Natural-Language-Guided Generator-Agnostic Shortlisting for Protein Binder Design

Anonymous Authors[1]

Abstract

Modern de novo design workflows generate many candidate protein binders, but wet-lab validation capacity remains limited, making shortlisting a major bottleneck. We study whether LLMs can generate interpretable multi-metric ranking policies from precomputed structural-confidence and interface-quality proxy scores. Rather than reproducing any one design pipeline's internal filtering stack, we evaluate a post-generation task in which candidates are already generated and only the final top-K shortlist is chosen. A global `gpt-4o-2024-11-20` policy averaging strategy inferred from development targets slightly outperforms the strongest fixed heuristic, and target-conditioned iterative policy averaging improves further on the 9-target held-out split in target-averaged $Recall$@10 (0.562 vs. 0.506). On a smaller 2-target subset (Nipah (Adaptyv Bio, 2026) and RBX1 (GEM Workshop & Adaptyv Bio, 2026)), applying the same iterative strategy with `gpt-5.5` also exceeds the fixed AF2 and Boltz-2 rules in $Recall$@10 (0.456). These results suggest that natural-language-guided shortlisting can produce interpretable feature-weighted ranking rules for new binder-design candidate pools.

1. Introduction

Recent de novo protein binder pipelines couple generative backbone models such as RFdiffusion (Watson et al., 2023), ProteinMPNN-style sequence designers (Dauparas et al., 2022), and structure-prediction filters based on AlphaFold2 (Jumper et al., 2021; Evans et al., 2021) or Boltz-2 (Passaro et al., 2025). These advances have made large-scale candidate generation increasingly routine, but experimental validation capacity remains limited. As a result, shortlisting has become a central determinant of how efficiently generated binders are converted into validated hits (Bennett et al., 2023; Pacesa et al., 2025; Adaptyv Bio, 2026).

A common practice is to prioritize or filter candidates using fixed thresholds, single-score rankings, or hand-tuned metric combinations over structure-prediction confidence scores and interface-centric proxies (Bennett et al., 2023; Pacesa et al., 2025; Adaptyv Bio, 2026). BindCraft's fixed AF2/Rosetta filter set and Adaptyv's Boltz-2 ipSAE-based computational selection are representative examples (Pacesa et al., 2025; Adaptyv Bio, 2026). These workflow-specific filters are important for candidate curation, but they do not eliminate the final selection problem. After designs have been generated, filtered, or collected from different workflows, only a small number can be experimentally tested. A recent meta-analysis of 3,766 experimentally tested de novo binders reports that interface-focused confidence metrics such as ipSAE and orthogonal physicochemical descriptors can improve binder selection, while predictive performance still varies substantially by target (Overath et al., 2025). Together, these observations motivate a post-generation shortlisting setting that combines complementary proxy scores while testing whether the ranking rule should be global or conditioned on the target pool.

To study this setting, we separate shortlisting from candidate binder generation and treat it as a post-generation decision problem. For each target protein, we fix the generated candidate pool and a common 17-feature panel of precomputed proxy scores before shortlisting. The panel combines model-native confidence scores from AF2-Multimer, Boltz-2, and Protenix with interface-level proxy descriptors computed from predicted complexes. We use *policy* broadly to denote any deterministic shortlisting rule that maps candidate proxy scores to a ranked or selected subset. We compare fixed heuristics, supervised machine learning (ML) baselines, and large language model (LLM)-based shortlisting methods that generate either a single global policy for all held-out targets or a separate target-conditioned policy for each held-out target.

Our contributions are threefold:

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

Submitted to the 2026 Workshop on Generative and Agentic AI for Biology (ICML 2026). Do not distribute.

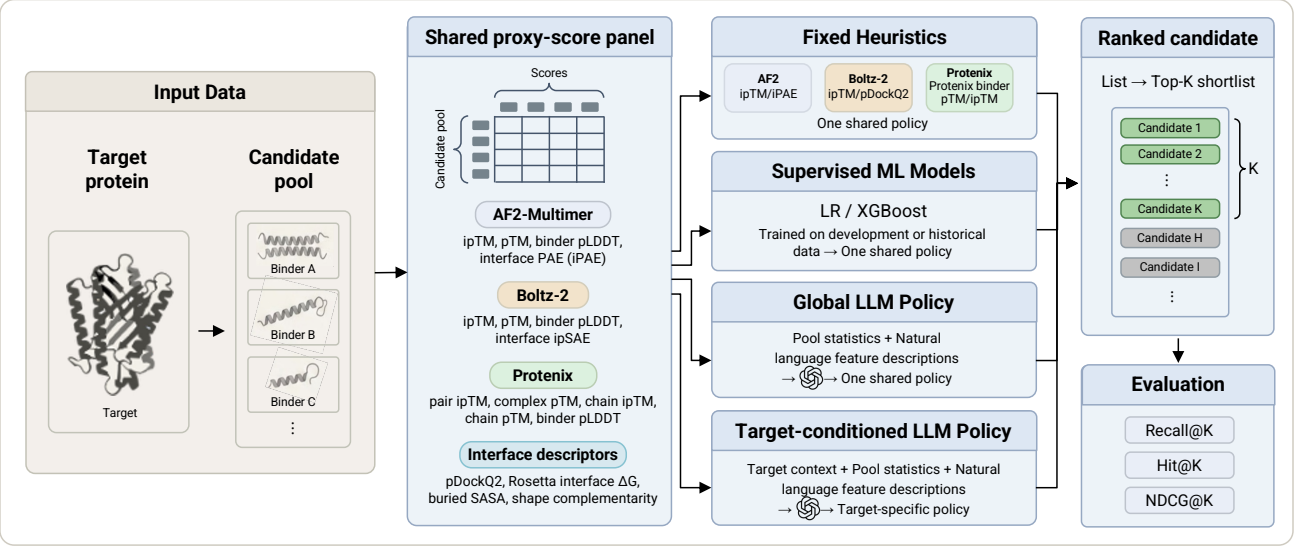

Figure 1. Overview of the post-generation binder shortlisting task. For each held-out target protein, heterogeneous upstream generators provide a candidate binder pool. All methods use the same extracted proxy-score panel from AF2-Multimer, Boltz-2, Rosetta, and Protenix/PXDesign, with target-MSA depth retained as target-level metadata. Fixed heuristics, supervised models, and global LLM policies apply one rule across held-out targets, whereas target-conditioned LLM policies emit one ranking policy per target from pool statistics and feature descriptions. Experimental labels are used only for evaluation of the top-10 shortlist.

- **A standardized post-generation shortlisting setup.** We align candidate pools from independent binder-design workflows to a common 17-feature proxy panel and evaluate fixed heuristics, supervised models, and LLM policies under the same top-K recall protocol.
- **LLM policies improve shortlisting over strong baselines.** Target-conditioned iterative policy averaging with `gpt-4o-2024-11-20` reaches 0.562 Recall@10 on the 9-target held-out split, above the strongest fixed heuristic, Boltz-2 pDockQ2 ranking, at 0.506 Recall@10.
- **Interpretable target-conditioned ranking rules.** We evaluate both global LLM-generated rules and per-pool target-conditioned rules. The target-conditioned form supports pool-specific adjustment of feature subsets and weights using each pool's score distributions, while preserving shortlisting logic that researchers can inspect, compare, and modify.

