ANCHORZYME: ANCHOR-CONDITIONED DUAL-MEMORY ENZYME GENERATION FOR UNSEEN SUBSTRATES

Anonymous authors
Paper under double-blind review

ABSTRACT

Designing enzymes for substrates with no known natural catalyst is a central AI for Science problem because it could expand biocatalysis to new reactions in sustainable synthesis, drug and materials manufacturing, and environmental remediation; we study strict substrate-only zero-shot enzyme generation, where the input is only a small query molecule and the output is a candidate catalytic protein sequence. The main obstacle is an activity–structure trade-off: in this setting, edits that increase predicted enzyme–substrate compatibility often reduce foldability or structural confidence. We show that this trade-off is amplified when retrieved enzymes are used as a single undifferentiated prompt, because global scaffold similarity, which supports foldability, is mixed with local pocket and active-site signals, which support substrate recognition and catalysis. To separate these roles, we propose ANCHORZYME, a retrieval-augmented, anchor-conditioned editing framework that selects a scaffold anchor, retrieves donor enzymes, aligns them to define site, shell, and distal edit regions, trains a teacher-guided generator to make better-than-anchor edits, and combines a catalytic-gain discriminator with a frozen sequence-plausibility discriminator for sampling and reranking. On REALUNKNOWN-389, a curated benchmark of 389 currently enzyme-unknown substrates, ANCHORZYME achieves the best frozen-surrogate scores among substrate-only baselines: predicted $\log_{10}(k_{\mathrm{cat}})$ reaches 0.384 under UniKP, and sequence-based structural confidence reaches 0.770 under ESMFold. Relative to SENZ, the two scores improve from 0.310 to 0.384 and from 0.499 to 0.770, respectively, establishing scaffold-preserving local editing as a practical route toward discovery-oriented zero-shot enzyme design for substrates beyond currently known biocatalysis.Our code is available at `https://anonymous.4open.science/r/code-EEE1`.

1 INTRODUCTION

Enzymes provide selective and sustainable routes for chemical synthesis, pharmaceutical manufacturing, materials production, and environmental remediation (Bornscheuer et al., 2012; Sheldon & Woodley, 2018). Although enzyme engineering and directed evolution have made biocatalysts practical for many reactions (Arnold, 2017), the most discovery-oriented cases remain difficult: a new small-molecule substrate may have no known natural catalyst and no close enzymatic precedent. Recent advances in protein structure prediction and generation make it increasingly plausible to propose new proteins rather than only recognize natural ones (Jumper et al., 2021; Lin et al., 2023; Nijkamp et al., 2023; Dauparas et al., 2022). In biochemical terms, a substrate is the small molecule transformed in a reaction, while an enzyme is a protein whose amino-acid sequence folds into a three-dimensional structure that can accelerate that reaction. Our learning problem therefore maps a molecular query to a ranked set of candidate protein sequences; it is not molecule classification or retrieval of a known enzyme label. We therefore study strict substrate-only zero-shot enzyme generation, where the test-time input is only such a query molecule, without the correct EC label, a protein–ligand complex, reaction products, an active-site structure, or any verified enzyme. The goal is to propose enzyme candidates that are plausible for the substrate and structurally reliable enough for downstream experimental prioritization.

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

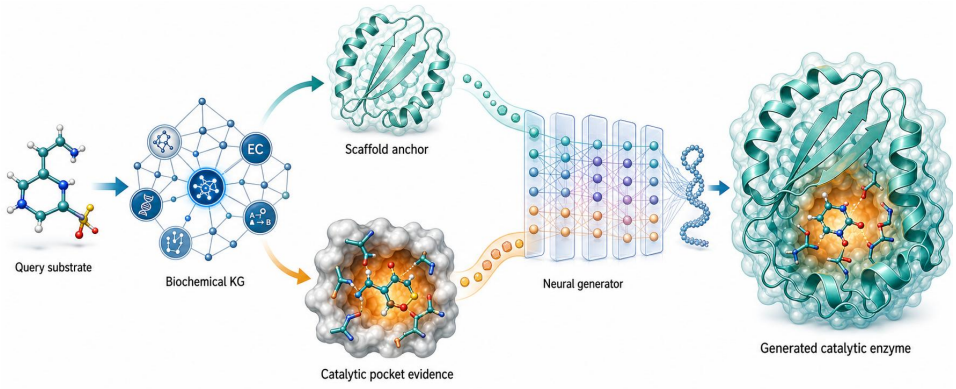

Figure 1: Principle schematic of ANCHORZYME. A query substrate indexes a biochemical KG, whose evidence is separated into a scaffold anchor and catalytic-pocket cues. A neural generator combines the preserved scaffold with local pocket evidence to produce a protein-like enzyme with an optimized catalytic pocket.

This setting is not directly addressed by existing protein generators. Generic sequence models provide broad protein priors (Ferruz et al., 2022; Nijkamp et al., 2023; Lin et al., 2023; Abramson et al., 2024), whereas function- or ligand-conditioned methods often require EC labels, functional sites, complexes, or inverse-folding inputs (Munsamy et al., 2022; Song et al., 2024; Yang et al., 2024; Dauparas et al., 2025; Krishna et al., 2024; Hua et al., 2024). In contrast, a biochemical knowledge graph (KG) represents molecules, catalytic records, protein families, functional annotations, and pathways as nodes connected by known biochemical relations. This form is closer to real discovery: it can support substrate-indexed retrieval without revealing a target enzyme. Yet retrieval is only useful if it can improve catalytic relevance without destroying protein plausibility. SENZ highlights this tension: better predicted enzyme–substrate compatibility can come with lower foldability or structural confidence (Du et al., 2025).

