comparator, region, and endpoint fields, with categorical vocabularies enumerated where appropriate. Structured decoding ensures that locally served models emit the same field set, making per-dimension F1 directly comparable across models. The full schema is given in Appendix A.1.

2.4. Outcome rewards and offline objectives

We compare four offline policy-learning objectives using the same backbone, prompts, action schema, decoding constraints, and fine-tuning setup. The backbone is Qwen-2.5-7B-Instruct, fine-tuned with QLoRA. Each input is rendered as a historical state for a program-window pair, and the target is the JSON serialization of the trials launched in that window. The first three objectives differ only in how examples are weighted during training. The fourth, implicit Q-learning, instead learns a value function from the logged outcomes and weights each example by the estimated advantage of its action, so it can favour decisions that beat the typical outcome from a comparable state rather than simply up-weighting high-reward windows.

The first objective is behavioral cloning. Every example receives weight 1.0, and the model is trained to imitate

the historical sponsor action distribution. This provides an unweighted supervised fine-tuning baseline that does not use downstream outcome information.

The second objective is reward-weighted behavioral cloning, an offline RL-style objective inspired by advantage-weighted regression. Each launched trial i receives the per-trial reward

$$r_i = 2\,a_i - f_i + \log(1 + \mathrm{rev}_i), \tag{1}$$

where $a_i = 1$ if trial i became pivotal in an FDA approval and $f_i = 1$ if it failed or was terminated. rev_i is the drug's peak annual revenue attributed to trial i, split across the program's approvals (weighting later approvals more) and then across the pivotal trials supporting each. The window score $R_{p,t}$ is the mean of r_i over the trials launched in that window, a direct attribution of realized regulatory and commercial outcomes to the trials that produced them. We standardize $R_{p,t}$ within each split and convert it into a clipped sampling weight $w_{p,t} = \mathrm{clip}(0.5 + \tilde{R}_{p,t}, 0.1, 3.0)$, training on the rebalanced dataset so higher-reward windows are sampled more often. We analyse the reward terms and this per-trial credit-assignment choice in Appendix B.3.

The third objective is learned-reward-weighted behavioral cloning. Rather than the fixed coefficients of Eq. 1, we *learn* the reward (Appendix A.7) by regressing trial-design features onto the same per-trial outcome target, so the model infers the objective from the structure of the trials instead of from a rule-based formula. The fitted per-trial scores are then turned into sampling weights by the same method as the rule-based reward.

The fourth objective is value-based offline reinforcement learning with implicit Q-learning (IQL). While the previous objectives reweight imitation by an *observed* reward, IQL weights examples by an *estimated* advantage, giving policy improvement beyond behavioral cloning. We treat each launched trial as an action and fit an IQL value model over all programs: a state-value function $V_\psi(s)$ trained by expectile regression with expectile $\tau = 0.7$ (so V_ψ tracks an upper quantile of return rather than its mean) and an action-value $Q_\phi(s,a) = R_{p,t} + \gamma\, V_\psi(s')$ with discount $\gamma = 0.95$ that bootstraps the discounted per-trial reward (Kostrikov et al., 2021). Each training example is then weighted by an advantage-weighted transform of $A(s,a) = Q_\phi(s,a) - V_\psi(s)$, which measures how much a launch decision beat the field-average value from a comparable program state, at inverse temperature $\beta = 0.5$, with per-trial advantages aggregated to a window weight under the same clip-and-oversample scheme used for the previous rewards.

2.5. Baselines and evaluation

Our primary comparison is the four fine-tuned models against four tool-using agents (GPT-5.4, Claude Opus 4.5, Gemini 3.1 Pro, and Qwen3-235B-Thinking-2507) on a shared scaffold. We also included prompt-only approaches for Qwen-2.5-7B-Instruct (backbone for the trained models), GPT-5.4 and a random baseline. All approaches return the same 14-field action schema. The fine-tuned models and the prompt-only LLMs are run with a structured summary of the program, its history, and wider context at the decision date (Appendix A.8). Each agent receives the drug identifier and decision date plus 10 typed retrieval tools covering sponsor trial history, competitor and indication landscape, SEER epidemiology, sponsor revenue, and document retrieval of FDA reviews, EMA EPARs, and publications from OpenAlex and PubMed. Each tool is date-gated, so information dated on or after the decision window is not returned. Prompt-format ablations for the prompt-only baselines are reported in Appendix B.8.

Each predicted portfolio is scored against the held-out ground-truth portfolio by greedy one-to-one pairing. Free-text fields (indication, trial description, comparator, and combination partners) are matched by embedding similarity, and categorical or enumerated fields by exact match; this rule applies to all metrics below, and credits same-disease paraphrases that lexical string matching would miss (Appendix A.10).

We report the following metrics. *Indication F1* is the standard F1 over indication matches: precision is matched pairs divided by predicted trials, recall is matched pairs divided by ground-truth trials. It measures whether the model identifies the correct disease and biomarker setting. *Strict F1* measures wider-trial alignment; a pair is formed only if indication, phase, and strategy all match. *Field-match %* is, for an indication paired trial, the percentage of the 13 design fields that match, averaged over matched pairs.

Beyond the next-window matching, we analyse model behaviour at historically decisive forks. At forks where a failed trial and a successful same-mechanism comparator started within two years (62 cross-drug forks across the five single-drug holdouts), we measure whether a model recommends the winning design and avoids repeating the failing one, reported as a volume-invariant net share of each model's own design predictions: the fraction matching the winning trial minus the fraction matching the failed one.