2. Related Work

Binder pipelines and candidate selection. Modern binder-design workflows often combine RFdiffusion backbone generation (Watson et al., 2023), ProteinMPNN-style sequence design (Dauparas et al., 2022), and AF2-based structural validation (Bennett et al., 2023). BindCraft (Pacesa et al., 2025) provides an automated AF2-backpropagation-based design pipeline with fixed AF2-confidence, Rosetta, and interface-composition filters. Recent studies suggest that AF3-derived scoring can improve binder selection. A 3,766-binder meta-analysis reports that ipSAE-based scores outperform common interface-confidence metrics and that Rosetta-derived descriptors provide complementary signal (Overath et al., 2025). AF3Score improves designed-binder screening when combined with AF2-derived methods (Liu et al., 2025). Adaptyv's Nipah release used Boltz-2 ipSAE ranking together with community voting and expert curation to select designs for experimental validation (Adaptyv Bio, 2026). These works motivate our use of interface-confidence, interface-geometry, and physicochemical proxy metrics, while leaving open how best to combine them for a given target.

Target-aware binder design. Cao et al. (2022) generate binders from target structure alone. Gainza et al. (2023) learn surface fingerprints to parameterize interaction design. APPRAISE (Ding et al., 2024) ranks engineered proteins by target-binding propensity through structure modeling. These works focus on generation or pairwise compatibility scoring. Our work differs in task: given a generated pool and precomputed proxy scores, we synthesize a policy to select K candidates as a separate decision problem.

LLMs and agents for protein design. ProtAgents (Ghafarollahi & Buehler, 2024) frames protein discovery as a multi-agent collaboration among LLM-backed roles that can retrieve knowledge, analyze structures, and call physics or machine-learning tools. ProteinCrow (Ponnapati et al., 2025) similarly builds an agentic protein-design assistant around curated tools, structural inputs, literature, and bio-

Table 1. Evaluation split after target-level merging, exact sequence de-duplication, and exclusion of zero-positive targets from Recall@10 analysis. Subtotal and total rows summarize the source rows above; they are not additional datasets.

Split	Source	Target pools	Designs	Binders	Targets
Development	BoltzGen	AMBP, HNMT, IDI2, IL7RA, insulin, MZB1/PERP1, PDGFR, PDL1, PHYH, PMVK, RFK	327	103	11
Held-out evaluation	BindCraft1 revalidation	spCas9, Der f 21, IFNAR2, Der f 7	76	31	4
	Merged EGFR	Adaptyv R1, Adaptyv R2, BindCraft1 revalidation	605	68	1
	pMHC minibinders	NY1, SILSY1	137	3	2
	Nipah release	Nipah G	1,030	103	1
	GEM/Adaptyv	RBX1	321	9	1
		Held-out subtotal	**2,169**	**214**	**9**
		Overall total (development + held-out)	**2,496**	**317**	**20**

chemical context. More broadly, Lee et al. (2025) review language-model use in protein design, including sequence modeling, context-conditioned design, and structure integration. Our use of LLMs is complementary: we apply them at the post-generation decision layer, where they propose ranking policies for constructing final shortlists from already-generated candidate pools.

3. Problem Setup

Shortlisting as policy synthesis. For a target protein t, we are given a generated candidate binder pool $\mathcal{C}_t$ of n_t designs. Each candidate $c \in \mathcal{C}_t$ has a fixed proxy-score vector $\mathbf{x}_{t,c} \in \mathbb{R}^m$ computed before shortlisting, where m is the number of common features available for every candidate. The task is to choose a subset $S_t \subseteq \mathcal{C}_t$ of size K that maximizes recall of experimentally verified binders under the validation budget. A method therefore outputs a ranking policy $\pi_t \in \Pi$, and a deterministic executor scores every candidate by π_t and returns the top K. In this formulation, ordinary metric-based ranking is a special case: sorting by one score, such as ipSAE_min, is a one-feature policy. Policy synthesis generalizes this by choosing which proxy scores to combine and how strongly to weight them for the target pool.

Policy space. We use the term *policy* for the structured triple (F, w, g) that specifies a ranking function. Here $F \subseteq \{f_1, \ldots, f_m\}$ selects a subset of features, $w \in \{1, 2, 3\}^{|F|}$ assigns integer weights, and each selected feature has a predefined higher-is-better or lower-is-better direction. In the main LLM policy space, g is a weighted normalized sum: the executor normalizes every selected feature within the target pool, computes the aggregate score, sorts candidates in descending order, and returns the top K.

4. Datasets

4.1. Sources and split

We collected labeled binder-design data from seven public source or workflow releases. We chose recent held-out releases so that their experimental labels postdate 2023-10-01, the documented knowledge cutoff of the primary `gpt-4o` experiment. A source is included only when it reports candidate-level experimental outcomes for tested designs, so that each target defines a retrospective shortlisting episode: the candidate pool is fixed, and every candidate has a binder/non-binder label.

We use two target-disjoint splits (Table 1). The *development* split contains 11 BoltzGen targets. The **9-target held-out evaluation split** contains targets from independent workflows and public validation releases; labels from this split are used only for final evaluation. Within the held-out split, we separately report the **2-target subset**, consisting of Nipah and RBX1, whose labels appeared after 2025-12-01, the documented knowledge cutoff for `gpt-5.5`. When the same biological target appears in multiple releases, we merge the corresponding pools and de-duplicate exact candidate amino-acid sequences; EGFR, for example, combines Adaptyv R1, Adaptyv R2, and BindCraft1 revalidation into 605 unique designs. Full preprocessing details, including binding-outcome parsing, source-specific target assignment, zero-positive target handling, and sequence de-duplication, are provided in Appendix A.

4.2. Feature extraction

Each candidate has 17 active proxy scores selected from the full extraction catalog (Table 2):

Boltz-2 pDockQ2 and ipSAE are post-processed interface-quality or interface-confidence proxies computed from Boltz-2 predicted complexes and confidence outputs; pDockQ2 follows Zhu et al. (2023), and ipSAE follows the PAE-based interprotein scoring approach of Dunbrack Jr (2025). Rosetta InterfaceAnalyzer metrics are computed on Boltz-2 target-side-MSA complexes and used as geometry, burial, and energy proxy scores, not ground-truth binding energies. Protenix/PXDesign features provide complex, binder-chain, and pairwise binder-target confidence; Protenix ipTM-style scores are not assumed to be numerically calibrated to AF2-Multimer or Boltz-2 ipTM. All evaluated candidates have non-missing values for the 17 active features. Appendix C gives the concrete feature columns,

Table 2. Candidate-level proxy score panel (17 active features). Exact feature columns, directions, and extraction details are in Appendix C.

Family	Features	Description
AF2-Multimer (Evans et al., 2021; Mirdita et al., 2022)	ipTM, pTM, binder pLDDT, interface PAE	Model-native complex confidence, binder local confidence, and interface PAE; lower interface PAE is better.
Boltz-2 (Passaro et al., 2025)	ipTM, pTM, binder pLDDT, ipSAE	Model-native confidence scores and ipSAE-style interface confidence (Dunbrack Jr, 2025) from the best target-MSA complex prediction.
Protenix/PXDesign (Team et al., 2025a;b)	pair ipTM, complex pTM, binder ipTM, binder pTM, binder pLDDT	Protenix complex, binder-chain, and binder-target confidence fields; binder metrics use the binder-chain index in our two-chain inputs.
Interface descriptors (Zhu et al., 2023; Stranges & Kuhlman, 2013; Lawrence & Colman, 1993)	Boltz-2 pDockQ2, Rosetta interface ΔG, buried SASA, shape complementarity	pDockQ2 is computed from the predicted complex and confidence outputs; Rosetta descriptors are computed on predicted complexes, not experimental structures.

monotonic directions, and extraction settings.