We argue that the tension arises not from KG retrieval itself, but from using the retrieved evidence too coarsely. Enzymes contain signals at different biological scales: a global scaffold supports family-level plausibility, foldability, and mutation tolerance, while pocket, active-site, shell, and distal residues drive substrate recognition and catalytic adaptation. For a truly unknown substrate, KG evidence is necessarily indirect; it can suggest relationally relevant enzymes, but it cannot certify one as the correct catalyst. If these candidates are consumed as a single mixed prompt, the generator has no clear way to decide what should be preserved as scaffold and what should be transferred as catalytic signal. This motivates our design principle: use KG-guided retrieval as a soft biochemical prior, choose one scaffold anchor, and perform substrate-conditioned local editing around that anchor.

Here, the *scaffold anchor* is one retrieved natural enzyme used as the full-length sequence that the editor starts from and largely preserves. *Catalytic donor memories* are other retrieved enzymes used only to provide residue-level alternatives at aligned positions. The name describes their editing role: they are not assumed to be verified catalysts for the query substrate.

Following this principle, ANCHORZYME treats KG retrieval as the beginning of the design process rather than as a direct answer. It first applies an inductive knowledge graph embedding (KGE) retriever over the biochemical KG and an enzyme-side sequence-aware reranker to form a substrate-specific candidate pool. A KGE model maps KG nodes and relation types to learned vectors and uses them to rank training enzymes that are relationally relevant to the query substrate. ANCHORZYME then separates this pool into a scaffold anchor and catalytic donor memories, aligns them in an anchor-conditioned frame, and edits site, shell, and distal regions using teacher positives scored by a frozen UniKP surrogate. During generation and reranking, a delta-catalytic discriminator encourages catalytic gain relative to the anchor, while a frozen sequence-plausibility discriminator penalizes edits that move away from protein-like sequence plausibility.

The same KG also grounds our evaluation. We use it to define REALUNKNOWN-389, a set of 389 currently enzyme-unknown substrates with no recorded catalytic protein in the construction snapshot, while auditing leakage from the original zero-shot splits. On this benchmark, ANCHORZYME improves over SENZ in both primary metrics, raising UniKP $\log_{10}(kcat)$ from 0.310 to 0.384 and

normalized pLDDT from 0.499 to 0.770. These results support scaffold-preserving local editing as a practical route toward discovery-oriented enzyme design for substrates beyond currently known biocatalysis.

The contributions of this work are four-fold:

- We construct a KG-grounded strict zero-shot discovery setting and define REALUNKNOWN-389, a curated set of 389 currently enzyme-unknown substrates.
- We identify the activity–structure trade-off as a consequence of mixing scaffold-level evidence with local catalytic cues in retrieved contexts.
- We propose ANCHORZYME, which separates inductive KGE retrieval, scaffold anchoring, catalytic donor transfer, local editing, and activity–structure-aware reranking.
- We show that ANCHORZYME improves both UniKP $\log_{10}(kcat)$ and normalized pLDDT over SENZ on REALUNKNOWN-389 under frozen evaluators.

2 RELATED WORK

Protein and enzyme generation. Protein generative models such as ProtGPT2 (Ferruz et al., 2022), ProGen2 (Nijkamp et al., 2023), ESM-2 (Lin et al., 2023), EvoDiff (Abramson et al., 2024), ProteinGAN (Repecka et al., 2021), and ProteinMPNN (Dauparas et al., 2022) provide strong priors for generating protein-like sequences or designing sequences for given structures. Enzyme- and ligand-aware methods further condition generation on EC labels, taxonomy, functional sites, ligand context, pocket structures, or reactions, as in ZymCTRL (Munsamy et al., 2022), ProCALM (Yang et al., 2024), EnzyGen (Song et al., 2024), LigandMPNN (Dauparas et al., 2025), RoseTTAFold All-Atom (Krishna et al., 2024), and GENzyme (Hua et al., 2024). These methods are effective when such annotations or structures are available, but our setting is stricter: only the substrate is given at test time, without oracle EC labels, active-site structures, complexes, or products.

KG and retrieval-augmented zero-shot design. Retrieval injects external evidence into molecular and protein models, from RetMol (Wang et al., 2022) to MSA Transformer (Rao et al., 2021), AlphaFold2 (Jumper et al., 2021), and RSA (Ma et al., 2024). SENZ (Du et al., 2025) is closest to our task: it formalizes zero-shot substrate-specified enzyme generation and retrieves enzymes through substrate similarity before conditioning generation on a multiple sequence alignment (MSA). An MSA places related protein sequences in shared residue columns, providing a common coordinate system for comparing or editing corresponding positions. Existing retrieval contexts, however, are usually treated as a single pool, or depend on sequence similarity to a known protein. We instead retrieve over a biochemical KG that links molecules, catalytic records, protein families, functional annotations, and pathways. ANCHORZYME then separates the retrieved evidence into a scaffold anchor and catalytic donor memories, making scaffold preservation and catalytic cue transfer explicit.

Surrogate-guided enzyme optimization and evaluation. For substrates without experimental labels, generated enzymes are commonly optimized or evaluated with frozen computational surrogates. UniKP (Yu et al., 2023) predicts kinetic parameters, while ProSmith (Kroll et al., 2024), NeuralPLexer (Qiao et al., 2024), AlphaFold3 (Abramson et al., 2024), ESM-2/ESMFold (Lin et al., 2023), and recent evaluation studies (Johnson et al., 2025) provide interaction, structure, or multi-metric assessment signals. Rather than using these models only as post hoc filters or generic guidance, ANCHORZYME uses frozen surrogates to define anchor-relative local optimization: catalytic gain is measured relative to the scaffold anchor, while structural plausibility constrains editing.

