Offline Policy Learning for Clinical-Trial Strategy

Anonymous Authors[1]

Abstract

Clinical development is sequential decision-making under uncertainty. We study this setting by framing oncology clinical development as an offline decision-making problem in which models predict 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 behavioral cloning, reward-weighted behavioral cloning, and learned-reward training against four frontier LLM agents that share a common date-gated retrieval scaffold across held-out drug, sponsor, drug-class, and temporal splits. Adapters trained offline outperform every non-fine-tuned baseline. In the post-August 2025 contamination-clean holdout, offline training reaches 39.9% Indication F1 against 11.2% for the strongest tool agent, suggesting that structured offline learning can capture clinical-development strategy beyond memorized trial records.

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 try 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 offline policy learning over structured trial portfolios.
- We construct a temporal oncology dataset of 881 decision 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 three offline training objectives against

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Submitted to the ICML 2026 Workshop on Decision-Making from Offline Datasets to Online Adaptation: Black-Box Optimization to Reinforcement Learning. Do not distribute.

four frontier LLM agents on a shared retrieval scaffold and a hybrid that composes agent retrieval with the offline policy, 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 as a proxy for patient reach, 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 three 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 objectives differ only in how examples are weighted during training.

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 program-window pair receives an observed outcome score defined as

$$R_{p,t} = 2.0\, n_{\text{approvals}} + 1.0\, \log(1 + \text{revenue}_{p,t}) + 0.5\, \log(1 + \text{beneficiaries}_{p,t}) - 0.3\, n_{\text{failed}}. \quad (1)$$

The terms capture whether trials were later cited in FDA

approval packages, subsequent product revenue, CMS utilization data as a proxy for patient reach, and failed or terminated trials. This reward acts as a retrospective label constructed from observed regulatory, commercial, utilization, and trial-registry sources. We convert the normalized reward into a clipped sampling weight $w_{p,t} = \mathrm{clip}(0.5 + \tilde{R}_{p,t}, 0.1, 3.0)$ and train on the resulting rebalanced dataset. This preserves the behavioral-cloning objective, but ensures higher-reward windows are sampled more often during fine-tuning.

The third objective is learned-reward-weighted behavioral cloning. We trained a small reward model from outcome-ranked state-action pairs. Each (s, a) pair is mapped to a 31-dimensional feature vector covering program maturity, indication breadth, competitive pressure, epidemiology, commercial signals, prior outcomes, and properties of the proposed trial portfolio. The reward model $r_\theta(s, a)$ is trained to score higher-outcome examples above lower-outcome examples using an outcome-gap-weighted margin loss:

$$\mathcal{L}_{\mathrm{rank}} = \mathbb{E}_{(x^+, x^-)\sim\mathcal{P}}\Big[(R^+ - R^-) \max\big(0,\, m - r_\theta(x^+) + r_\theta(x^-)\big)\Big], \tag{2}$$

where $x = (s, a)$ and $m = 1.0$. The learned scores are applied to all training examples, standardized, converted into weights using a softmax with temperature 1.0, and clipped before fine-tuning.

2.5. Baselines and evaluation

We evaluate the three fine-tuned adapters against random baselines; prompt-only LLM baselines on Qwen-2.5-7B-Instruct (same backbone as the adapters) and GPT-5.4. All systems return the same 14-field action schema. The prompt-only LLM baselines are run with either a minimal prompt containing only the drug identifier and decision date, or a full state prompt used by the fine-tuned adapters. The full state prompt is a structured summary of the program, its history, and wider context at the decision date (Appendix B.6). The fine-tuned adapters are trained and evaluated only with the full state prompt. We additionally evaluate four tool-using agent backbones (GPT-5.4, Claude Opus 4.5, Gemini 3.1 Pro, and Qwen3-235B-Thinking-2507) on a shared scaffold. Each receives the minimal prompt 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.

Each predicted portfolio is scored against the held-out ground-truth portfolio by greedy one-to-one pairing on the indication and trial-description strings. Strings are lowercased, stripped of stage and severity prefixes, and normalized with a hand-curated oncology dictionary. A predicted trial is paired with its best-matching unpaired ground-truth trial; the pair is counted as an *indication match* when the normalized indication strings are equal or one is a substring of the other.