Inference settings. Reported features use target-side MSA complex predictions. For each target, we precompute one MMseqs2 MSA against UniRef30 + ColabFoldDB and reuse it for all candidate binders; the de novo binder chain is kept single-sequence because designed binders have no natural homologs. We run AF2-Multimer with 3 models and 3 recycles, and Boltz-2 with 5 diffusion samples, 3 recycling steps, and 200 sampling steps. Rosetta InterfaceAnalyzer is applied to Boltz-2 target-side-MSA predicted complexes. Templates are disabled in all complex-prediction runs. Additional MSA-generation details and per-target depths are reported in Appendix D.

5. Experimental Setup

5.1. Shortlisting methods

Global and target-conditioned policies. Some shortlisting methods emit one rule for all held-out targets, while others emit a separate rule for each target pool. We call the former *global*: one feature-weighted rule is used unchanged for every held-out target. We call the latter *target-conditioned*: a policy is generated separately for each held-out target and may choose different feature subsets or weights for different candidate pools. In both cases, the final scoring executor is deterministic; only the policy-generation step differs.

Fixed ranking heuristics. We evaluate target-agnostic fixed rules as single-score references spanning the main structure-prediction signals used for binder triage: AF2-Multimer ipTM and interface PAE (lower is better) (Evans et al., 2021; Mirdita et al., 2022), Boltz-2 ipTM and a post-processed interface-quality estimate (pDockQ2 computed from Boltz-2 predicted complexes) (Passaro et al., 2025; Zhu et al., 2023), and Protenix/PXDesign binder-chain confidence (binder pTM and binder ipTM) (Team et al., 2025a;b). Each rule ranks candidates within a target pool by one score only, testing how far a commonly used single metric can go before any learned or LLM-composed policy is introduced. The Protenix binder metrics are the binder-chain confidence fields used by the PXDesign Protenix filters, evaluated here as single-score ranking heuristics rather than hard thresholds.

Logistic regression and XGBoost. We evaluate two ML supervised baselines, neither of which sees held-out evaluation labels. The first fits L2-regularized logistic regression and XGBoost (Chen & Guestrin, 2016) on the 11-target BoltzGen development split using the same 17-feature panel as the LLM policies; these models test whether direct supervised learning over the feature panel is sufficient without target-conditioned policy synthesis. The second is a transfer baseline using Cao binder pools (Cao et al., 2022) with retrospective AF2 scores from Bennett et al. (2023). For this transfer setting, we use the available AF2 interaction-error and binder-confidence scores as the closest historical counterparts to our AF2 interface PAE and binder-chain pLDDT features. These transfer features come from the historical AF2 initial-guess scoring protocol rather than our AF2-Multimer target-side-MSA feature extraction, so they test cross-protocol transfer rather than a matched supervised re-training setting. Cao targets that overlap evaluation targets (EGFR, IL7Ra, and PDGFR) are removed before fitting. At evaluation time, each supervised model assigns every held-out candidate a fitted probability of being a binder, and candidates are ranked by this probability in descending order.

LLM policy sampling and averaging. We evaluate four LLM policy settings with `gpt-4o-2024-11-20` on the 9-target held-out evaluation split, and the corresponding settings with `gpt-5.5` on the 2-target subset. In the *global LLM* setting, the LLM sees only development-target feature descriptions, development-pool statistics, and development single-feature diagnostics, emits one global policy, and that policy is fixed before held-out evaluation. In the *global iterative LLM* setting, the final accepted policy from development-split iterative search is likewise fixed and applied unchanged to every held-out target. The *target-conditioned LLM* setting is a label-free test-time adaptation setting: it instantiates one prompt per held-out target and includes that target pool's identifier, represented source releases, natural-language feature descriptions, and score-distribution statistics. The *target-conditioned iterative LLM* setting adds accepted/rejected development-search feedback to the target-conditioned prompt. In all LLM settings, the model returns JSON policies that select 3 to 5

Table 3. Held-out evaluation ($K = 10$), grouped by method type. Hit denotes Precision@10. The 9-target split is the primary `gpt-4o` evaluation; the 2-target subset is Nipah and RBX1 and is used for the `gpt-5.5` analysis, with fixed and supervised baselines evaluated on the same subset for reference. Dashes mark model/split combinations not reported in the main comparison. LLM rows average five sampled policies. Global LLM policies are inferred from development targets only and fixed before held-out evaluation.

	9-target held-out			2-target held-out		
Method	Recall	Hit	NDCG	Recall	Hit	NDCG
Fixed ranking heuristics						
AF2 ipTM	0.419	0.333	0.388	0.350	0.350	0.393
AF2 interface PAE	0.408	0.267	0.321	0.200	0.200	0.204
Boltz-2 ipTM	0.414	0.289	0.372	0.050	0.050	0.037
Boltz-2 pDockQ2	0.506	0.311	0.362	0.300	0.300	0.339
Protenix binder pTM	0.368	0.222	0.335	0.206	0.200	0.352
Protenix binder ipTM	0.173	0.144	0.184	0.056	0.050	0.118
Supervised ML baselines						
LR-BG, 17 feat.	0.328	0.222	0.305	0.000	0.000	0.000
XGB-BG, 17 feat.	0.310	0.211	0.270	0.106	0.100	0.154
LR-Cao AF2	0.344	0.244	0.275	0.200	0.200	0.199
XGB-Cao AF2	0.408	0.278	0.349	0.250	0.250	0.294
LLM policies with policy averaging: `gpt-4o`						
Global LLM	0.514	0.344	0.414	–	–	–
Target-conditioned LLM	0.529	0.356	0.419	–	–	–
Global iterative LLM	0.529	0.356	0.428	–	–	–
Target-conditioned iterative LLM	**0.562**	**0.378**	**0.444**	–	–	–
LLM policies with policy averaging: `gpt-5.5`						
Global LLM	–	–	–	0.400	0.400	0.432
Target-conditioned LLM	–	–	–	0.400	0.400	0.433
Global iterative LLM	–	–	–	**0.456**	**0.450**	**0.471**
Target-conditioned iterative LLM	–	–	–	**0.456**	**0.450**	**0.471**

features and assign positive integer weights in $\{1, 2, 3\}$. The LLM does not choose feature directions: a deterministic executor applies fixed monotonic directions, rescales each selected feature to a 0 to 1 range within the target's candidate pool, reverses lower-is-better features so that larger normalized values are always better, computes a weighted score, and selects the top 10 designs. Appendix F gives the prompt schema, exact prompt templates, and a concrete target-conditioned prompt example for the 17-feature policy panel. The reported LLM results average five independently generated policies before ranking candidates, which reduces run-to-run variability without treating policy averaging as a separate shortlisting method.

5.2. Evaluation protocol

Protocol. $K = 10$ is fixed before evaluation. The primary metric is Recall@10 over verified binders, with denominator $\min(K, n_{\text{binders}})$ so that targets with fewer than 10 verified binders can still attain a maximum score of 1.0 by recovering all positives. Secondary metrics are Precision@10 and normalized discounted cumulative gain (NDCG@10). Precision@10 is the wet-lab hit rate among the 10 selected designs, whereas NDCG@10 measures whether verified binders are concentrated near the top of the shortlist; its ideal DCG is computed with the same $\min(K, n_{\text{binders}})$ number of positives for each target. Statistics are averaged across target pools rather than pooled across individual candidates.