3 PROPOSED METHOD: ANCHORZYME

3.1 PROBLEM SETUP AND DESIGN RATIONALE

Given a substrate x represented as a SMILES string, the task is to generate an amino-acid sequence y that is compatible with x while remaining protein-like. SMILES (Simplified Molecular Input Line Entry System) is a text serialization of a molecular graph: its symbols encode atoms, bonds, branches, rings, and, when specified, stereochemistry. It provides the substrate identity and connectivity used

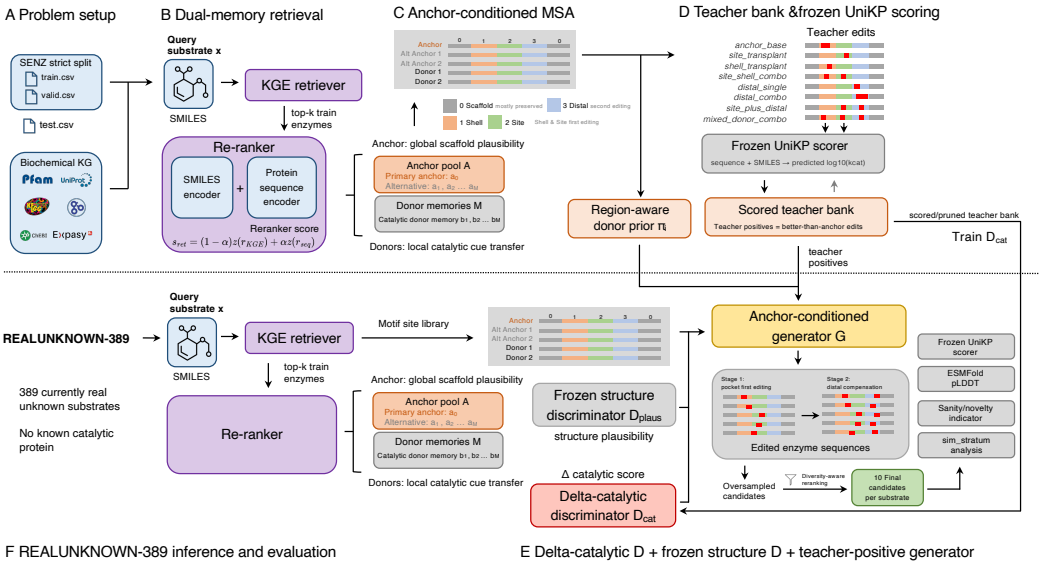

Figure 2: Overview of ANCHORZYME. The framework links KG-guided dual-memory retrieval, anchor-conditioned edit regions, anchor-relative teacher supervision, and two-stage local editing for strict zero-shot enzyme design.

for retrieval and conditioning, but no protein or protein–ligand complex. Our goal is therefore to propose a ranked candidate set from the information available in the substrate, not to claim that a SMILES string uniquely determines one enzyme. In the strict zero-shot setting, evaluation substrates have no accessible enzyme labels: the correct EC label, protein–ligand complex, reaction products, active-site structure, and verified enzyme sequence are all unavailable at test time. An Enzyme Commission (EC) number is a hierarchical label for the reaction catalyzed by a known enzyme. Excluding it prevents the task from reducing to generation conditioned on the target reaction class. Let $\mathcal{G}$ denote the train-only biochemical knowledge graph (KG), $\mathcal{E}_{\text{train}}$ the training-enzyme pool, and $e \in \mathcal{E}_{\text{train}}$ a retrieved enzyme record with sequence y_e. All learnable components use only training records for optimization; validation records are used only for checkpoint or hyperparameter selection.

The key difficulty is that the available evidence is indirect. A KG can retrieve enzymes that are relationally plausible for x, but for a truly unknown substrate it cannot reveal a verified target enzyme. We therefore treat retrieval as a *soft biochemical prior*: related enzymes provide scaffold and residue evidence, while the generator must decide how to adapt them to x. This view follows scaffold-centered enzyme engineering and retrieval-augmented protein modeling (Arnold, 2017; Bornscheuer et al., 2012; Rao et al., 2021; Jumper et al., 2021; Du et al., 2025). Formally, for a selected scaffold anchor a, ANCHORZYME searches for candidates that improve a frozen catalytic surrogate while remaining close to a plausible protein sequence prior:

$$y^\star \in \arg\max_y \Big[u(x,y) + \lambda_{\text{plaus}}\, s_{\text{plaus}}(y) + \lambda_{\text{anc}}\, s_{\text{anc}}(y,a) \Big], \tag{1}$$

where $u(x,y)$ is a frozen enzyme–substrate activity surrogate, $s_{\text{plaus}}(y)$ measures protein-like sequence plausibility, $s_{\text{anc}}(y,a)$ measures preservation of anchor a, and $\lambda_{\text{plaus}}, \lambda_{\text{anc}} \geq 0$ are trade-off weights. Eq. (1) is not optimized as a single black-box objective; it motivates the three algorithmic choices below: KG-guided retrieval, anchor-relative learning, and local two-stage editing.

3.2 DUAL-MEMORY RETRIEVAL AND ALIGNED EDIT SPACE

Candidate retrieval. Retrieval can inject external molecular or protein evidence into generative models (Wang et al., 2022; Rao et al., 2021; Jumper et al., 2021; Ma et al., 2024; Du et al., 2025). For substrate x, an inductive KGE retriever scans $\mathcal{E}_{\text{train}}$ and returns $\mathcal{C}_0(x) \subset \mathcal{E}_{\text{train}}$ with score $r_{\text{kge}}(x,e)$; a sequence-aware reranker $r_{\text{seq}}(x,e)$ refines the same candidates using y_e and the substrate graph. For a minibatch $\mathcal{B}$ of training positives (x, e^+) and J sampled negatives $\{e_j^-\}_{j=1}^J$ per positive, the

reranker uses

$$\mathcal{L}_{\text{rerank}} = \frac{1}{|\mathcal{B}|J} \sum_{(x,e^+)\in\mathcal{B}} \sum_{j=1}^{J} \text{softplus}\left(-\left[r_{\text{seq}}(x,e^+) - r_{\text{seq}}(x,e_j^-)\right]\right). \tag{2}$$

We write $z_{\mathcal{U}}^{[f]}(\cdot)$ for within-pool standardization of score map f_x over finite pool $\mathcal{U}$; Appendix A.2 gives the near-zero-variance rule. With $\alpha \in [0,1]$,

$$\widetilde{s}_{\text{ret}}(x,e) = (1-\alpha) z_{\mathcal{C}_0(x)}^{[r_{\text{kge}}]}(r_{\text{kge}}(x,e)) + \alpha z_{\mathcal{C}_0(x)}^{[r_{\text{seq}}]}(r_{\text{seq}}(x,e)). \tag{3}$$