We report four 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 launches into the right disease and biomarker setting. *Soft Dim* averages, over paired trials, the fraction of the remaining 13 fields that match (categorical and ordinal fields by exact equality, secondary endpoints by set overlap, combination partners and comparators by a normalized-substring rule), with unpaired predictions contributing zero. *Strict F1* is a stricter version of Indication F1 in which a pair counts only if indication, phase, *and* strategy all match within the paired trial; it measures whole-trial alignment, not just disease-area alignment.

To probe robustness to compounding state error, we additionally evaluate the offline-trained adapters, Qwen-2.5-7B full, GPT-5.4 full, 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 Indication F1 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.

To probe whether agent retrieval and offline policy learning compose, we run a hybrid evaluation in which the GPT-5.4 agent's full tool-call transcript for each window is serialized as a retrieved evidence block, appended to the same full state prompt the adapters were trained on, and decoded by the held-out adapter for that split under the same JSON-schema constraint.

3. Results and Discussion

Figure 1 shows that the offline-trained adapters outperform every baseline on both the full held-out set and the contamination-free post-August 2025 cutoff subset, where RW-BC reaches 39.9% Indication F1 against the best agent at 11.2% (GPT-5.4). The collapse in all baseline models' performance in this setting suggests that they were primarily relying on contextual knowledge from pre-training when it is available, whereas the fine-tuned models learn a decision mapping from the offline episodes. Agentic systems outperformed their one-shot counterparts in the prospective subset as well. GPT-5.4 tool agent reaches 11.2% Indication F1,

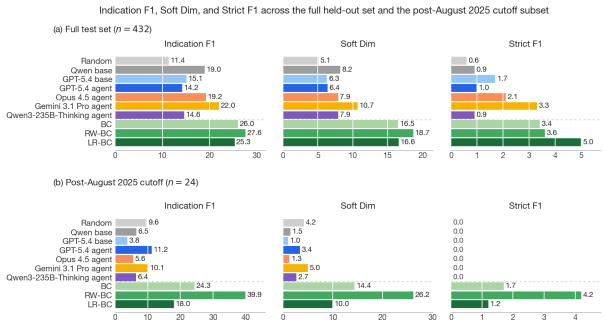

Figure 1. Results are reported as windows-weighted means across the eight held-out evaluation splits. [a] is the union of all n=432 held-out windows, while [b] is the cross-split subset of n=24 windows whose decision date falls after August 2025, the most recent knowledge cutoff for the models evaluated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

Table 1. Indication F1 by holdout split.

		Tool agents				Offline-trained adapters		
Split	n	GPT-5.4	Opus 4.5	Gemini 3.1	Qwen3-235B	BC	RW-BC	LR-BC
tagrisso	25	28.2	29.8	**55.1**	27.6	39.0	49.3	47.7
gleevec	35	14.8	16.8	25.4	12.4	34.3	**46.4**	42.8
imbruvica	19	5.3	5.5	7.6	10.3	17.1	**18.4**	13.7
keytruda	26	8.5	11.7	18.3	9.4	22.2	**22.7**	15.7
opdivo	31	14.4	**25.1**	23.1	11.2	21.2	22.1	18.1
AstraZeneca sponsor	96	17.4	19.7	19.8	16.5	**30.8**	24.5	24.5
Drug class (PD-1/L1)	155	12.8	22.6	22.2	15.4	19.4	**25.5**	23.9
post-Sep 2024 (Qwen 2.5 cutoff)	45	10.7	8.3	12.6	9.5	**34.1**	25.9	20.8
† post-Aug 2025 (all-model cutoff)	24	11.2	5.6	10.1	6.4	24.3	**39.9**	18.0

† Subset of the post-Sep 2024; not included into the 432-window total.

roughly 3× its prompt-only full-state counterpart at 3.8% (Qwen-2.5-7B showed a similar trend with 14.1% vs 6.5%).

The absolute values across all models show that the task is difficult. Even the best models recover only a minority of held-out indications and a small fraction of exact indication, phase, and strategy triples. 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 three offline objectives do not rank in one fixed order across holdouts. The training objective changes the failure mode, and more work is needed to identify which objective is best suited to a generalized setting.

Static evaluation gives each model the true historical launch history at every window. Autoregressive evaluation instead feeds each model its own previous predictions into later windows, testing sensitivity to compounding errors. Among the offline-trained adapters, LR-BC is most robust, dropping only 2.3 points from static to AR and reaching the highest AR Indication F1 at 26.5%, consistent with learned-reward training producing a policy that is less brittle when conditioned on its own imperfect history. GPT-5.4 with the full state prompt improves under AR (+7.1), most plausibly because it has memorised the drug's launch sequences and can generate a consistent sequence of key trials, while real history also contains opportunistic and one-off trials that the model struggles to reconstruct purely from its weights.