Held-out evaluation. Held-out labels are never used during policy generation or ranking. Fixed heuristics, supervised baselines, global LLM policies, and global iterative LLM policies are fixed before held-out evaluation. Target-conditioned LLM policies additionally use unlabeled held-out pool statistics at policy-generation time, and are therefore reported as label-free test-time adaptation settings. The primary held-out evaluation uses 9 targets from multiple generator workflows and public validation releases (BindCraft, Adaptyv/GEM, pMHC, Nipah), so this split evaluates binder ranking for previously unseen target proteins under unseen-target, unseen-generator, and unseen-domain shifts. We separately report the 2-target subset for the `gpt-5.5` temporal-leakage analysis. The source-by-source temporal leakage audit, historical dataset handling, and remaining leakage caveats are provided in Appendix B and Section 7.

6. Results

Primary 9-target held-out performance. Table 3 reports the held-out evaluation with separate columns for the 9-target split and the 2-target subset. On the 9-target split, the strongest fixed rule is Boltz-2 pDockQ2 ranking, with Re-

Figure 2. Feature weights in target-conditioned iterative `gpt-4o` ranking policies over the 17-feature panel. Rows are held-out targets and columns are proxy-score features. Colored cells indicate features selected by the policies; the color scale shows the average weight across five sampled policies, meaning how much each score influenced the binder ranking for that target. Blank cells indicate features not included in the ranking policy.

call@10 = 0.506, Hit = 0.311, and NDCG@10 = 0.362. A global `gpt-4o` policy inferred only from development targets slightly improves all three metrics, reaching 0.514, 0.344, and 0.414, respectively. Target-conditioned single-turn policy averaging further improves Recall@10 to 0.529. The best overall performance is obtained by target-conditioned iterative `gpt-4o`, which reaches Recall@10 = 0.562, Hit = 0.378, and NDCG@10 = 0.444. Supervised baselines, whether trained on the development split or transferred from Cao AF2 data, do not match the strongest fixed rule or the best LLM-based setting.

2-target held-out subset. We separately evaluate Nipah and RBX1 because these are the held-out releases in our collection that appear after the documented knowledge cutoff of `gpt-5.5`. This is a smaller supporting analysis, not the primary held-out result. On this subset, iterative `gpt-5.5` reaches Recall@10 = 0.456, Hit = 0.450, and NDCG@10 = 0.471, above AF2 ipTM (0.350/0.350/0.393) and Boltz-2 pDockQ2 (0.300/0.300/0.339). The development-trained LR baseline selects no verified binders in the top 10 for either Nipah or RBX1, while XGBoost reaches only 0.106 Recall@10 and 0.100 Hit, illustrating limited supervised transfer to this small subset.

Generated LLM policy composition. The target-conditioned iterative `gpt-4o` policies concentrate weight on a small set of structure-confidence and interface-quality scores rather than spreading weight uniformly (Figure 2). Across the five policy samples for each of the 9 held-out targets, every policy selects AF2 ipTM and Rosetta shape complementarity, and 44 of 45 policies select Boltz-2 pDockQ2. Boltz-2 ipSAE appears in 24 policies, Rosetta buried SASA in 20, Boltz-2 ipTM in 14, Protenix pair ipTM in 10, and Boltz-2 binder pLDDT in 2. By total selected weight, the policies allocate 36.0% to Boltz-2 features, 33.3% to AF2 features, 26.8% to interface descriptors, and 3.9% to Protenix features. Thus, the best-performing LLM settings are not discovering unrelated rules for each target. Instead, they use a stable AF2/Boltz/Rosetta backbone and make modest target-conditioned adjustments through feature inclusion and weighting within this recurring policy family.

Example generated policies. Table 4 shows RBX1 examples, including two fixed one-feature rules and the averaged global and target-conditioned iterative policies used by the reported LLM settings. For LLM rows, coefficients are average weights across five sampled policies, with unselected features contributing 0. Positive terms reward higher raw feature values, while negative terms reward lower raw feature values.

7. Analysis and Discussion

Global versus target-conditioned policies. The global LLM uses one rule inferred from development-target information only, then applies that rule unchanged to all held-out targets. Its 9-target Recall@10 is 0.514, slightly above Boltz-2 pDockQ2. The target-conditioned single-turn `gpt-4o` policy reaches 0.529 Recall@10, suggesting that unlabeled target-pool statistics can help policy synthe-

Table 4. Example RBX1 ranking rules. Each $\hat{x}$ is a within-target 0 to 1 normalized feature; $+$ rewards higher raw values and $-$ rewards lower raw values. For LLM rows, coefficients are average policy weights across five sampled policies; unselected features contribute weight 0 to the average. Global policies use one averaged rule for every target, whereas target-conditioned policies are averaged for RBX1 specifically. Candidates are sorted by the resulting rank score in descending order, and the top K candidates are selected.

Policy	Rank score
Fixed AF2 ipTM, RBX1	$\hat{x}_{\text{AF2 ipTM}}$
Fixed Boltz-2 pDockQ2, RBX1	$\hat{x}_{\text{Boltz pDockQ2}}$
Global iterative `gpt-4o`	$3.0\hat{x}_{\text{AF2 ipTM}} + 2.0\hat{x}_{\text{Boltz pDockQ2}} + 2.0\hat{x}_{\text{Rosetta SC}} + 0.6\hat{x}_{\text{Boltz ipSAE}} + 0.4\hat{x}_{\text{Boltz ipTM}}$
Target-conditioned iterative `gpt-4o`, RBX1	$3.0\hat{x}_{\text{AF2 ipTM}} + 2.0\hat{x}_{\text{Boltz pDockQ2}} + 2.0\hat{x}_{\text{Rosetta SC}} + 0.8\hat{x}_{\text{Boltz ipTM}} + 0.6\hat{x}_{\text{Boltz ipSAE}} + 0.4\hat{x}_{\text{Rosetta buried SASA}} + 0.2\hat{x}_{\text{Protenix pair ipTM}}$
Global iterative `gpt-5.5`	$3.0\hat{x}_{\text{AF2 ipTM}} + 2.0\hat{x}_{\text{Rosetta SC}} + 1.4\hat{x}_{\text{Boltz pDockQ2}} + 0.6\hat{x}_{\text{Boltz ipTM}} + 0.4\hat{x}_{\text{Rosetta buried SASA}} + 0.4\hat{x}_{\text{Protenix pair ipTM}} - 0.2\hat{x}_{\text{AF2 interface PAE}}$
Target-conditioned iterative `gpt-5.5`, RBX1	$3.0\hat{x}_{\text{AF2 ipTM}} + 2.0\hat{x}_{\text{Rosetta SC}} + 1.4\hat{x}_{\text{Boltz pDockQ2}} + 0.6\hat{x}_{\text{Boltz ipTM}} + 0.4\hat{x}_{\text{Rosetta buried SASA}} + 0.4\hat{x}_{\text{Protenix pair ipTM}} - 0.2\hat{x}_{\text{AF2 interface PAE}}$

sis. After development-search feedback is added, target-conditioned iterative `gpt-4o` gives the strongest 9-target result across all three metrics. We therefore interpret target conditioning as useful when combined with policy averaging and development feedback, while noting that the selected rules still share a stable AF2/Boltz/Rosetta backbone across targets.