Anchor and donor memories. Following scaffold-centered enzyme engineering (Bornscheuer et al., 2012; Arnold, 2017), anchors provide starting scaffolds and donor memories provide local catalytic evidence. Let $b_{\text{bg}}(e)$ be a bounded train-only KG background score, $n_{\text{motif}}^{\text{raw}}(e)$ the number of motif/site annotations before anchor-frame projection, $\eta, \mu_{\text{motif}} \geq 0$ augmentation weights, and $\mathbf{1}[\cdot]$ the indicator function. We score the two roles by

$$\begin{aligned} s_{\text{anchor}}(x,e) &= \widetilde{s}_{\text{ret}}(x,e) + \eta b_{\text{bg}}(e) + \mu_{\text{motif}} \mathbf{1}[n_{\text{motif}}^{\text{raw}}(e) > 0], \\ s_{\text{mem}}(x,e) &= \widetilde{s}_{\text{ret}}(x,e) + \eta b_{\text{bg}}(e) + \mu_{\text{motif}} \min(n_{\text{motif}}^{\text{raw}}(e), 3)/3. \end{aligned} \tag{4}$$

The top representative is the primary scaffold anchor a_0; additional high-scoring representatives form the alternative-anchor set $\mathcal{A}_{\text{alt}}(x)$; and diverse high-scoring non-anchors form donor memories $\mathcal{M}(x) = \{m_1, \ldots, m_M\}$. The sampling anchor set is $\mathcal{A}(x) = \{a_0\} \cup \mathcal{A}_{\text{alt}}(x)$.

Anchor-conditioned edit space. Aligned related proteins expose conserved and variable positions (Rao et al., 2021; Jumper et al., 2021); we use them as an anchor-conditioned edit frame. Let Σ_{aa} be the amino-acid alphabet, "$-$" an alignment gap, and mask and pad masking and padding tokens. For each query, anchors, donor memories, and the training target when available are aligned to length L. We write $A_y \in \Sigma_{\text{aln}}^L$ for sequence y, with

$$\Sigma_{\text{aln}} = \Sigma_{\text{aa}} \cup \{-\} \cup \{\text{mask}, \text{pad}\}. \tag{5}$$

For amino-acid columns, the aligned column i maps to the residue index in the unaligned sequence by

$$g_y(i) = 1 + \sum_{k<i} \mathbf{1}[A_y(k) \in \Sigma_{\text{aa}}], \qquad A_y(i) \in \Sigma_{\text{aa}}. \tag{6}$$

The final sequence strips non-amino-acid tokens from A_y.

Let $\rho = (\rho_1, \ldots, \rho_L)$ be the region-label vector, where $\rho_i \in \{0,1,2,3\}$ denotes scaffold, shell, site, or distal. We define $\mathcal{I}_{\text{site}} = \{i : \rho_i = 2\}$, $\mathcal{I}_{\text{shell}} = \{i : \rho_i = 1\}$, and $\mathcal{I}_{\text{distal}} = \{i : \rho_i = 3\}$. Site columns come from train-only motif/site annotations, shell columns are local windows around them, distal columns are donor-supported high-variation columns outside site/shell, and all others are scaffold columns.

Donor-induced edit priors. Rather than copy donor residues, we convert donor evidence into a soft edit prior (Rao et al., 2021; Ma et al., 2024). Let $\Sigma'_{\text{aln}} = \Sigma_{\text{aln}} \setminus \{\text{mask}, \text{pad}\}$, donor m have aligned sequence A_m and score ϱ_m, and $\tau_{\text{mem}} > 0$ be the memory temperature. With $\omega_m^{\text{mem}} = \exp(\varrho_m/\tau_{\text{mem}}) / \sum_{m'} \exp(\varrho_{m'}/\tau_{\text{mem}})$ and smoothing $\epsilon_\pi > 0$, the donor profile at column i is

$$q_i(\beta) = \frac{\epsilon_\pi + \sum_{m\in\mathcal{M}(x)} \omega_m^{\text{mem}} \mathbf{1}[A_m(i) = \beta]}{\sum_{\beta'\in\Sigma'_{\text{aln}}} \left(\epsilon_\pi + \sum_{m\in\mathcal{M}(x)} \omega_m^{\text{mem}} \mathbf{1}[A_m(i) = \beta']\right)}, \qquad \beta \in \Sigma'_{\text{aln}}. \tag{7}$$

For editable columns $\rho_i \in \{1,2,3\}$ and current source anchor a_s with aligned sequence A_{a_s}, anchor weight $\kappa_a \geq 0$ gives

$$\pi_i^{(a_s)}(\beta) \propto q_i(\beta) + \kappa_a \mathbf{1}[\beta = A_{a_s}(i)], \tag{8}$$

with scaffold-column normalization in Appendix A.4. Inference uses

$$\mathcal{I}_{\text{edit}}^{(1)} = \mathcal{I}_{\text{site}} \cup \mathcal{I}_{\text{shell}}, \qquad \mathcal{I}_{\text{edit}}^{(2)} = \mathcal{I}_{\text{distal}}. \tag{9}$$

3.3 ANCHOR-RELATIVE TEACHER AND LEARNING OBJECTIVES

Teacher bank. Frozen surrogates can guide enzyme optimization but may drift off the protein-like manifold when maximized directly (Yu et al., 2023; Johnson et al., 2025). We use UniKP only for anchor-relative supervision. For each training query, the teacher bank $\mathcal{T}(x, a_0, \mathcal{M}(x))$ contains the primary anchor a_0, aligned positives, donor wild types, and anchor-centered proposals such as site, shell, distal, and mixed donor edits. Each $y \in \mathcal{T}(x, a_0, \mathcal{M}(x))$ is scored offline by

$$u(x, y) = \mathrm{UniKP}(x, y). \tag{10}$$

where $u(x, y)$ is the surrogate in Eq. (1). With threshold $\delta_{\mathrm{tp}} > 0$, teacher positives are

$$\mathcal{P}(x, a_0, \mathcal{M}(x)) = \{y \in \mathcal{T}(x, a_0, \mathcal{M}(x)) : u(x, y) > u(x, a_0) + \delta_{\mathrm{tp}}\}. \tag{11}$$