To probe robustness to compounding state error, we additionally evaluate the offline-trained models, Qwen-2.5-7B, GPT-5.4, and the GPT-5.4 agent in an autoregressive rollout on the five single-drug holdouts. In this setting, each model sees its own earlier predicted launches as part of the launch history for later windows, replacing the oracle history given

by the static evaluation. We report the change in metrics from the static to the autoregressive evaluation as a measure of how much each method's accuracy degrades when conditioning on its own outputs rather than ground truth.

3. Results and Discussion

Figure 1 shows results on both the full held-out set and the contamination-free post-August 2025 subset, on which no model could have memorised the decisions. On the post-August 2025 subset the offline-trained models outperform every non-fine-tuned baseline; the best offline method, IQL, reaches 51.4% Indication F1 against 17.7% for the strongest tool agent (Gemini 3.1 Pro), and the advantage is smaller on the full set. On the larger post-September 2024 set where some memorization could have occurred from frontier models, IQL significantly exceeds the strongest agent on Indication F1 (49.2% vs 21.0%); over the 43 windows both were scored on, the paired-mean gap is 29.4 points (95% CI [13.1, 45.9], permutation p = 0.0016). The agentic setup did not show differentiated performance vs the one-shot counterpart for GPT-5.4. The reduction in the agents' scores on the post-cutoff subset suggests they rely on pre-training knowledge where it is available, whereas the fine-tuned models learn a decision mapping from the offline episodes.

The offline trained models show particularly strong performance in the Strict F1 metric, which rewards broader trial design decisions over just indication prediction (Figure 1, Appendix Table 6). IQL obtains 20.3% versus 4.4% for the strongest tool agent (Gemini 3.1 Pro). The low absolute Strict F1 values are expected, since the historical action is one realized decision among multiple plausible counterfactuals. We therefore treat the metrics as alignment with historical sponsor behavior rather than proof of optimality.

Per-split breakdowns (Table 1) show that the four trained models do not rank in one fixed order across holdouts. Due to memorisation, the base Qwen-2.5-7B model has highly erratic performance across splits (Indication F1 10.1 to 56.3). It excels on narrow single-indication programs predicting NSCLC for 98% of its Tagrisso-holdout trials but collapses on broad multi-indication drugs like Imbruvica (ibrutinib spans CLL, mantle-cell lymphoma, Waldenström's, and GVHD), where no single memorized prior fits.

By evaluating decisions at historically decisive forks, we can understand the propensity for models to avoid proposing failed trials (Table 2). Behavioral cloning significantly repeats failed designs; however, the IQL objective enables models to avoid significant repetition of failed trials (net -1.3%), demonstrating the advantages of value-based credit assignment over behavioral cloning.

Static evaluation gives each model the true historical launch history at every window. Autoregressive evaluation instead

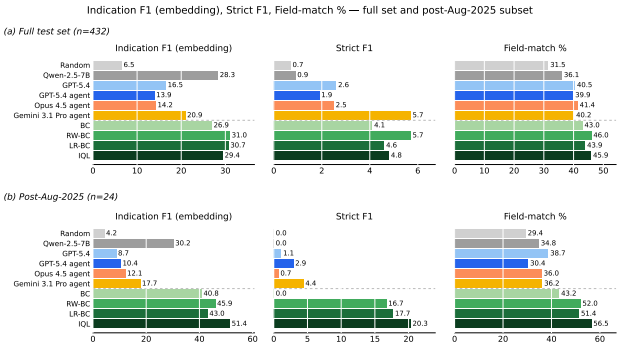

Figure 1. Results are reported as windows-weighted means across the eight held-out evaluation splits. [a] is the union of all n=432 held-out windows, while [b] is the cross-split subset of n=24 windows whose decision date falls after August 2025, the most recent knowledge cutoff for the models evaluated here. BC, RW-BC, LR-BC, and IQL denote behavioral cloning, reward-weighted behavioral cloning, learned-reward behavioral cloning, and value-based implicit Q-learning.

Table 1. Indication F1 by holdout split

		Baseline + tool agents					Offline-trained models			
Split	n	Qwen-2.5-7B	GPT-5.4	Opus 4.5	Gemini 3.1 Pro	Qwen3-235B	BC	RW-BC	LR-BC	IQL
tagrisso	25	**56.3**	19.6	16.0	16.9	33.9	32.9	45.9	48.3	50.3
gleevec	35	30.6	10.0	9.2	7.7	7.0	36.4	40.6	26.3	**43.5**
imbruvica	19	10.1	5.1	1.6	2.5	21.4	14.9	38.1	**45.2**	40.2
keytruda	26	32.3	12.0	12.4	17.5	24.4	**37.1**	20.2	25.8	15.8
opdivo	31	20.6	18.0	21.5	25.6	24.2	26.7	27.3	**30.1**	26.3
AstraZeneca sponsor	96	31.1	13.5	14.3	22.5	25.7	21.6	**34.1**	33.4	28.1
Drug class (PD-1/L1)	155	**24.4**	13.5	15.9	24.3	21.7	21.6	23.8	23.5	19.8
post-Sep 2024 (Qwen-2.5 cutoff)	44	28.7	17.3	11.7	21.0	17.4	44.7	40.3	41.4	**49.2**
† post-Aug 2025 (all-model cutoff)	24	30.2	10.4	12.1	17.7	11.9	40.8	45.9	43.0	**51.4**

Table 2. Avoiding the failed trial: net share across 62 cross-drug forks, with bootstrap 95% CIs.