What iterative feedback adds. Iterative prompting adds development-search feedback: the prompt summarizes which feature-weighted policies were accepted or rejected on the development split and their aggregate metrics. This changes the LLM's role from producing a rule from development summaries alone to producing a rule calibrated by prior policy search. In the main 9-target evaluation, target-conditioned iterative `gpt-4o` improves Recall@10 from 0.529 for target-conditioned single-turn policy averaging to 0.562, and NDCG@10 from 0.419 to 0.444. In the 2-target post-cutoff subset, iterative `gpt-5.5` improves over the single global policy from 0.400 to 0.456 Recall@10 and from 0.432 to 0.471 NDCG@10.

Why pDockQ2 remains competitive. Boltz-2 pDockQ2 estimates interface quality from the predicted binder-target complex and its confidence outputs, without using an experimental or designed reference structure. Its strong performance is consistent with the fact that the available proxy panel is dominated by predicted-complex confidence and interface-quality signals. The LLM does not replace this signal; 44 of 45 target-conditioned iterative `gpt-4o` policies retain Boltz-2 pDockQ2 and combine it with AF2 ipTM and Rosetta shape complementarity.

Feature-family ablation of generated policies. To probe which selected feature families carry the generated policies, we removed each family from the saved target-conditioned iterative policies and re-ran the deterministic executor without making new LLM calls. This is a diagnostic over the five individual policies, so its baseline is the mean single-policy result rather than the score-level policy ensemble. For `gpt-4o` on the 9-target split, the mean single-policy Recall@10 is 0.508; removing Boltz-2 features reduces it to 0.412, removing interface descriptors reduces it to 0.430, and removing AF2 features reduces it to 0.461. Removing Protenix has a smaller effect, reducing Recall@10 to 0.486. The same diagnostic on the 2-target `gpt-5.5` policies shows the strongest dependence on AF2 features, dropping Recall@10 from 0.433 to 0.163 when AF2 is removed. These diagnostics support the interpretation that the generated policy family relies mainly on AF2, Boltz-2, and Rosetta-derived interface signals, with Protenix contributing occasionally rather than dominating the final rankings.

Why supervised baselines lag. The supervised baselines learn a single target-agnostic rule from limited or mismatched training data. BoltzGen-trained logistic regression and XGBoost use only 327 development examples and do not adapt to held-out target-pool statistics. The Cao/Bennett transfer baselines use aligned AF2-derived features, but differ from our AF2-Multimer target-MSA protocol. Thus, these baselines test useful transfer settings, but they lack the target-conditioned pool adaptation and development-feedback calibration used by the strongest LLM policies.

8. Conclusion

We cast post-generation binder shortlisting as a constrained policy-synthesis problem over a fixed 17-feature proxy-score panel and compared fixed heuristics, supervised baselines, and LLM-generated ranking policies. On the primary 9-target held-out split, Boltz-2 pDockQ2 is a strong fixed rule, reaching Recall@10 = 0.506. A global LLM policy

inferred from development targets improves this to 0.514, while target-conditioned iterative policy averaging gives the best row, with Recall@10 = 0.562.

The generated policies do not replace structure-prediction and interface-quality metrics. Instead, they combine AF2, Boltz-2, and Rosetta-derived signals into interpretable feature-weighted ranking rules that can be adjusted to each target pool. This suggests that natural-language-guided policy synthesis can serve as a lightweight decision layer for converting heterogeneous generated binder pools into experimentally actionable shortlists.

References

Adaptyv Bio. Nipah competition results. Proteinbase collection, 2026. URL https://proteinbase.com/collections/nipah-binder-competition-results. Experimental validation results released January 21, 2026.

Alford, R. F., Leaver-Fay, A., Jeliazkov, J. R., O'Meara, M. J., DiMaio, F. P., Park, H., Shapovalov, M. V., Renfrew, P. D., Mulligan, V. K., Kappel, K., et al. The rosetta all-atom energy function for macromolecular modeling and design. *Journal of chemical theory and computation*, 13 (6):3031–3048, 2017.

Bennett, N. R., Coventry, B., Goreshnik, I., Huang, B., Allen, A., Vafeados, D., Peng, Y. P., Dauparas, J., Baek, M., Stewart, L., et al. Improving de novo protein binder design with deep learning. *Nature Communications*, 14 (1):2625, 2023.

Cao, L., Coventry, B., Goreshnik, I., Huang, B., Sheffler, W., Park, J. S., Jude, K. M., Marković, I., Kadam, R. U., Verschueren, K. H., et al. Design of protein-binding proteins from the target structure alone. *Nature*, 605 (7910):551–560, 2022.

Chen, T. and Guestrin, C. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Dauparas, J., Anishchenko, I., Bennett, N., Bai, H., Ragotte, R. J., Milles, L. F., Wicky, B. I., Courbet, A., de Haas, R. J., Bethel, N., et al. Robust deep learning-based protein sequence design using proteinmpnn. *Science*, 378(6615): 49–56, 2022.

Ding, X., Chen, X., Sullivan, E. E., Shay, T. F., and Gradinaru, V. Fast, accurate ranking of engineered proteins by target-binding propensity using structure modeling. *Molecular Therapy*, 32(6):1687–1700, 2024.

Dunbrack Jr, R. L. Rēs ipsae loquunt: What's wrong with alphafold's iptm score and how to fix it. *bioRxiv*, 2025.

Evans, R., O'neill, M., Pritzel, A., Antropova, N., Senior, A., Green, T., Žídek, A., Bates, R., Blackwell, S., Yim, J., et al. Protein complex prediction with alphafold-multimer. *biorxiv*, pp. 2021–10, 2021.

Gainza, P., Wehrle, S., Van Hall-Beauvais, A., Marchand, A., Scheck, A., Harteveld, Z., Buckley, S., Ni, D., Tan, S., Sverrisson, F., et al. De novo design of protein interactions with learned surface fingerprints. *Nature*, 617 (7959):176–184, 2023.

GEM Workshop and Adaptyv Bio. GEM x Adaptyv: RBX1 binder design competition. Proteinbase competition page, 2026. URL https://proteinbase.com/competitions/gem-adaptyv-rbx1. Experimental validation results released April 26, 2026.

Ghafarollahi, A. and Buehler, M. J. Protagents: protein discovery via large language model multi-agent collaborations combining physics and machine learning. *Digital Discovery*, 3(7):1389–1409, 2024.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

Lawrence, M. C. and Colman, P. M. Shape complementarity at protein/protein interfaces, 1993.

Lee, J. S., Abdin, O., and Kim, P. M. Language models for protein design. *Current Opinion in Structural Biology*, 92:103027, 2025.

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637): 1123–1130, 2023.

Liu, Y., Yu, Q., Wang, D., and Chen, M. Af3score: A score-only adaptation of alphafold3 for biomolecular structure evaluation. *Journal of Chemical Information and Modeling*, 65(15):8207–8214, 2025.

Mirdita, M., Schütze, K., Moriwaki, Y., Heo, L., Ovchinnikov, S., and Steinegger, M. Colabfold: making protein folding accessible to all. *Nature methods*, 19(6):679–682, 2022.

Overath, M. D., Rygaard, A. S., Jacobsen, C. P., Brasas, V., Morell, O., Sormanni, P., and Jenkins, T. P. Predicting experimental success in de novo binder design: a meta-analysis of 3,766 experimentally characterised binders. *BioRxiv*, pp. 2025–08, 2025.

Pacesa, M., Nickel, L., Schellhaas, C., Schmidt, J., Pyatova, E., Kissling, L., Barendse, P., Choudhury, J., Kapoor, S., Alcaraz-Serna, A., et al. One-shot design of functional protein binders with bindcraft. *Nature*, 646(8084):483–492, 2025.