Table 2. Static versus autoregressive Indication F1 across the five single-drug holdouts (n=136).

Method	Static	AR	Δ
Qwen-2.5-7B full	18.2%	14.6%	−3.6
GPT-5.4 full	16.3%	23.4%	+7.1
GPT-5.4 agent	11.7%	13.4%	+1.7
BC	27.5%	23.7%	−3.8
RW-BC	**33.0%**	24.6%	−8.4
LR-BC	28.8%	**26.5%**	−2.3

A naive offline-to-online composition was evaluated on the post-August 2025 cell (n=24). Hybrid BC degrades to 21.0% (pure 24.3%), hybrid RW-BC is essentially unchanged (39.6% vs 39.9%), and hybrid LR-BC improves to 21.1% (pure 18.0%). Retrieval did not lift our strongest offline policy on the contamination-clean subset, but it helped the learned-reward variant, suggesting that evidence access and policy learning are separable.

This work has several limitations. First, the rewards are retrospective associations, not causal estimates of the value of a trial-launch decision. Approval, revenue, and patient reach depend on drug quality, sponsor resources, pricing, competition, and other factors not fully captured in the public state. The post-August 2025 cutoff evaluation set is small (n=24), so it is a useful comparison but not a definitive estimate of prospective performance. Finally, the schema developed ensures comparable model outputs, but it also abstracts away many details that matter in real trial design, including definitions and operational feasibility. This study should therefore be interpreted as evidence that historical development data can contain a learnable strategic signal, not as a claim that the learned policy is optimal.

These results motivate hybrid systems in which retrieval builds richer state representations while offline-trained policies select actions. Natural offline-to-online directions include training the offline objective on retrieval-augmented prompts using already-collected agent transcripts, folding new sponsor decisions back into the offline base as fresh episodes are observed, and hierarchical offline RL with subgoal models for long horizons and sparse delayed rewards (Shin & Kim, 2023; Levine et al., 2020; Prudencio et al., 2023). Clinical development strategy remains a difficult sequential decision-making problem that frontier models alone do not yet solve.

Code and data availability

Code and the dataset will be released publicly following review.

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.

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 adapters. 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 3, and an example portfolio entry is shown in Listing 2.

Table 3. 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 SFT, 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:

$$w_{p,t} = \mathrm{clip}\Big(0.5 + \tilde{R}_{p,t},\, 0.1,\, 3.0\Big)\,. \tag{3}$$

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 at context level L2 using `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, each (s, a) pair is mapped to a 31-dimensional state-action feature vector. The state features cover program maturity, indication breadth, competitive pressure, epidemiology, commercial signals, and outcome

history. The action features cover portfolio size, phase mix, novelty mix, and indication diversity.

The reward model $r_\theta(s, a)$ is a two-layer MLP with 128 hidden units, ReLU activations, and dropout 0.1. Examples are partitioned at the median observed reward into high- and low-outcome sets. We sample $2 \cdot \min(|H|, |L|)$ ranked pairs and minimize:

$$\mathcal{L}_{\text{rank}} = \mathbb{E}_{(x^+, x^-) \sim \mathcal{P}} \Big[(R^+ - R^-) \max\big(0,\, m - r_\theta(x^+) + r_\theta(x^-)\big) \Big], \tag{4}$$

where $x = (s, a)$ and $m = 1.0$.

The reward model is trained with Adam, learning rate 10^{-3}, weight decay 0.01, 200 epochs, batch size 32, and gradient clipping at 1.0. Learned scores are applied to all training examples, standardized, converted into weights using a softmax with temperature 1.0, and clipped to $[0.3, 3.0]$.

A.8. 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.9. Metric implementation

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. A pair is counted as an indication match if the normalized strings are equal or if either is a substring of the other.

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. The soft dimension score averages the fraction 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 Soft Dim and Strict F1 to test increasingly structured agreement, and report a stricter token-Jaccard sensitivity analysis in Appendix A.10.