Model	winner%	repeat%	net [95% CI]
IQL	67.3	68.6	-1.3 $[-11.5, +9.0]$
GPT-5.4 agent	53.8	60.3	-6.4 $[-18.6, +5.1]$
Opus 4.5 agent	70.1	76.5	-6.4 $[-15.0, +1.6]$
LR-BC	71.1	81.5	-10.4 $[-20.0, 0.0]$
RW-BC	60.4	76.2	-15.9 $[-25.0, -6.1]$
BC	62.8	78.6	-15.9 $[-26.9, -4.8]$

feeds each model its own previous predictions, testing sensitivity to compounding errors. Among the offline-trained models, IQL is the most robust on the design-discriminating Field-match % (Table 3), whereas the reward-weighted arm degrades most. Across all dimensions, the fine-tuned arms remain well above the baselines, demonstrating that policies enable the strategic direction of drug development to be trained into the model's weights, making it robust to its own errors.

This work has several limitations. First, the rewards are retrospective associations, not causal estimates of the value of a trial-launch decision. Approval and revenue depend on drug quality, sponsor resources, pricing, and competition that the public state only partly captures, and no objective we tried, including value-based IQL, cleanly separates a

Table 3. Static versus autoregressive (AR) performance on the five single-drug holdouts.

	Indication F1			Strict F1			Field-match %		
Method	Static	AR	Δ	Static	AR	Δ	Static	AR	Δ
BC	29.6	28.9	−0.7	4.8	5.6	+0.8	48.4	46.3	−2.1
RW-BC	34.4	26.8	−7.6	6.4	2.8	−3.6	49.3	45.0	−4.3
LR-BC	35.1	30.2	−4.9	4.5	4.2	−0.3	46.8	45.5	−1.3
IQL	35.2	33.3	−1.9	3.9	3.9	0.0	47.1	46.5	−0.6
Qwen-2.5-7B	30.0	24.3	−5.7	0.8	1.2	+0.4	37.9	37.4	−0.5
GPT-5.4 tool-agent	12.9	11.8	−1.1	1.6	1.7	+0.1	40.1	36.7	−3.4

decision's own quality from the program trajectory it sits in (Appendix B.3). Relatedly, per-trial credit is local: a trial is penalised only for its own failure, so an early trial that enabled a program later abandoned in a subsequent phase is scored roughly neutrally; propagating a program's eventual failure back to its enabling trials (causal-lineage credit) is future work, since naive back-propagation reblames trials indiscriminately. Second, the post-August 2025 contamination-controlled subset is small ($n = 24$), so it is a useful comparison but not a definitive estimate of prospective performance. Finally, the schema ensures comparable outputs but abstracts away details that matter in real trial design, including endpoint definitions and operational feasibility. We therefore read this as evidence that historical development data can contain a learnable strategic signal enabling agents to plan biomedical experiments, not a claim that the learned policy is optimal.

These results motivate more work on closed-loop agentic systems and offline-to-online methods. A north star would be a clinical-development simulator that supports counterfactual rollouts to turn the retrospective reward into a causal one and enable online adaptation. Nearer-term steps include training the offline objective on retrieval-augmented prompts using already-collected agent transcripts, so that hypothesis generation, evidence retrieval, and structured action planning are learned jointly. Furthermore, new sponsor decisions can be incorporated as fresh episodes, and hierarchical RL with sub-goals could be used to tackle the problems of long horizons and sparse delayed rewards (Shin & Kim, 2023; Levine et al., 2020; Prudencio et al., 2023). Formulating, planning, and acting on clinical-development strategy remains a difficult sequential decision-making problem that frontier agents do not yet solve.

Code and data availability

Code available at https://github.com/WilliamBolton/clinical-trial-strategy-agents.
The structured dataset is available at https://huggingface.co/datasets/WillBolton/oncology-trial-strategy.

Impact Statement

This paper presents a research artifact for studying clinical-development strategy from public oncology data. The models are intended for retrospective structured forecasting and evaluation, not for autonomous trial design, regulatory submission, investment decisions, or patient care. The main risks are misuse and bias amplification, since public development histories overrepresent large sponsors, marketed assets, common indications, and strategies that leave visible regulatory or commercial traces. Models trained on this history may therefore reproduce incumbent development patterns and underrepresent rare, pediatric, low-resource, or commercially weaker indications. We mitigate these risks by using only public sources, date-gating evidence at the decision window, evaluating across drug, sponsor, drug-class, and temporal splits, and treating outputs as program-level research predictions rather than clinical or commercial recommendations. No patient-level records, confidential sponsor materials, or proprietary information was used in this research.

Acknowledgements

This work was funded by ARIA, DSIT and Pillar VC under the Encode: AI for Science Fellowship.

References

Chen, J., Hu, Y., Cai, M., Lu, Y., Wang, Y., Cao, X., Lin, M., Xu, H., Wu, J., Cao, X., et al. Trialbench: Multi-modal ai-ready datasets for clinical trial prediction. *Scientific Data*, 12(1):1564, 2025.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Fu, T., Huang, K., Xiao, C., Glass, L. M., and Sun, J. Hint: Hierarchical interaction network for clinical-trial-outcome predictions. *Patterns*, 3(4), 2022.

Huang, K., Zhang, S., Wang, H., Qu, Y., Lu, Y., Roohani, Y., Li, R., Qiu, L., Li, G., Zhang, J., et al. Biomni: A general-purpose biomedical ai agent. *biorxiv*, 2025.

Janner, M., Li, Q., and Levine, S. Offline reinforcement learning as one big sequence modeling problem. *Advances in neural information processing systems*, 34: 1273–1286, 2021.