Delta-catalytic discriminator. The catalytic discriminator $D_{\mathrm{cat}}(x, A, \rho)$ maps substrate x, aligned sequence A, and region labels ρ to a scalar score. It predicts anchor-relative gain: for $\mathcal{T} = \{y_1, \ldots, y_n\}$ with anchor index j_a,

$$\begin{aligned} \Delta_j^t &= u(x, y_j) - u(x, y_{j_a}), \\ \Delta_j^s &= D_{\mathrm{cat}}(x, A_{y_j}, \rho) - D_{\mathrm{cat}}(x, A_{y_{j_a}}, \rho). \end{aligned} \tag{12}$$

Let $\bar{\Delta}^t = \mathcal{N}_{\mathrm{teach}}(\Delta^t)$ normalize teacher gains for listwise distillation; ranking and sign targets use raw gains. With temperatures $\tau_t, \tau_s > 0$ and nonnegative weights $\lambda_{\mathrm{KL}}, \lambda_{\mathrm{rank}}, \lambda_{\Delta}, \lambda_{\mathrm{sign}}, \lambda_{\mathrm{reg}}, \lambda_{\mathrm{top1}}$,

$$\begin{aligned} \mathcal{L}_{\mathrm{D}} = \lambda_{\mathrm{KL}} \mathrm{KL}\left(\mathrm{softmax}(\bar{\Delta}^t/\tau_t) \,\|\, \mathrm{softmax}(\Delta^s/\tau_s)\right) + \lambda_{\mathrm{rank}}\mathcal{L}_{\mathrm{rank}} + \lambda_{\Delta}\mathcal{L}_{\Delta}^{\mathrm{code}} \\ + \lambda_{\mathrm{sign}}\mathcal{L}_{\mathrm{sign}} + \lambda_{\mathrm{reg}}\mathcal{L}_{\mathrm{reg}} + \lambda_{\mathrm{top1}}\mathcal{L}_{\mathrm{top1}}. \end{aligned} \tag{13}$$

The terms perform listwise distillation, ranking, delta regression, sign prediction, geometry matching, and best-candidate separation; Appendix A.6 gives normalizers and masks.

Generator. The generator is an anchor-conditioned denoising transformer (Abramson et al., 2024). The reconstruction target y_{tar} is sampled from high-gain teacher positives with probability $p_{\mathrm{tp}} \in [0, 1]$; otherwise the aligned training positive is used, with the anchor as fallback. A corruption level $t \sim \mathrm{Uniform}(0, 1)$ gives substitution probability $p(t) = p_{\min} + (p_{\max} - p_{\min})t$, where $0 \le p_{\min} \le p_{\max} \le 1$:

$$\widetilde{A}_y \sim q_t(\widetilde{A}_y \mid A_{y_{\mathrm{tar}}}). \tag{14}$$

Here q_t is the corruption kernel and $\widetilde{A}_y$ is the corrupted aligned input; the model predicts $p_\theta(Y_i = \beta \mid \widetilde{A}_y, A_{a_0}, \mathcal{M}(x), x, t)$ for amino-acid token Y_i at column i. At inference, the same model uses current source anchor a_s. Let $D_{\mathrm{plaus}}(A)$ be a frozen sequence-plausibility discriminator. For differentiable sample $\widehat{A}$,

$$\begin{aligned} \mathcal{L}_{\mathrm{G}} = \mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{plaus}}^G \mathrm{BCEWithLogits}(D_{\mathrm{plaus}}(\widehat{A}), 1) - \lambda_{\mathrm{cat}}^G D_{\mathrm{cat}}(x, \widehat{A}, \rho) \\ + \lambda_{\mathrm{anchor}}^G \mathcal{L}_{\mathrm{anchor}} + \lambda_{\mathrm{mem}}^G \mathcal{L}_{\mathrm{mem}}. \end{aligned} \tag{15}$$

Here $\mathcal{L}_{\mathrm{rec}}$ reconstructs $A_{y_{\mathrm{tar}}}$, $\lambda_{\mathrm{plaus}}^G, \lambda_{\mathrm{cat}}^G, \lambda_{\mathrm{anchor}}^G, \lambda_{\mathrm{mem}}^G \ge 0$ are weights, $\mathcal{L}_{\mathrm{anchor}}$ preserves scaffold columns, and $\mathcal{L}_{\mathrm{mem}}$ matches donor priors. pLDDT and ESMFold are not online training signals.

3.4 TWO-STAGE SAMPLING AND RERANKING

Two-stage editing. At test time, ANCHORZYME edits selected anchors. Let $\mathcal{A}(x)$ be the anchor set, initialize $A \leftarrow A_{a_s}$ for each source anchor $a_s \in \mathcal{A}(x)$, use $\mathcal{I}_{\mathrm{edit}}^{(1)}$ for site/shell and $\mathcal{I}_{\mathrm{edit}}^{(2)}$ for distal columns, and set $t_{\mathrm{edit}} = 1 - t_{\mathrm{step}}/T_r$ at step $t_{\mathrm{step}} \in \{0, \ldots, T_r - 1\}$ of stage r. For amino-acid substitution (i, β) with $\beta \in \Sigma_{\mathrm{aa}}$, let $A_{i\to\beta}$ be the edited sequence. With nonnegative weights $\lambda_{\mathrm{prior}}, \gamma_{\mathrm{cat}}^{(r)}, \gamma_{\mathrm{str}}^{(r)}$ and $\epsilon_{\log} > 0$,

$$\begin{aligned} S_r(i, \beta; A, a_s) = \log p_\theta(Y_i = \beta \mid A, A_{a_s}, \mathcal{M}(x), x, t_{\mathrm{edit}}) + \lambda_{\mathrm{prior}} \log(\pi_i^{(a_s)}(\beta) + \epsilon_{\log}) \\ + \gamma_{\mathrm{cat}}^{(r)} \left[D_{\mathrm{cat}}(x, A_{i\to\beta}, \rho) - D_{\mathrm{cat}}(x, A, \rho)\right] + \gamma_{\mathrm{str}}^{(r)} \left[D_{\mathrm{plaus}}(A_{i\to\beta}) - D_{\mathrm{plaus}}(A)\right]. \end{aligned} \tag{16}$$

Events combine high-priXority positions, generator top tokens, and donor-supported tokens; implausible edits are filtered when alternatives exist. The sampler chooses the best edit or samples from the event softmax, with online UniKP disabled inside the loop.