A.10. Jaccard sensitivity analysis

Substring matching can over-credit verbose indication strings that contain the ground-truth indication. We therefore compute a stricter token-Jaccard indication score over normalized indication strings:

$$J(y, \hat{y}) = \frac{|\text{tok}(y) \cap \text{tok}(\hat{y})|}{|\text{tok}(y) \cup \text{tok}(\hat{y})|}. \tag{5}$$

The tokenizer lowercases each indication string and removes non-discriminative oncology terms such as cancer, tumor,

carcinoma, disease, and neoplasm. The headline substring metric and the token-Jaccard alternative correlate at $r = 0.83$ across the 40 split-by-baseline cells. The Jaccard metric is on average 16 percentage points lower and reverses the best-baseline winner on 4 of 8 splits. The largest observed gap occurs for tagrisso GPT-5.4 L3, where the lenient match score is 70.4 percent and the Jaccard score is 24.1 percent.

A.11. Reproducibility and release

The random seed is fixed for dataset splitting, reward-weighted oversampling, reward-model pair sampling, and fine-tuning. Each adapter 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-adapter wall-clock time is approximately 13 to 30 minutes depending on split size. The full sweep contains 24 adapters from 8 splits and 3 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 4) and is reported as "Random" in the main paper.

Table 4. 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	Soft Dim	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. Per-dimension paired F1

Paired F1 per structured trial-design field, full held-out set (n=432). Numbers are weighted means across the eight evaluation splits; for the offline-trained adapters, each method is the windows-weighted mean across its eight per-holdout adapters.

Table 5. Per-dimension paired F1 on the full held-out set (n=432). Bold marks the best method per row. Offline-trained adapters lift every dimension; combination partners is the one field where mode-random beats the offline-trained methods, an artefact of the high modal combination frequency in oncology. The indication row matches Table 1. Numbers reproducible from the canonical metrics dumps (`scripts/build_canonical_results.py`).

Dimension	Random	Qwen-2.5-7B full	GPT-5.4	BC	RW-BC	LR-BC
indication	11.4	19.0	15.1	26.0	**27.6**	25.3
phase	2.4	3.8	4.8	9.3	**10.5**	10.2
strategy	0.9	4.9	4.0	7.5	9.0	**9.2**
study type	1.8	14.9	11.7	18.2	**20.6**	19.3
allocation	0.8	5.4	4.9	9.8	10.4	**10.9**
masking	2.0	13.9	11.2	21.8	**22.9**	20.6
n_arms	0.7	6.7	5.5	10.3	9.1	**10.8**
enrollment	1.1	6.0	4.7	**9.9**	9.7	8.8
combo partners	**9.3**	5.1	4.5	5.1	4.0	4.0
comparator	9.3	2.4	5.3	**17.4**	15.6	16.4
region	1.1	9.4	8.0	13.2	**16.1**	12.9
primary endpoint	0.7	2.1	3.0	5.2	**6.2**	6.1
secondary endpoints	10.3	2.8	5.0	14.4	**15.1**	11.8

BC tends to win surface-level fields where the training labels admit a clear convention, including masking, comparator, region, and allocation. LR-BC wins fields with cross-dimensional structure such as strategy, study type, n_arms, and enrollment, consistent with its smooth learned reward picking up correlations between fields. RW-BC dominates the endpoint-type fields where outcome-weighted training emphasises trials that succeeded on those endpoints. Combination partners is the only field where Random beats the offline-trained methods. The modal multi-drug combination is so common in oncology that mode-random gets credit for every multi-arm trial while the structured methods spread predictions across more partners. GPT-5.4 *full* sits below Qwen-2.5-7B full on most dimensions despite being a frontier model. Consistent with the headline story, the structured-output training signal is what closes the gap on this task rather than raw language-model capability.

Figure 3 restates Table 5 as a grouped bar chart, ordered by best per-dimension F1.