Jin, Q., Wang, Z., Floudas, C. S., Chen, F., Gong, C., Bracken-Clarke, D., Xue, E., Yang, Y., Sun, J., and Lu, Z. Matching patients to clinical trials with large language models. *Nature communications*, 15(1):9074, 2024.

Kostrikov, I., Nair, A., and Levine, S. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Levine, S., Kumar, A., Tucker, G., and Fu, J. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Nair, A., Gupta, A., Dalal, M., and Levine, S. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

Peng, X. B., Kumar, A., Zhang, G., and Levine, S. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

Prudencio, R. F., Maximo, M. R., and Colombini, E. L. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE transactions on neural networks and learning systems*, 35(8):10237–10257, 2023.

Schmidgall, S., Ziaei, R., Harris, C., Reis, E., Jopling, J., and Moor, M. Agentclinic: a multimodal agent benchmark to evaluate ai in simulated clinical environments. *arXiv preprint arXiv:2405.07960*, 2024.

Shin, W. and Kim, Y. Guide to control: Offline hierarchical reinforcement learning using subgoal generation for long-horizon and sparse-reward tasks. In *IJCAI*, pp. 4217–4225, 2023.

A. Supplementary Methods

A.1. Action schema

Every predicted and ground-truth trial in our benchmark conforms to a JSON Schema with 14 required fields, enumerated vocabularies, and a portfolio-size cap of 10 trials per decision window. The schema is enforced at decode time by vLLM's `GuidedDecodingParams` for the locally served Qwen-2.5-7B full and the offline-trained models. For the prompted-LLM baselines, including the Qwen API and the GPT-5.4 *min*, *full*, and *agent* variants, the schema is described in the system prompt and JSON output is parsed post-hoc with brace-balanced extraction and truncation repair. The 14 fields and their enumerated vocabularies are listed in Table 4, and an example portfolio entry is shown in Listing 2.

Table 4. Action schema. Every predicted trial is a 14-field structured object. Categorical fields are constrained to a fixed enum at decode time. List fields (combination partners, secondary endpoint types) are open-vocabulary. The comparator field accepts a string or `null`.

Field	Type	Vocabulary or notes
`indication`	string	free-text disease label
`trial_description`	string	free-text trial summary
`phase`	enum	Phase 1, Phase 2, Phase 3, Phase 4
`strategy`	enum	confirmatory, indication_expansion, combination, dose, geographic, rwe
`study_type`	enum	interventional, observational
`allocation`	enum	randomized, non_randomized, na
`masking`	enum	open_label, blinded
`n_arms_bucket`	enum	1, 2, 3+
`enrollment_bucket`	enum	small, mid, large
`combo_partners`	list of strings	open-vocabulary (drug names)
`comparator`	string | null	free-text or null
`region`	enum	single_country, multi_regional
`primary_endpoint_type`	enum	ORR, PFS, OS, DLT, safety, other
`secondary_endpoint_types`	list of strings	open-vocabulary

```
{
  "indication": "Non-Small Cell Lung Cancer",
  "trial_description":
      "Pembrolizumab vs platinum chemotherapy in 1L NSCLC",
  "phase": "Phase 3",
  "strategy": "indication_expansion",
  "study_type": "interventional",
  "allocation": "randomized",
  "masking": "open_label",
  "n_arms_bucket": "2",
  "enrollment_bucket": "large",
  "combo_partners": [],
  "comparator": "platinum doublet chemotherapy",
  "region": "multi_regional",
  "primary_endpoint_type": "PFS",
  "secondary_endpoint_types": ["OS", "ORR"]
}
```

Figure 2. An example trial conforming to the action schema. Real predictions and ground-truth trials follow this shape exactly. The surrounding portfolio is a list of one to ten such objects.

A.2. Primary data sources and auxiliary corpus

The benchmark state, action, and reward are constructed from five primary public-source streams. ClinicalTrials.gov provides trials, eligibility criteria, outcomes, and trial timing. Drugs@FDA provides review PDFs used by the approval linker. SEC EDGAR provides 10-K, 10-Q, 6-K, and 20-F filings parsed for quarterly product revenue. CMS Part D provides

a patient-reach signal. SEER provides US cancer epidemiology.

The same ingestion pipeline also collects auxiliary sources, including EMA EPARs, DailyMed, OpenAlex, PubMed, GDELT, AACT, ESMO, PatentsView, and FDA designations. These sources populate the document corpus available to the LLM tool-using baseline, but they do not contribute to the behavioral-cloning, reward-weighted, or learned-reward training state or reward.

Raw fetches are stored immutably with content-hash addressing and full provenance. Each fetch records the source URL, fetch timestamp, content type, and source identifier. Fetches are then normalized into trial records, document records, per-trial competitive snapshots evaluated as of each trial start date, and quarterly revenue entries.

A.3. Parsing and temporal snapshot construction

ClinicalTrials.gov entries are converted into trial records containing arms, eligibility criteria, outcome measures, dates, references, and registry status. Drugs@FDA review PDFs are parsed into document records with extracted text and metadata. SEC filings are parsed from HTML and used to extract quarterly drug-level revenue where available. CMS Part D records are used to derive beneficiary counts. SEER tables provide indication-level epidemiological context.

For each sponsor trial start date, the pipeline builds a competitive snapshot using only information available up to that date. The snapshot records indication-relevant competitor trials, prior approvals, treatment alternatives, and epidemiology. Later events are excluded from the snapshot used to construct the state.

A.4. Approval linkage

FDA approval rewards require linking approvals to the trials that supported them. The linker first searches FDA review text for direct NCT identifier mentions. It then applies a trial-acronym map built from ClinicalTrials.gov brief titles and acronym fields. This handles variants such as `KEYNOTE-024`, `KEYNOTE 024`, and `keynote024`. If direct identifier and acronym matching fail, the linker falls back to sponsor trials in the approved indication, prioritizing phase 3 or later trials within a window before approval.