Reranking. After oversampling and deduplication, candidates are rescored within $\mathcal{Y}(x)$. Let $a(y)$ be the source anchor for y, and $A_y^{a(y)}$ its aligned row in that query frame. With nonnegative weights $\lambda^R_{\text{cat}}, \lambda^R_{\text{plaus}}, \lambda^R_{\text{uni}}$,

$$R(y) = \lambda^R_{\text{cat}} z^{[D_{\text{cat}}]}_{\mathcal{Y}(x)} \left(D_{\text{cat}}(x, A_y^{a(y)}, \rho) \right) + \lambda^R_{\text{plaus}} z^{[D_{\text{plaus}}]}_{\mathcal{Y}(x)} \left(D_{\text{plaus}}(A_y^{a(y)}) \right) + \lambda^R_{\text{uni}} z^{[u]}_{\mathcal{Y}(x)} \left(u(x, y) \right). \tag{17}$$

Each feature is standardized within $\mathcal{Y}(x)$.

Final sequences use greedy diversity. Let $\mathcal{S}$ be selected candidates, $\text{sim}_{\text{aln}}(y, y')$ padded aligned identity, $\lambda_{\text{div}} \geq 0$ the diversity penalty, and $\beta_{\text{anc}} \geq 0$ the unused-anchor bonus:

$$R_{\text{div}}(y; \mathcal{S}) = R(y) - \lambda_{\text{div}} \max_{y' \in \mathcal{S}} \text{sim}_{\text{aln}}(y, y') + \beta_{\text{anc}} \mathbf{1}[a(y) \notin \{a(y') : y' \in \mathcal{S}\}]. \tag{18}$$

When $\mathcal{S} = \emptyset$, the similarity penalty is zero.

Appendix A gives the full standardization rule, donor-prior normalization, discriminator and generator losses, sequence-plausibility proxy details, hyperparameter map, and complete sampling algorithm.

4 Experiments

4.1 Dataset

We construct RealUnknown-389 to match the discovery setting studied in this paper: the input is a substrate molecule, but no verified catalytic protein is known or allowed at test time. Existing enzyme-design benchmarks usually contain known enzyme–substrate associations or provide labels such as EC classes, structures, reactions, or homologous proteins. These resources are useful for supervised learning, but they do not isolate the substrate-only zero-shot problem. RealUnknown-389 instead contains 389 currently enzyme-unknown substrates with no recorded catalytic protein in the construction snapshot.

The dataset is built from a ChEBI-derived pool of canonical SMILES; ChEBI is a curated database of chemical entities. We remove invalid and multi-component molecules, keep molecules with 4–80 heavy atoms and molecular weight between 50 and 1000 Da, exclude molecules linked to a recorded catalytic protein, and remove all molecules appearing in the original zero-shot train, validation, or test molecule sets. From 160,019 enzyme-unknown molecules passing these rules, we select 389 chemically diverse targets by Morgan-fingerprint subset selection. A Morgan fingerprint maps a molecular graph to a binary pattern of local chemical neighborhoods, and Tanimoto similarity measures the overlap between two such fingerprints, from 0 for no shared encoded features to 1 for identical fingerprints. Figure 3 shows that the benchmark is not concentrated near known substrates: 336 targets have nearest-training Tanimoto similarity below 0.25, and the median is 0.160. The RDKit-sanitizable subset also spans a broad size range, with median molecular weight 248.3 Da and median heavy-atom count 13. Detailed construction and leakage checks are given in Appendix C.

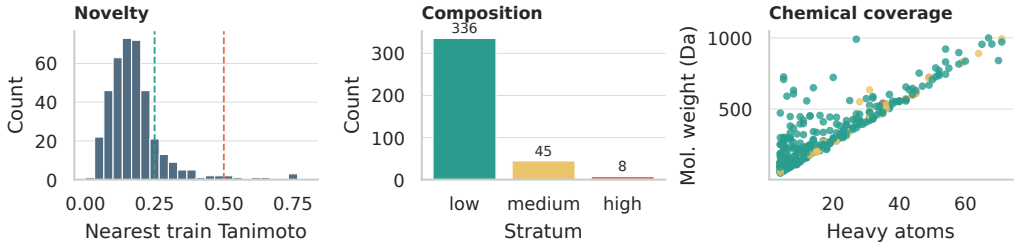

Figure 3: Dataset characterization of RealUnknown-389. Novelty and stratum counts are computed for all 389 substrates; chemical descriptors are computed for the 388 substrates sanitized by RDKit.

All learnable components use only the original zero-shot train and validation splits. No target-specific enzyme label, EC label, reaction product, complex, pocket structure, or verified sequence from RealUnknown-389 is used for training, model selection, or retrieval supervision.

4.2 BASELINES, EVALUATION, AND IMPLEMENTATION DETAILS

We compare against methods that operate from the same substrate-only input. ProGen2 (Nijkamp et al., 2023) and EvoDiff (Alamdari et al., 2023) represent generic protein generation, while adapted NOS (Gruver et al., 2023) represents guided sequence generation. SENZ (Du et al., 2025) is the closest strict zero-shot baseline because it also begins from a substrate query and combines retrieval with generation. Methods that require a correct EC label, a matched protein–ligand complex, reaction products, or an active-site structure are outside this setting.