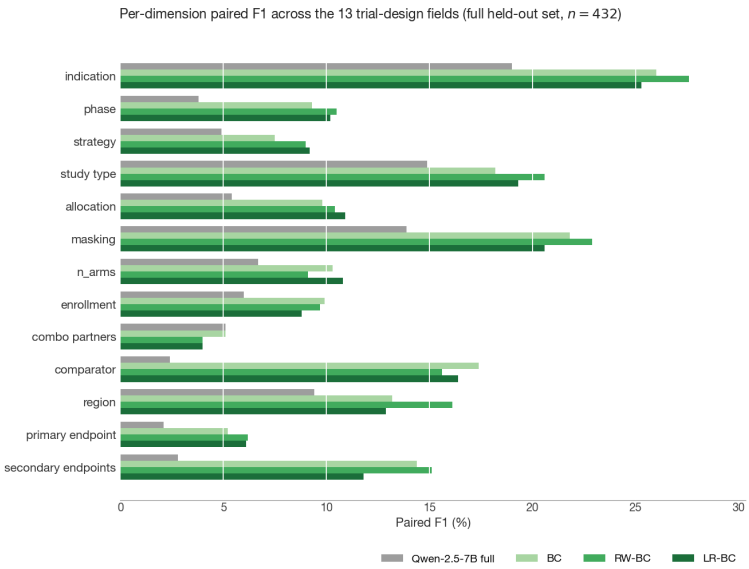

Figure 3. Per-dimension paired F1 on the full held-out set (n=432), ordered by best per-dimension F1. Visual restatement of Table 5.

B.3. Precision and recall decomposition of Indication F1

The headline tables report Indication F1, which is the harmonic mean of precision and recall over indication matches between predicted and ground-truth trials. Decomposing this into precision and recall, alongside mean predicted and ground-truth portfolio sizes, exposes a behavioural difference between methods that the F1 number alone hides. Table 6 reports the breakdown on the full held-out set.

Table 6. Indication-level precision, recall, and F1 on the full held-out set (n=432), with the mean predicted ($\bar{n}_{\text{pred}}$) and ground-truth ($\bar{n}_{\text{gt}}$) portfolio sizes per window. Values are windows-weighted means across the eight evaluation splits. Bold marks the best method per column.

Method	Precision	Recall	F1	$\bar{n}_{\text{pred}}$	$\bar{n}_{\text{gt}}$
Random	14.1%	11.5%	11.4%	2.00	2.88
Qwen-2.5-7B full	20.6%	22.9%	19.0%	2.91	2.88
GPT-5.4 full	14.2%	21.2%	15.1%	4.25	2.88
GPT-5.4 agent	10.8%	17.5%	11.6%	5.27	2.88
BC	33.4%	**31.4%**	26.0%	2.82	2.88
RW-BC	**38.3%**	28.6%	**27.6%**	2.05	2.88
LR-BC	32.9%	28.2%	25.3%	2.33	2.88

Two patterns are visible in Table 6. The tool agents emit between 4.25 and 5.27 trials per window against 2.88 in the ground truth, so they over-predict by a factor of roughly 1.5 to 2. Their verbose portfolios pad recall to between 17.5 and 21.2 percent while diluting precision to between 10.8 and 14.2 percent. The offline-trained adapters move in the opposite direction. RW-BC predicts only 2.05 trials per window, falling below the ground truth, but reaches 38.3 percent precision. BC predicts 2.82 trials per window, which is closest to the sponsor portfolio size, and is balanced between precision (33.4 percent) and recall (31.4 percent). This decomposition explains why agents look superficially competitive on Indication F1 but collapse on Strict F1 in the headline figure. A verbose portfolio is unlikely to match the ground-truth indication, phase, and strategy tuple on any single trial even when one of its many predictions hits the right disease.

B.4. 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 4 shows the methods on these two axes for the full held-out set.

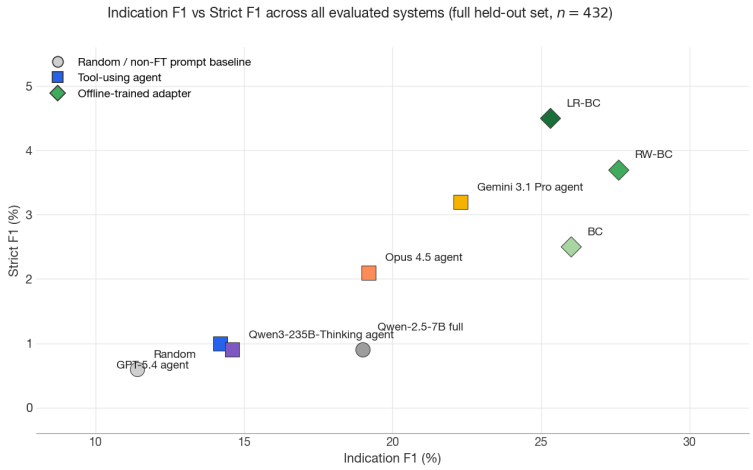

Figure 4. Indication F1 versus Strict F1 across methods on the full held-out set (n=432). Higher and more rightward is better. The offline-trained adapters cluster in the upper-right region while non-fine-tuned baselines cluster near the origin.