Passaro, S., Corso, G., Wohlwend, J., Reveiz, M., Thaler, S., Somnath, V. R., Getz, N., Portnoi, T., Roy, J., Stark, H., et al. Boltz-2: Towards accurate and efficient binding affinity prediction. *BioRxiv*, 2025.

Ponnapati, M., Cox, S., Gordon, C. W., Hammerling, M. J., Narayanan, S., Laurent, J. M., Braza, J. D., Hinks, M. M., Skarlinski, M. D., Rodriques, S. G., et al. Proteincrow: A language model agent that can design proteins. In *ICML 2025 Generative AI and Biology (GenBio) Workshop*, 2025.

Stark, H., Faltings, F., Choi, M., Xie, Y., Hur, E., O'Donnell, T., Bushuiev, A., Uçar, T., Passaro, S., Mao, W., et al. Boltzgen: Toward universal binder design. *bioRxiv*, pp. 2025–11, 2025.

Stranges, P. B. and Kuhlman, B. A comparison of successful and failed protein interface designs highlights the challenges of designing buried hydrogen bonds. *Protein Science*, 22(1):74–82, 2013.

Team, B. A. A., Chen, X., Zhang, Y., Lu, C., Ma, W., Guan, J., Gong, C., Yang, J., Zhang, H., Zhang, K., et al. Protenix-advancing structure prediction through a comprehensive alphafold3 reproduction. *BioRxiv*, pp. 2025–01, 2025a.

Team, P., Ren, M., Sun, J., Guan, J., Liu, C., Gong, C., Wang, Y., Wang, L., Cai, Q., Ma, W., et al. Pxdesign: Fast, modular, and accurate de novo design of protein binders. *bioRxiv*, pp. 2025–08, 2025b.

Watson, J. L., Juergens, D., Bennett, N. R., Trippe, B. L., Yim, J., Eisenach, H. E., Ahern, W., Borst, A. J., Ragotte, R. J., Milles, L. F., et al. De novo design of protein structure and function with rfdiffusion. *Nature*, 620(7976): 1089–1100, 2023.

Zhu, W., Shenoy, A., Kundrotas, P., and Elofsson, A. Evaluation of alphafold-multimer prediction on multi-chain protein complexes. *Bioinformatics*, 39(7):btad424, 2023.

A. Dataset Preprocessing

Candidate pools are defined at the target level before feature extraction. A design is included only when the public release provides a candidate amino-acid sequence, a target sequence, and a candidate-level experimental binding outcome. Entries without an explicit binding measurement are treated as unlabeled rather than as negatives and are excluded from recall-based evaluation; this criterion removes 7 Adaptyv EGFR round-1 submissions. For ProteinBase-style releases, binding outcomes are read from the release-provided evaluation records. A design is labeled positive if any intended-target binding record is positive and negative if all intended-target binding records are negative. Expression measurements and binding-strength annotations are retained as metadata, but they do not define the binary label.

Target identifiers are assigned according to the experimental assay target reported by each source. Single-target competitions, including Adaptyv EGFR R1/R2 and Nipah, use the competition target, and off-target or control assay records are not used for the binary intended-target label. BindCraft1 revalidation rows are assigned by their assay target in `evaluations`. For BoltzGen, PDB-like structural seed identifiers are mapped to the released biological assay target rather than used directly as target names; for example, `1g13`, `3apu`, `2a1x`, and `3qkg` correspond to GM2A, ORM2, PHYH, and AMBP, respectively.

Source-level pools are retained for audit and feature-extraction checks, including BoltzGen targets with no released positives. Targets with zero positives (GM2A, ORM2, and TNF-α) are excluded from recall-based evaluation because Recall@10 is undefined when the target-level positive denominator is zero. When multiple releases contain the same biological target, the evaluation pool merges those releases and de-duplicates exact candidate amino-acid sequences. If duplicate sequences have discordant labels, the merged label is positive if any duplicate record is positive. For EGFR, this merge combines Adaptyv R1, Adaptyv R2, and BindCraft1 revalidation into 605 unique candidate sequences from 615 labeled rows, with 68 positives after positive-if-any label aggregation.

B. Temporal Leakage Audit

Table 5 records the public-release dates used for the temporal-leakage argument. The primary 9-target analysis uses models whose documented knowledge cutoffs predate all held-out candidate-label releases. GPT-5.5 has a later 2025-12-01 cutoff, so only Nipah and RBX1 remain strictly post-cutoff for that model; those two targets define the separate GPT-5.5 check in Table 3.

C. Feature Glossary

The main policy panel contains 17 reproducible candidate-level features. Higher-is-better features are AF2-Multimer ipTM, AF2-Multimer pTM, AF2 binder pLDDT, Boltz-2 ipTM, Boltz-2 pTM, Boltz-2 binder pLDDT, Boltz-2 ipSAE, Boltz-2 pDockQ2, Protenix pair ipTM, Protenix complex pTM, Protenix binder ipTM, Protenix binder pTM, Protenix

Source / pool	Targets used	Public date	After 4o?	After GPT-5.5?
BindCraft1 revalidation	spcas9, derf21, ifnar2, derf7, EGFR rows	2024-10-01	yes	no
Adaptyv EGFR R1	EGFR	2024-10-18	yes	no
pMHC minibinders	pMHC_NY1, pMHC_SILSY1	2024-12-03	yes	no
Adaptyv EGFR R2	EGFR	2025-01-15	yes	no
BoltzGen release	development targets only	2025-11-24	yes	no
GEM/Adaptyv RBX1	RBX1	2026-04-26	yes	yes
Nipah release	nipah_g	2026-01-21	yes	yes
Cao/Bennett historical	transfer baseline only	2022 to 2023	no	no
RFdiffusion wet-lab historical	excluded from main held-out evaluation	2023	no	no

Table 5. Dataset-level temporal-leakage audit. The cutoff dates are 2023-10-01 for `gpt-4o-2024-11-20` and 2025-12-01 for GPT-5.5.

binder pLDDT, Rosetta buried SASA, and Rosetta shape complementarity. Lower-is-better features are AF2 interface PAE and Rosetta interface ΔG. We keep target-MSA depth only as target-level metadata.

Inference settings. AF2-Multimer and AF2-monomer runs use ColabFold v1.5.5 with three models, three recycles, and no custom templates. The single-sequence complex wrapper uses single-sequence MSA mode. The target-MSA complex wrapper supplies a precomputed multimer A3M for the target chain, while the binder chain remains single-sequence because no homologs exist for a de novo binder.

Boltz-2 runs use 5 diffusion samples, 3 recycling steps, 200 sampling steps, and the `boltz2` model, matching BoltzGen's de novo binder refolding configuration (Stark et al., 2025). In the single-sequence baseline, both chains are modeled without MSA information. In the target-MSA variant, the target chain receives the precomputed target MSA and the binder chain remains single-sequence.

Protenix runs use the public PXDesign Protenix setting: `protenix_base_default_v0.5.0`, one sample, two diffusion steps, four recycles, no templates, and MSA enabled. The target chain receives the precomputed target MSA and the binder chain remains single-sequence. Our Protenix JSON inputs place the binder chain first and the target chain second; therefore binder-chain fields use index 0. The active Protenix policy features include the pairwise binder-target ipTM entry `chain_pair_iptm[0][1]`, complex pTM, and binder-chain ipTM, pTM, and pLDDT fields.