Because REALUNKNOWN-389 contains no verified target enzyme, we evaluate it as a prospective screening task. UniKP (Yu et al., 2023) is a pretrained model that predicts enzyme kinetic parameters from a substrate–sequence pair; we report its predicted $\log_{10}(k_{\mathrm{cat}})$, where larger values mean higher predicted catalytic turnover. ESMFold (Lin et al., 2023) predicts a three-dimensional protein structure from an amino-acid sequence, and its predicted Local Distance Difference Test (pLDDT) score measures confidence in that predicted structure. We average pLDDT over residues and divide by 100, so a normalized score of 0.770 corresponds to a mean confidence of $77.0/100$. These are frozen computational screening proxies, not measurements of biochemical activity or experimentally determined structure. ESM-2 perplexity, repeated-residue runs, cluster-count difference, and amino-acid repetition provide additional sequence diagnostics. All methods use the same frozen scoring pipeline.

We use a separate set of 128 labeled development targets to analyze candidate recovery. Candidate sets are fixed before the natural reference sequences are opened. Full recovery measures the best complete-sequence match in the returned set; local identity and coverage describe the aligned region. Complete paired statistics and candidate accounting are reported in the appendix.

All trainable modules use the original train split for optimization and the validation split for checkpoint selection. Unless stated otherwise, inference returns up to ten candidates with the same retriever, memories, editor, and frozen reranker. Full settings are given in Appendix A.12.

4.3 MAIN RESULTS ON REALUNKNOWN-389

Table 1: Main comparison on REALUNKNOWN-389 under the substrate-only zero-shot setting. Higher is better for UniKP $\log_{10}(kcat)$ and normalized pLDDT; lower is better for the remaining diagnostics.

	Primary metrics		**Sequence diagnostics**			
Method	**UniKP** $\log_{10}(kcat)\uparrow$	**Norm. pLDDT** $\uparrow$	**ESM-2 ppl** $\downarrow$	**RepeatRun** $\downarrow$	**Cluster-count difference** $\downarrow$	**Repeat-AA distance** $\downarrow$
ProGen2	0.300	0.559	**5.211**	47.266	8.113	80.958
EvoDiff	0.293	0.333	15.484	3.258	9.000	60.742
adapted NOS	0.322	0.740	13.067	3.008	0.715	60.992
SENZ	0.310	0.499	10.485	3.884	7.452	**60.116**
ANCHORZYME	**0.384**	**0.770**	11.915	**2.862**	**0.541**	61.131

Table 1 presents the main result. ANCHORZYME is the only method that ranks first in both primary metrics. Against SENZ, the closest substrate-specified retrieval baseline, it raises the UniKP score from 0.310 to 0.384 and normalized pLDDT from 0.499 to 0.770. It also gives the lowest RepeatRun and cluster-count difference. The higher substrate score therefore does not come from repetitive sequences or cluster collapse. This is the operating point targeted by our design: the anchor preserves a reliable protein context, while local editing adapts that context to the query substrate.

The component results follow the same pattern. Removing catalytic guidance or the search policy reduces UniKP from 0.384 to 0.245 and 0.255, while removing teacher-positive supervision gives 0.371. Removing plausibility guidance instead lowers normalized pLDDT from 0.770 to 0.761. Catalytic guidance and search drive the substrate-facing gain; plausibility guidance protects structural confidence.

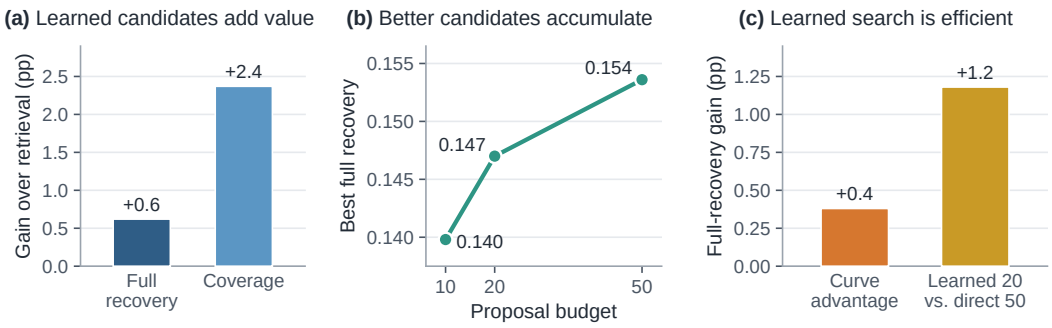

Figure 4: Learned editing expands the retrieval pool, continues to find better candidates as the budget grows, and reaches a stronger operating point with fewer proposals than direct search.

4.4 Learned editing complements retrieval

Retrieval remains the strongest fixed-size core, so we keep it at the center of the final system. Figure 4 shows what the editor adds. Learned candidates improve full recovery by 0.0062 and coverage by 0.0237 over the retrieval parent, with strict gains on 49 of 128 targets. Best available recovery rises from 0.140 at budget 10 to 0.154 at budget 50, and learned search at budget 20 exceeds direct search at budget 50 by 0.0118. Retrieval supplies the starting pool; editing efficiently exposes candidates that retrieval alone misses.

4.5 Anchor and donor memories play different roles

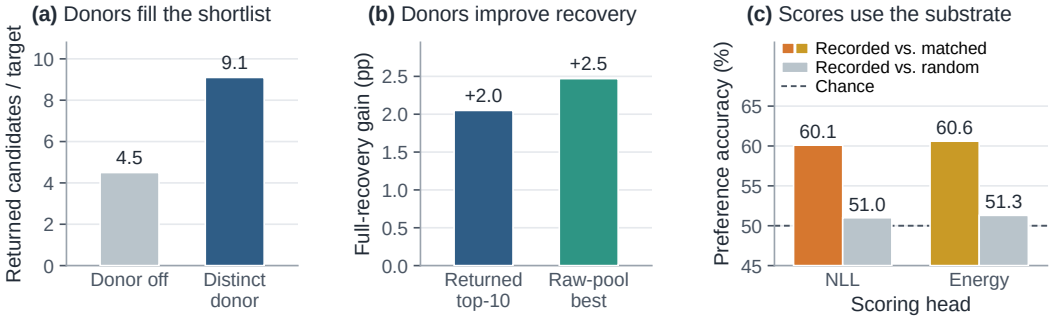

Figure 5: Distinct donor memory fills more of the shortlist and improves recovery. On fixed sequences, both learned heads prefer the recorded substrate to a chemistry-matched alternative.