The offline-trained cluster sits several percentage points above the non-fine-tuned cluster on Strict F1 while remaining roughly 1.5× ahead on Indication F1. The implication is that the gain from offline training is not just at the indication level

("predicted the right disease") but at the structured-trial level ("predicted indication, phase, and strategy correctly together").

B.5. Indication F1 by holdout split (full table including baselines)

Table 7 extends the main per-split breakdown by adding the prompt-only baselines (Random, Qwen-2.5-7B full, and GPT-5.4 full) as additional columns. The eight split rows sum to n=432, and the post-August 2025 row is the contamination-clean subset of the temporal split (n=24) reported in Table 1 of the main paper.

Table 7. Indication F1 by holdout split, full table. Anchors are Random (mode), Qwen-2.5-7B full, and GPT-5.4 full. Tool agents are GPT-5.4, Opus 4.5, Gemini 3.1 Pro, and Qwen3-235B-Thinking. Adapters are BC, RW-BC, and LR-BC. Bold marks the best method per row.

		Anchors			Tool agents				Adapters		
Split	n	Random	Qwen	GPT-5.4	GPT-5.4	Opus 4.5	Gemini 3.1	Qwen3-235B	BC	RW-BC	LR-BC
tagrisso	25	**69.8**	27.9	25.6	28.2	29.8	55.1	27.6	39.0	49.3	47.7
gleevec	35	0.0	19.5	19.8	14.8	16.8	25.4	12.4	34.3	**46.4**	42.8
imbruvica	19	0.0	13.1	5.7	5.3	5.5	7.6	10.3	17.1	**18.4**	13.7
keytruda	26	32.2	12.5	8.0	8.5	11.7	18.3	9.4	22.2	**22.7**	15.7
opdivo	31	27.3	17.0	18.5	14.4	**25.1**	23.1	11.2	21.2	22.1	18.1
AstraZeneca sponsor	96	2.2	20.0	17.8	17.4	19.7	19.8	16.5	**30.8**	24.5	24.5
Drug class (PD-1/L1)	155	0.0	18.7	13.9	12.8	22.6	22.2	15.4	19.4	**25.5**	23.9
post-Sep 2024 (Qwen 2.5 cutoff)	45	19.3	20.6	9.8	10.7	8.3	12.6	9.5	**34.1**	25.9	20.8
† post-Aug 2025 (all-model cutoff)	24	9.6	6.5	3.8	11.2	5.6	10.1	6.4	24.3	**39.9**	18.0
Weighted mean (8 splits)	432	11.4	19.0	15.1	14.2	19.2	22.0	14.6	26.0	**27.6**	25.3

† Strict sub-slice of the post-Sep 2024 row, not summed into the 432-window total. Reported as the post-cutoff column of Table 1 in the main paper.

Figure 5 restates the adapter and baseline columns of Table 7 as a grouped bar chart, with the four tool agents omitted for legibility.

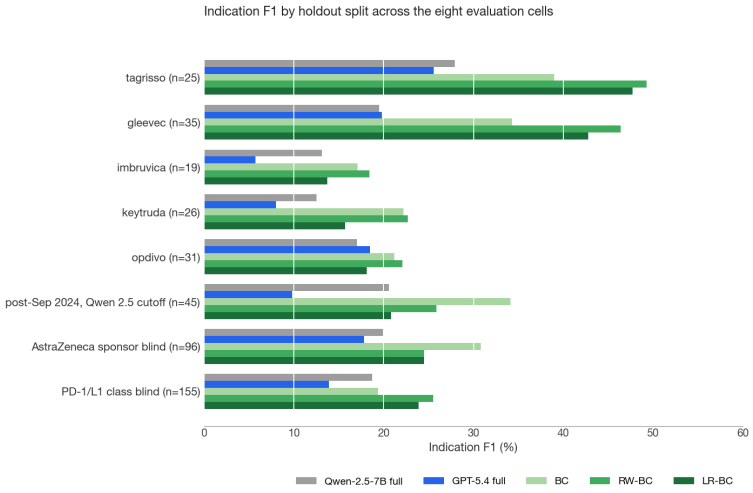

Figure 5. Indication F1 by holdout split for Qwen-2.5-7B full, GPT-5.4 full, and the three offline-trained adapters. Numbers match Table 7. Tool agents are omitted for legibility and reported in the table.