The reported candidate tables use target-MSA Boltz-2, AF2-Multimer, Protenix, and Rosetta complex outputs. Single-sequence complex predictions are retained for ablation but are not part of the main 17-feature panel. ESMFold (Lin et al., 2023), using the HuggingFace `facebook/esmfold_v1` weights, is deterministic and run once per binder. Rosetta `InterfaceAnalyzerMover` (Stranges & Kuhlman, 2013) with the `ref2015` scorefunction (Alford et al., 2017) is applied to Boltz-2 predicted complexes after constrained pre-relaxation.

Column	Direction	Implementation
`af2_multimer_iptm_mean_tmsa`	higher	Mean AF2-Multimer ipTM across aggregated target-MSA complex models.
`af2_multimer_ptm_mean_tmsa`	higher	Mean AF2-Multimer pTM across aggregated target-MSA complex models.
`af2_multimer_binder_plddt_mean_mean_tmsa`	higher	Mean binder-chain AF2-Multimer pLDDT across aggregated target-MSA complex models.
`af2_multimer_interface_pae_mean_mean_tmsa`	lower	Mean cross-chain AF2-Multimer PAE over target-binder interface residue pairs, averaged across models.
`boltz2_multimer_iptm_best_tmsa`	higher	Boltz-2 ipTM from the best target-MSA complex sample.
`boltz2_multimer_ptm_best_tmsa`	higher	Boltz-2 pTM from the best target-MSA complex sample.
`boltz2_multimer_binder_plddt_mean_best_tmsa`	higher	Binder-chain pLDDT from the best Boltz-2 target-MSA complex sample.
`boltz2_multimer_ipsae_min_best_tmsa`	higher	Conservative ipSAE-style interface confidence from the best Boltz-2 target-MSA complex sample.
`boltz2_multimer_pdockq2_best_tmsa`	higher	pDockQ2 computed from the best Boltz-2 target-MSA predicted complex and confidence outputs.
`protenix_pxdesign_binder_target_iptm_best`	higher	Protenix pairwise binder-target ipTM from `chain_pair_iptm[0][1]` in the best sample.
`protenix_pxdesign_ptm_best`	higher	Protenix complex pTM in the best sample.
`protenix_pxdesign_binder_iptm_best`	higher	Protenix binder-chain ipTM from `chain_iptm[0]` in the best sample.
`protenix_pxdesign_binder_ptm_best`	higher	Protenix binder-chain pTM from `chain_ptm[0]` in the best sample.
`protenix_pxdesign_binder_plddt_best`	higher	Protenix binder-chain pLDDT from chain index 0 in the best sample.
`rosetta_interface_dG_tmsa`	lower	Rosetta InterfaceAnalyzer interface ΔG on the Boltz-2 target-MSA predicted complex.
`rosetta_dsasa_tmsa`	higher	Rosetta buried solvent-accessible surface area on the Boltz-2 target-MSA predicted complex.
`rosetta_sc_tmsa`	higher	Rosetta InterfaceAnalyzer shape complementarity (Lawrence & Colman, 1993) on the Boltz-2 target-MSA predicted complex.

Table 6. Concrete columns used in the 17-feature policy panel. The suffix `tmsa` denotes target-side MSA complex predictions; designed binder chains remain single-sequence.

D. Target MSA Depth

Hits per target after concatenating paired and unpaired MMseqs2 hits into a single target MSA. Hits include the query sequence itself. The 23 entries cover every distinct target sequence used for feature extraction or audit after dataset aliases sharing a target sequence are deduplicated; pMHC_NY1 splits into two length variants.

Target	Hits	Target	Hits
insulin	20,099	pdl1	2,917
egfr	16,555	3qkg	2,099
pdgfr	15,830	7aah	1,797
3apu	11,907	tnfalpha	1,784
2a1x	10,873	3ch4	1,530
pMHC_SILSY1	10,629	2pny	1,085
pMHC_NY1 (385aa var.)	10,540	il7ra	541
pMHC_NY1 (189aa var.)	9,200	1jqd	528
1nb0	8,975	ifnar2	409
derf21	8,964	nipah_g	28
spcas9	3,384	derf7	9
1g13	3,354		

Table 7. Target-MSA depth (number of hits) after concatenating UniRef and environmental hits. Allergen and viral glycoprotein targets are intentionally shallow.

E. Information Available by Method

This section summarizes the inputs available to each baseline family:

- **Supervised ML baselines**: logistic regression and XGBoost fit either on the 11 BoltzGen development targets with the same 17-feature panel, or on a historical Cao/Bennett AF2-confidence transfer matrix after removing targets that overlap the held-out split.
- **Global LLM**: one LLM call using development targets only. The prompt contains the 17 feature descriptions, development target-pool distribution statistics, and development single-feature diagnostics, and emits one weighted ranking policy that is fixed before held-out evaluation.
- **Global iterative LLM**: iterative policy search on development targets only. The model proposes candidate global policies, receives aggregate development-split feedback from previous proposals, and the final accepted policy is fixed before held-out evaluation.
- **Target-conditioned LLM**: one LLM call per test target. The prompt contains the target-pool identifier, represented source releases, the 17 feature descriptions, and target-pool distribution statistics for that target. It emits one target-conditioned weighted ranking policy with 3 to 5 selected features and integer weights in $\{1, 2, 3\}$.
- **Target-conditioned iterative LLM**: one LLM call per test target. The prompt contains the target-conditioned inputs plus a summary of accepted and rejected development-split policies from the iterative search logs, and emits one target-conditioned weighted ranking policy.

Cao/Bennett AF2 transfer baseline. The historical Cao/Bennett transfer baseline is trained on the AF2 confidence measurements that have compatible counterparts in the current held-out pools: binder-target interface PAE and binder-chain pLDDT. On the historical training pools, these are the AF2 interaction PAE and binder pLDDT scores reported with the Cao/Bennett retrospective scoring data. On the current held-out pools, we recompute the analogous quantities using our AF2-Multimer target-MSA protocol: mean cross-chain PAE over target-binder interface residue pairs and mean binder-chain pLDDT, each averaged across AF2-Multimer model runs. RMSD-based historical features are excluded because the current held-out candidates do not generally include the original designed-complex reference structures needed to compute the same designed-versus-predicted RMSD terms. Thus, this baseline tests transfer across compatible AF2 confidence feature types, not an identically matched AF2 scoring protocol.

F. Prompt Examples for the 17-feature Policy Panel

The reported target-conditioned LLM settings use natural-language feature descriptions and unlabeled target-pool statistics. Each held-out target receives the same instruction schema, but with its own feature-distribution summaries. Global settings instead use development-target summaries and do not receive held-out pool statistics. The prompt requires the model to choose 3 to 5 features and assign integer weights in $\{1, 2, 3\}$; feature directions, within-target normalization, score aggregation, and top-K selection are handled by the deterministic executor. Table 8 summarizes the target-conditioned prompt blocks, and Table 9 gives a concrete RBX1 target-conditioned prompt-statistics excerpt.

Prompt block	Contents
Task	Emit one weighted JSON ranking policy for a target-conditioned candidate pool.
Allowed action	Choose 3 to 5 features and assign each selected feature an integer weight in $\{1, 2, 3\}$.
Disallowed action	No hard filters, no thresholds, no affinity claims, no model-chosen directions, and no model-chosen aggregation.
Executor	Fixed feature directions, within-target normalization, weighted normalized sum, descending sort, top 10.
Inputs	Target-pool identifier, represented source releases, natural-language descriptions of the 17 features, and per-target min, quartiles, maximum, mean, and standard deviation.
Output schema	JSON object with brief reasoning plus a named list of selected feature-weight terms and a rationale.