A.5. Reward normalization and sampling weights

Observed rewards are normalized within each training split before being converted into sampling weights. For a program-window pair (p, t), the normalized reward $\tilde{R}_{p,t}$ is converted into a clipped sampling weight:

$$w_{p,t} = \mathrm{clip}\Big(0.5 + \tilde{R}_{p,t},\, 0.1,\, 3.0\Big)\,. \tag{2}$$

The lower bound keeps low-reward windows present with non-zero probability. The upper bound prevents a small number of high-reward windows from dominating the fine-tuning dataset.

Reward-weighted training is implemented through stochastic oversampling. Each example is included $\lfloor w_{p,t} \rfloor$ times, with one additional copy sampled with probability $w_{p,t} - \lfloor w_{p,t} \rfloor$. The resulting dataset is then trained with the same next-token prediction objective as behavioral cloning.

A.6. Fine-tuning implementation

All fine-tuned models use Qwen-2.5-7B-Instruct with QLoRA. States are rendered as the state prompt by `build_temporal_prompt`. QLoRA uses 4-bit NF4 quantization, LoRA rank $r = 64$, $\alpha = 128$, dropout 0.05, and target modules `q_proj`, `k_proj`, `v_proj`, `o_proj`, `gate_proj`, `up_proj`, and `down_proj`.

Training uses 8 epochs, per-device batch size 4, gradient accumulation 4, effective batch size 16, learning rate $3{\times}10^{-4}$, cosine scheduling, 3 percent warmup, bf16, maximum sequence length 4096, a 10 percent held-out validation split, and random seed 42. Optimisation uses `adamw_8bit` through Unsloth or `adamw_torch` through Hugging Face, with TRL `SFTTrainer`.

A.7. Learned reward model

For the learned-reward objective we replace the rule-based reward of Eq. 1 with one learned from data. Each trial is mapped to a feature vector summarising its design (phase, strategy, study type, enrollment, combination, endpoint, and novelty) together with program context (maturity, indication breadth, competitive pressure, and prior outcomes). A regression model is fit to predict the per-trial reward target of Eq. 1 from these features, so the learned reward captures which design choices tend to precede good outcomes rather than encoding them by hand. Its per-trial predictions are averaged over each window, standardized, and clipped into sampling weights exactly as for the rule-based reward (Eq. 2); training is otherwise identical.

A.8. Prompt

The state prompt contains, for the program at the decision date:

- drug name, mechanism of action, and target;
- completed and ongoing sponsor trials, each with phase, start and completion dates, enrolment, and outcome label where available;
- prior FDA approvals and their indications;
- the current competitive landscape: other sponsors' active trials in the same indication, MoA-enriched, with CMS Part D spending where available;
- indication alternatives (treatments competing for the same patient population);
- SEER incidence and 5-year survival for each indication of interest;
- the last eight quarters of sponsor product revenue;
- current market position (revenue rank, share trajectory).

None of these fields contain information dated on or after the decision window. The same prompt is shown to the fine-tuned models at training and inference time, and to the prompt-only LLM baselines that are run in "full" mode.

A.9. Inference and baseline implementation

Local inference uses vLLM 0.7.3 with bf16, `gpu_memory_utilization=0.85`, and `max_model_len=16384`. LoRA serving uses `enable_lora=True` and `max_lora_rank=64`. Local models use temperature-zero decoding and the JSON-schema-guided decoder.

GPT-5.4 baselines run through the OpenAI API at temperature zero. Tool-using agents on backbones other than GPT-5.4 (Claude Opus 4.5, Gemini 3.1 Pro, Qwen3-235B-Thinking-2507) are routed through OpenRouter so the same agent scaffold and 10-tool retrieval interface can be applied uniformly across backbones. For all API-served models the schema is described in the system prompt with a worked example rather than enforced by the local constrained-decoding stack, and a lightweight parser removes Markdown fences, extracts the largest balanced JSON object, and attempts simple truncation repair. API cache keys hash the state, candidate set, evaluation mode, and prompt version, so re-running an evaluation incurs no API cost.

The random baselines use empirical distributions from the training data. The mode-random baseline samples common field values from marginal modes. The marginal-random baseline samples field values independently from empirical training marginals.

A.10. Metric implementation

Scoring proceeds in two steps. *First, pairing.* Predicted portfolios are greedily paired to ground-truth portfolios using indication similarity. For each predicted trial, the evaluator selects the best-matching unpaired ground-truth trial using a rule-based score over the indication and trial-description strings. Strings are lowercased, stripped of stage and severity prefixes, and normalized with a hand-curated oncology dictionary.

Second, field matching. Once trials are paired, free-text fields (indication, trial description, comparator, and combination partners) are matched by cosine similarity between sentence embeddings (OpenAI `text-embedding-3-small`) with threshold τ=0.62, while categorical and enumerated fields (phase, strategy, study type, allocation, masking, arm and enrollment buckets, region, and endpoint types) are matched exactly. Embedding matching credits paraphrases and abbreviations of the same disease (for example "CLL" and "chronic lymphocytic leukemia") that lexical string matching scores as misses, which otherwise understates the free-text agent outputs and overstates the gap to the models. The threshold τ=0.62 is the midpoint between same-disease paraphrases (cosine $\approx$ 0.7–0.9) and distinct diseases ($\approx$ 0.3–0.5), and the method ranking is invariant for $\tau \in [0.50, 0.75]$.