B.6. Full state prompt

The full 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 adapters at training and inference time, and to the prompt-only LLM baselines that are run in "full" mode.

B.7. Effect of the soft prompt

To isolate the contribution of the agent system prompt format from retrieval itself, we compare the full state prompt against the soft prompt (the agent system prompt with an empty tool list). Neither variant receives retrieved evidence. Table 8 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 full 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-clean cell the comparison flips for GPT-5.4, where the soft prompt reaches 6.4 percent against the full 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 full 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 8. Indication F1 under the full 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.8. Autoregressive evaluation

The autoregressive (AR) evaluation rolls out each agent chronologically on the five drug holdouts (tagrisso, gleevec, imbruvica, keytruda, opdivo, 136 decision windows in total). At each window the agent sees its own previous predictions in place of the real launch history. FDA approvals and revenue are dropped because they are causally entangled with the agent's own actions, while competitor activity is kept exogenous. We roll out the three offline-trained adapters (BC, RW-BC, LR-BC), Qwen-2.5-7B full, GPT-5.4 full, and the GPT-5.4 tool agent. Qwen-2.5-7B full and GPT-5.4 full receive the same L2 state prompt as in the static evaluation.

Figure 6 plots Indication F1 against rollout depth, smoothed over a five-window rolling mean. LR-BC and GPT-5.4 full hold a flat band of roughly 25 to 30 percent across rollout depth, RW-BC spikes high early and degrades, and BC tracks LR-BC closely. The GPT-5.4 tool agent sits clearly below the offline adapters across rollout depth, consistent with its window-weighted AR score of 13.4 percent reported in Table 2 of the main paper.

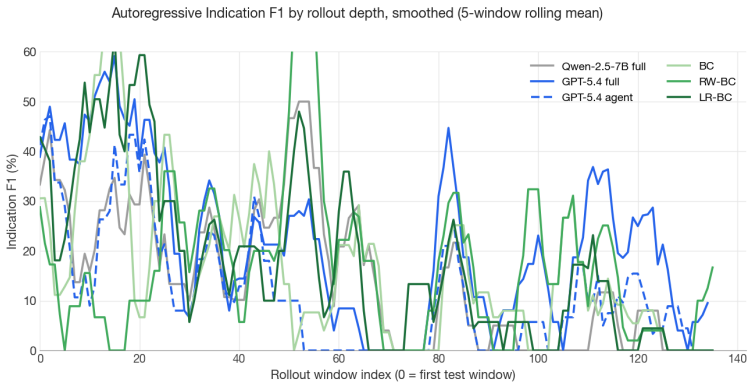

Figure 6. Autoregressive Indication F1 by rollout depth, smoothed over a five-window rolling mean.

Figure 7 aggregates the same rollouts per drug, showing that the relative method ordering depends on which drug is held out.

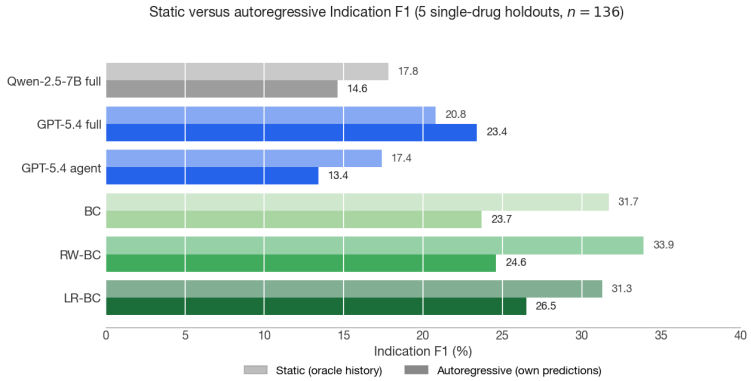

Figure 7. Autoregressive Indication F1 per drug. Each method's bar is the mean across that drug's rollout windows.

Restricting AR rollouts to post-cutoff decision windows yields n=10 post-Qwen and n=5 post-August 2025. These subsets are too small for reliable per-method comparison and are reported here only for completeness. On the n=5 post-August 2025 AR subset, RW-BC reaches 50% Indication F1, BC 28%, GPT-5.4 full 13%, with Qwen-2.5-7B full and LR-BC at 0%.