Table 8. Schematic prompt used for target-conditioned LLM policies.

Target-conditioned prompt template. The target-conditioned LLM runs use the following instruction template, with the target identifier, represented datasets, and pool-level feature distributions filled in separately for each held-out target.

```
You are selecting 10 protein binder designs for wet-lab validation from one
target-conditioned candidate pool.

Task:
- Emit one weighted ranking policy as JSON.
- Choose 3 to 5 features and assign each selected feature an integer weight
  from 1 to 3.
- Select the smallest number of features within the allowed range that still
  captures complementary, non-redundant evidence.
- Do not use hard filters or thresholds.
- Do not assume any feature is a direct affinity measurement.
- Use only the feature list, distribution statistics, and any explicitly
  provided development examples below.
- First write a short high-level reasoning summary, then the final answer
  policy.
- Do not specify feature directions or aggregation. A deterministic executor
  will apply fixed feature directions, normalize features within the target
  pool, compute a weighted normalized sum, sort candidates by that aggregate
  score in descending order, and select the top 10 designs.

Allowed JSON schema:
{
  "reasoning": "brief high-level reasoning, not step-by-step hidden chain of thought",
  "answer": {
    "name": "short_name",
    "selected_terms": [
      {"feature": "feature_name", "weight": 3}
    ],
    "rationale": "brief rationale"
  }
}

Allowed weights: 1, 2, or 3 only.
Select only features whose direction is listed as higher or lower. Features
marked control or context-dependent are provided as context, but the executor
will not use them as ranking terms.
Do not add weak features just to make the policy look comprehensive; every
selected feature must provide complementary evidence.

Target pool:
- eval_target: {target_id}
- datasets represented: {datasets}

Feature source context:
- AF2-Multimer means AlphaFold2-Multimer, a protein complex structure-prediction model.
- Boltz-2 is a biomolecular complex structure-prediction model. In this feature
  table, Boltz-derived fields are either native confidence outputs or
  post-processed interface-quality proxies computed from Boltz-2 predicted
  complexes and associated confidence outputs.
- Protenix is an AlphaFold3-style structure-prediction model used here for
```

```
  PXDesign-inspired confidence scores.
- Rosetta InterfaceAnalyzer computes physics-inspired interface geometry,
  burial, hydrogen-bond, and energy-related metrics on the Boltz-2 predicted
  complex, not on an experimental structure.

Candidate features:
{the 17 allowed feature descriptions and fixed directions listed in
Table~\ref{tab:feature-columns}}

Pool-level feature distributions:
Total candidates: {n_candidates}
{per-feature min, quartiles, maximum, mean, and standard deviation}

Return only valid JSON.
```

Listing 1. Target-conditioned prompt template. Braced fields are filled separately for each held-out target pool.

Concrete RBX1 prompt-statistics excerpt. Table 9 shows an excerpt from the target-pool statistics block in one RBX1 target-conditioned prompt. The full prompt contains the same summary fields for all 17 active features.

Feature excerpt from the RBX1 target-conditioned prompt	Direction	Min	Q25	Median	Q75	Max	Mean	SD
af2_multimer_iptm_mean_tmsa	Higher	0.09	0.1567	0.1833	0.3267	0.9233	0.2881	0.2136
af2_multimer_interface_pae_mean_mean_tmsa	Lower	10.56	20.28	21.98	23.46	26.98	21.4	3.305
boltz2_multimer_ipsae_min_best_tmsa	Higher	0.01797	0.03585	0.04785	0.06829	0.2707	0.057	0.03368
boltz2_multimer_pdockq2_best_tmsa	Higher	0	0.8861	0.9768	0.9891	1.002	0.9187	0.1251
rosetta_sc_tmsa	Higher	0.4042	0.5782	0.6175	0.6568	0.7364	0.6149	0.05758
rosetta_dsasa_tmsa	Higher	631.9	1993	2445	3131	10570	2865	1610
protenix_pxdesign_binder_ptm_best	Higher	0.1189	0.7462	0.786	0.8011	0.8425	0.7428	0.1326
protenix_pxdesign_binder_target_iptm_best	Higher	0.7905	0.8201	0.8318	0.8383	0.8591	0.8296	0.01646

Table 9. Concrete target-conditioned prompt-statistics excerpt for the RBX1 held-out pool. The full prompt contains the target identifier, represented source release, total candidate count, all 17 feature descriptions, fixed monotonic directions, and pool-level feature distributions before requesting one JSON ranking policy.

Target-conditioned iterative prompt template. Target-conditioned iterative policy sampling reuses the target-conditioned prompt structure, but adds a development-feedback block before the held-out target statistics. The feedback block summarizes accepted and rejected policies from iterative search on the development split, including aggregate Recall@10/NDCG@10 values and the selected feature-weight terms. The LLM then emits a new policy for the current held-out target, not a single policy shared by all held-out targets.

```
You are selecting 10 protein binder designs for wet-lab validation from one
target-conditioned candidate pool.

You will propose one JSON ranking policy for the held-out target below. A
deterministic executor will score every candidate within that target pool and
select the top 10 designs. You may use the development-search feedback only to
calibrate which feature combinations have worked on training targets.

Rules:
- Use only the allowed 17 features listed below.
- Choose 3 to 5 features and assign each selected feature an integer weight
  from 1 to 3.
- Select complementary, non-redundant evidence from different proxy families
  when possible.
- Do not use hard filters.
- Do not assume any feature is a direct affinity measurement.
- Do not specify feature directions or aggregation. A deterministic executor
  will apply fixed feature directions, normalize features within each target
  pool, compute a weighted normalized sum, sort candidates by that aggregate
  score in descending order, and select the top 10 designs.
- First write a short high-level reasoning summary, then the final answer
  policy.

Allowed JSON schema:
{
  "reasoning": "brief high-level reasoning, not step-by-step hidden chain of thought",
  "answer": {
    "name": "short_name",
    "selected_terms": [
      {"feature": "feature_name", "weight": 3}
    ],
    "rationale": "brief rationale"
  }
}

Development-search feedback:
{accepted and rejected policies with aggregate development metrics}

Feature source context:
{same source context as the target-conditioned prompt}

Allowed features:
{same 17 feature descriptions and fixed directions as the target-conditioned prompt}

Held-out target pool:
- eval_target: {target}
- datasets represented: {datasets}
```

```
Held-out target feature distributions:
{per-feature min, quartiles, maximum, mean, and standard deviation}

Return only valid JSON.
```

Listing 2. Target-conditioned iterative prompt template. The development-search feedback block contains accepted and rejected policy summaries with aggregate development metrics, followed by the held-out target pool statistics.

G. Per-target Recall

Table 10 reports per-target Recall@10 for the strongest fixed heuristic and the primary policy-averaging LLM setting.

Target	n	n_b	Fixed	Iterative GPT-4o
spcas9	20	14	0.800	0.900
derf21	16	4	0.750	0.750
nipah_g	1030	103	0.600	0.800
ifnar2	20	8	0.500	0.750
pMHC_NY1	41	1	1.000	1.000
derf7	20	5	0.600	0.600
egfr	605	68	0.300	0.400
pMHC_SILSY1	96	2	0.000	0.000
rbx1	321	9	0.000	0.111
mean			0.506	**0.562**

Table 10. Per-target Recall@10. Fixed is Boltz-2 pDockQ2. Iterative GPT-4o is the policy-averaged target-conditioned iterative `gpt-4o-2024-11-20` setting reported in Table 3.