For paired trials, categorical and ordinal fields are evaluated by exact equality. Secondary endpoints are evaluated by set overlap. Combination partners and comparators are evaluated by a normalized substring rule. Field-match % averages the percentage of matched fields per paired trial, and Strict F1 requires indication, phase, and strategy to all match within a paired prediction.

Because portfolio prediction is set-valued, evaluation depends on how predicted trials are paired to observed trials. We therefore treat Indication F1 as the headline alignment metric, use Field-match % and Strict F1 to test increasingly structured agreement.

A.11. Reproducibility and release

The random seed is fixed for dataset splitting, reward-weighted oversampling, reward-model pair sampling, and fine-tuning. Each model directory contains the fine-tuning split, per-step `train_history.jsonl`, and an MLflow run. Training runs on a single NVIDIA H100 80 GB. Per-model wall-clock time is approximately 13 to 30 minutes depending on split size. The full sweep contains 32 models from 8 splits and 4 training objectives.

B. Supplementary results

B.1. Random baselines: uniform, mode, and marginal

We evaluated three random baselines. *Uniform* samples each categorical dimension uniformly over the schema vocabulary. *Mode* predicts the most-common training value for every dimension. *Marginal* samples each dimension from the training marginal. All three predict a portfolio whose size is sampled from the training distribution of portfolio sizes, with mode-random using the rounded mean. Results are averaged over three seeds. Mode-random is strongest on every metric (Table 5) and is reported as "Random" in the main paper.

Table 5. Random baselines on the full held-out set (n=432). Mode-random predicts the same modal tuple per window; uniform and marginal sample per categorical dimension. Mode is the strongest of the three on every metric.

Variant	Ind. F1	Field-match %	Strict F1
Uniform	7.3%	2.9%	0.1%
Mode	**11.4%**	**5.1%**	**0.6%**
Marginal	6.6%	3.3%	0.2%

Mode-random reaches 0.0% Strict F1 on the 24-window post-August 2025 subset because Strict F1 requires the predicted trial to match on indication, phase, and strategy and to land on the right pair, and mode-random predicts the same fixed tuple in every window. On the full 432-window set the modal tuple coincides with the ground-truth tuple in roughly three windows, giving 0.6 percent. On the 24-window subset the intersection happened to be empty, which is a small-n artefact rather than an evaluation error and is consistent with the 0.6 percent rate observed on the full set.

B.2. Headline leaderboard with confidence intervals

Table 6 reports the exact headline numbers visualised in Figure 1: the contamination-controlled post-August 2025 cross-split subset (n=24). Brackets are 95% confidence intervals from a nonparametric bootstrap over the held-out windows (percentile method). The intervals are wide at n=24, but on Strict F1 the three per-trial-reward arms exclude zero (IQL $[7.2, 36.2]$, RW-BC $[5.8, 29.7]$, LR-BC $[5.8, 31.3]$) while behavioral cloning and the base model are identically zero, so the whole-decision result is supported even on this small slice.

Table 6. Contamination-clean leaderboard, post-August 2025 cross-split subset (n=24), with bootstrap 95% CIs. Bold marks the best method per column. IQL is the value-based offline-RL arm.

Method	Indication F1 [95% CI]	Field-match % [95% CI]	Strict F1 [95% CI]
IQL (value-based)	**51.4** [34.1, 68.1]	**56.5** [49.2, 64.2]	**20.3** [7.2, 36.2]
RW-BC	45.9 [30.0, 61.6]	52.0 [42.3, 60.9]	16.7 [5.8, 29.7]
LR-BC	43.0 [28.4, 57.5]	51.4 [44.1, 59.0]	17.7 [5.8, 31.3]
BC	40.8 [26.9, 55.6]	43.2 [38.5, 47.8]	0.0 [0.0, 0.0]
Qwen-2.5-7B base	30.2 [19.8, 40.4]	34.8 [29.4, 40.5]	0.0 [0.0, 0.0]
Gemini 3.1 Pro agent	17.7 [6.0, 31.0]	36.2 [23.2, 49.1]	4.4 [0.0, 9.7]
Opus 4.5 agent	12.1 [5.7, 19.7]	36.0 [30.4, 41.8]	0.7 [0.0, 2.2]
Qwen3-235B agent	11.9 [4.3, 20.6]	35.4 [30.0, 46.0]	0.0 [0.0, 0.0]
GPT-5.4 agent	10.4 [3.1, 18.9]	30.4 [20.9, 39.8]	2.9 [0.0, 8.7]

B.3. Reward component ablation and credit assignment

The per-trial reward (Eq. 1) attributes approval and failure to the specific trial that earned them rather than pooling them over a window. At the canonical first-line NSCLC fork, KEYNOTE-024 (the PD-L1-selected trial that succeeded) scores $+11.0$, a pivotal-approval credit plus attributed revenue, while CheckMate-026 (the all-comers trial that failed) scores -1.0: a clean separation assigned directly to the two decisions.

We decompose the outcome reward across its terms. Table 7 reports each term's mean contribution over the 881 windows, its correlation with the total reward, and a leave-one-out measure of how much dropping it perturbs the training weights. The reward is dominated by approval-linkage (correlation 0.98 with the total); the fixed revenue and failure coefficients move the weights little (dropping the failure penalty shifts the mean absolute weight by only 0.095). The reward's signal is therefore an approval-linkage signal, which is a program-level quantity, consistent with the task-hardness probe (state→reward $R^2 = 0.49$ versus decision→reward $R^2 = 0.025$). This is why we do not over-claim a finely tuned reward: the reward's strength comes from grounding it in real approval outcomes.

Table 7. Reward component leave-one-out over the 881 saved window rewards. "corr" is the correlation of each term with the total reward; the last column is the mean absolute change in training weight when the term is dropped. Approval-linkage dominates.

Component	mean	corr with total	$\overline{\lvert\Delta w\rvert}$ on drop
approval reward	$+2.87$	$+0.98$	0.358
revenue reward	$+0.63$	$+0.21$	0.215
failure penalty	$+0.18$	-0.24	0.095

The IQL arm is robust to its hyperparameters: the per-example weights it induces are rank-correlated at ≥ 0.81 across the discount γ and expectile τ we swept, and the inverse-temperature β is essentially inert over the same range. Together with the component ablation above, this means neither the reward terms nor the value-model settings need fine tuning to obtain the reported behaviour.

B.4. Per-dimension paired F1

Per-dimension F1 for each structured trial field on the post-August 2025 subset (n=24).

No single system dominates every field. The offline arms (especially IQL) lead on the fields that define the strategic decision: indication, phase, masking, comparator, and primary endpoint. The Gemini 3.1 Pro agent, by contrast, is competitive or best on several categorical design fields (study type, strategy, region, enrollment, and the free-text trial description). This is consistent with the headline story: the agents can often get individual design choices right, but only the per-trial-reward arms assemble them into the correct whole decision, which is why the agents trail sharply on Strict F1 despite matching individual dimensions here.

Table 8. Per-dimension F1 on the post-August 2025 subset (n=24). Bold marks the best system per field. No single system dominates: the offline arms lead on indication, phase, masking, comparator, and primary endpoint, while the Gemini 3.1 Pro agent is competitive or best on several categorical design fields.

Field	IQL	RW-BC	LR-BC	BC	Qwen-2.5-7B	GPT-5.4	Gemini 3.1 Pro	Opus 4.5	Qwen3-235B
indication	**63.0**	62.1	58.6	56.2	48.6	19.4	38.5	31.1	21.9
trial description	40.7	34.5	41.4	40.6	40.0	47.2	**53.8**	46.7	37.5
phase	**51.9**	48.3	44.8	15.6	11.4	30.6	46.2	26.7	31.2
strategy	40.7	31.0	41.4	12.5	22.9	30.6	**46.2**	26.7	25.0
study type	77.8	69.0	65.5	53.1	65.7	69.4	**88.5**	68.9	78.1
allocation	59.3	62.1	**75.9**	50.0	40.0	52.8	69.2	35.6	43.8
masking	**85.2**	82.8	72.4	75.0	51.4	77.8	65.4	68.9	68.8
n_arms	40.7	41.4	41.4	**53.1**	22.9	27.8	50.0	26.7	34.4
enrollment	40.7	37.9	31.0	31.2	37.1	36.1	**50.0**	42.2	25.0
region	70.4	72.4	62.1	53.1	54.3	50.0	**80.8**	44.4	46.9
primary endpoint	**44.4**	37.9	37.9	34.4	20.0	27.8	15.4	22.2	12.5
secondary endpoints	88.9	89.7	**100.0**	87.5	88.6	91.7	92.3	71.1	84.4
comparator	**77.8**	75.9	55.2	65.6	0.0	50.0	30.8	22.2	15.6
combo partners	37.0	**41.4**	27.6	31.2	28.6	25.0	26.9	17.8	34.4

B.5. Design accuracy conditional on indication

To isolate "when a model proposes the right trial, how well does it get the design?" we score per-field accuracy on indication-matched pairs only, without a volume penalty, on the contamination-controlled temporal split. Table 9 reports phase accuracy and mean design-field accuracy. Conditional on a correct indication, the offline-RL arms predict the design substantially better than the base model (IQL 53.9% versus 34.7%; LR-BC leads on phase at 57.1%), so the learnable signal is genuinely in the design. The GPT-5.4 agent's design is competent when it identifies the right trial (44.8%), but it matches the correct indication far less often, so its deficit is decision targeting, not design knowledge. Cell counts are small (27 to 43 matched pairs) and absolute design accuracy plateaus near 50% even for the best arm, consistent with the associational ceiling.

Table 9. Per-field accuracy conditional on an indication match, on the post-September 2024 temporal split. DESIGN is the mean over the design fields; matched pairs is the conditioning sample size.

Arm	PHASE	DESIGN	matched pairs
Qwen-2.5-7B	25.6%	34.7%	43
GPT-5.4 agent	22.2%	44.8%	27
BC	14.0%	37.8%	43
RW-BC	47.2%	46.7%	36
LR-BC	**57.1%**	48.1%	28
IQL	41.9%	**53.9%**	43

B.6. Precision and recall decomposition of Indication F1

The headline tables report Indication F1, the harmonic mean of precision and recall over indication matches. Decomposing it, alongside mean predicted and ground-truth portfolio sizes, exposes a behavioural difference the F1 number alone hides (Table 10).

Table 10. Indication-level precision, recall, and F1 on the post-August 2025 subset (n=24), with the mean predicted ($\bar{n}_{\mathrm{pred}}$) and ground-truth ($\bar{n}_{\mathrm{gt}}$) portfolio sizes per window. Bold marks the best method per column.

Method	Precision	Recall	F1	$\bar{n}_{\mathrm{pred}}$	$\bar{n}_{\mathrm{gt}}$
Qwen-2.5-7B base	21.7%	53.3%	30.2%	3.4	1.5
GPT-5.4 agent	8.2%	16.7%	10.4%	3.0	1.9
Gemini 3.1 Pro agent	19.8%	21.8%	17.7%	2.7	2.3
Opus 4.5 agent	9.7%	25.4%	12.1%	6.2	2.1
Qwen3-235B agent	14.5%	16.0%	11.9%	2.4	2.2
BC	43.2%	47.9%	40.8%	2.2	1.7
RW-BC	44.9%	**53.6%**	45.9%	1.9	1.5
LR-BC	44.2%	50.4%	43.0%	1.9	1.5
IQL	**55.4%**	52.2%	**51.4%**	1.3	1.5

Two patterns are visible in Table 10. The tool agents over-predict: Opus 4.5 emits 6.2 trials per window and GPT-5.4 3.0, against roughly 1.5 in the ground truth, which pads recall but dilutes precision into the 8 to 20 percent range. The offline-trained models are calibrated in the opposite direction: IQL predicts only 1.3 trials per window yet reaches 55.4 percent precision, and the other models track the sponsor portfolio size closely. This decomposition explains why the agents look superficially competitive on indication overlap but collapse on Strict F1: a verbose portfolio rarely matches the full indication, phase, and strategy tuple on any single trial even when one of its many predictions hits the right disease.

B.7. Method Pareto over Indication F1 and Strict F1

Plotting Indication F1 against Strict F1 separates methods that get the disease right from methods that get the whole structured trial right. Figure 3 shows the methods on these two axes for the full held-out set.

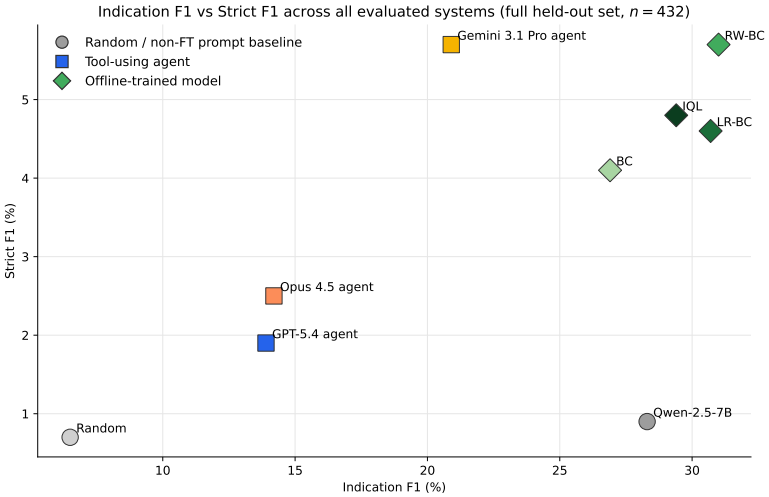

Figure 3. Indication F1 versus Strict F1 across methods on the full held-out set (n=432). Higher and more rightward is better. The offline-trained models (diamonds) occupy the upper-right, scoring well on both axes; the tool-using agents (squares) sit lower; the non-fine-tuned Qwen-2.5-7B reaches high Indication F1 but near-zero Strict F1 (it names the right disease without the matching structured trial), and the random baseline sits near the origin.

The offline-trained models occupy the upper-right, scoring highest on both axes. The contrast is sharpest against the non-fine-tuned Qwen-2.5-7B, which matches the offline models on Indication F1 (it names the right disease) yet collapses to near-zero Strict F1; the tool-using agents, by contrast, trail on Indication F1. The gain from offline training is therefore at the structured-trial level (indication, phase, and strategy correct together), not the indication level alone.

B.8. Reduced prompt

To isolate the contribution of the agent system prompt format from retrieval itself, we compare the state prompt against the soft prompt (the agent system prompt with an empty tool list). Neither variant receives retrieved evidence. Table 11 reports the comparison on GPT-5.4 and Qwen-2.5-7B, the two backbones for which we have complete coverage. On the full held-out set the state prompt is slightly stronger for both backbones, by roughly half a percentage point for GPT-5.4 and roughly two for Qwen. On the post-August 2025 contamination-controlled cell the comparison flips for GPT-5.4, where the soft prompt reaches 6.4 percent against the state prompt at 3.8 percent. Together these numbers indicate that the agent system prompt format on its own does not consistently lift performance over the state prompt, and the gains of the tool-using agent backbones reported in the headline therefore come from retrieval rather than from the prompt format.

Table 11. Indication F1 under the state prompt versus the soft prompt (agent system prompt with an empty tool list). Neither variant receives retrieved evidence.

Backbone	Subset	Full state	Soft prompt	Δ
GPT-5.4	Full set (n=432)	15.1%	14.8%	-0.3
GPT-5.4	Post-cutoff (n=24)	3.8%	6.4%	$+2.6$
Qwen-2.5-7B	Full set (n=432)	19.0%	16.8%	-2.2
Qwen-2.5-7B	Post-cutoff (n=24)	6.5%	4.2%	-2.3

B.9. Autoregressive evaluation

The autoregressive (AR) evaluation rolls out each method chronologically over the five single-drug holdouts (tagrisso, gleevec, imbruvica, keytruda, opdivo). At each window the method sees its own previous predictions in place of the real launch history; FDA approvals and revenue are dropped, since they are causally entangled with the method's own actions, while competitor activity is kept exogenous. We roll out the four offline-trained models (BC, RW-BC, LR-BC, IQL) and the Qwen-2.5-7B base model, scoring static and AR. The full three-dimension results are in Table 3: on Field-match % the fine-tuned arms are robust and IQL drops least ($47.1 \rightarrow 46.5\%$), while the Qwen-2.5-7B base degrades on Indication F1 ($30.0 \rightarrow 24.3\%$).