Figure 5 tests the two signals used by the editor. Under matched anchors, edit support, seeds, budgets, and selection, distinct donor memory increases the returned set from 4.5 to 9.1 candidates per target. It improves returned-set full recovery by 0.0205 and raw-pool recovery by 0.0247, showing that donor memory supplies local alternatives unavailable from the anchor alone.

The substrate condition is also used. With sequence and model state fixed, the NLL and energy heads prefer the recorded substrate to a chemistry-matched alternative with 60.1% and 60.6% accuracy. The direction holds in all 15 checkpoint–mask settings, while random controls remain near 50%. Anchors preserve the scaffold, donors provide edit choices, and the substrate guides those choices.

5 Conclusion

We introduced AnchorZyme for substrate-specified enzyme generation when no verified catalyst is available. It separates retrieved evidence into a scaffold anchor and catalytic donor memory, then combines them through substrate-conditioned local editing. On RealUnknown-389, AnchorZyme achieves the strongest UniKP and normalized pLDDT among substrate-only baselines. Retrieval provides a strong core, while editing, donor memory, and substrate conditioning add complementary candidates. Together, these results establish a retrieval-first route to full-length enzyme proposals for previously unsupported substrates.

Ethics Statement

This work is a purely computational study. All data used are publicly available and non-sensitive: curated chemical and biochemical database resources and published protein sequence records. No human subjects, personally identifiable information, or newly generated wet-lab data are involved.

Reproducibility Statement

Our code is available at the anonymized repository linked in the abstract. The main text and appendices describe the model, benchmark construction, data splits, training settings, hyperparameters, and evaluation protocols needed to reproduce our results.

AI Use Statement

In this work, we used generative AI tools to assist in providing feedback on experimental design and on the interpretation and presentation of experimental results. We also used generative AI tools to draft and edit portions of the manuscript, improve the clarity and organization of the writing, conduct literature search, identify relevant references, and format citations. The authors reviewed and verified all content and take full responsibility for the final manuscript.

References

Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

Sarah Alamdari, Nitya Thakkar, Rianne Van Den Berg, Neil Tenenholtz, Robert Strome, Alan M Moses, Alex X Lu, Nicolò Fusi, Ava P Amini, and Kevin K Yang. Protein generation with evolutionary diffusion: sequence is all you need. *BioRxiv*, pp. 2023–09, 2023.

Frances H Arnold. Directed evolution: bringing new chemistry to life. *Angewandte Chemie (International Ed. in English)*, 57(16):4143, 2017.

Uwe T Bornscheuer, GW Huisman, RJ Kazlauskas, S Lutz, JC Moore, and K Robins. Engineering the third wave of biocatalysis. *Nature*, 485(7397):185–194, 2012.

Justas Dauparas, Ivan Anishchenko, Nathaniel Bennett, Hua Bai, Robert J Ragotte, Lukas F Milles, Basile IM Wicky, Alexis Courbet, Rob J de Haas, Neville Bethel, et al. Robust deep learning–based protein sequence design using proteinmpnn. *Science*, 378(6615):49–56, 2022.

Justas Dauparas, Gyu Rie Lee, Robert Pecoraro, Linna An, Ivan Anishchenko, Cameron Glasscock, and David Baker. Atomic context-conditioned protein sequence design using ligandmpnn. *Nature Methods*, 22(4): 717–723, 2025.

Jiahe Du, Kaixiong Zhou, Xinyu Hong, Zhaozhuo Xu, Jinbo Xu, and Xiao Huang. Retrieval augmented zero-shot enzyme generation for specified substrate. 2025.

Noelia Ferruz, Steffen Schmidt, and Birte Höcker. Protgpt2 is a deep unsupervised language model for protein design. *Nature communications*, 13(1):4348, 2022.

Nate Gruver, Samuel Stanton, Nathan Frey, Tim GJ Rudner, Isidro Hotzel, Julien Lafrance-Vanasse, Arvind Rajpal, Kyunghyun Cho, and Andrew G Wilson. Protein design with guided discrete diffusion. *Advances in neural information processing systems*, 36:12489–12517, 2023.

Chenqing Hua, Jiarui Lu, Yong Liu, Odin Zhang, Jian Tang, Rex Ying, Wengong Jin, Guy Wolf, Doina Precup, and Shuangjia Zheng. Reaction-conditioned de novo enzyme design with genzyme. *arXiv preprint arXiv:2411.16694*, 2024.

Sean R Johnson, Xiaozhi Fu, Sandra Viknander, Clara Goldin, Sarah Monaco, Aleksej Zelezniak, and Kevin K Yang. Computational scoring and experimental evaluation of enzymes generated by neural networks. *Nature biotechnology*, 43(3):396–405, 2025.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

Rohith Krishna, Jue Wang, Woody Ahern, Pascal Sturmfels, Preetham Venkatesh, Indrek Kalvet, Gyu Rie Lee, Felix S Morey-Burrows, Ivan Anishchenko, Ian R Humphreys, et al. Generalized biomolecular modeling and design with rosettafold all-atom. *Science*, 384(6693):eadl2528, 2024.

Alexander Kroll, Sahasra Ranjan, and Martin J Lercher. A multimodal transformer network for protein-small molecule interactions enhances predictions of kinase inhibition and enzyme-substrate relationships. *PLoS computational biology*, 20(5):e1012100, 2024.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, et al. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, 2023.

Chang Ma, Haiteng Zhao, Lin Zheng, Jiayi Xin, Qintong Li, Lijun Wu, Zhi-Hong Deng, Yang Young Lu, Qi Liu, Sheng Wang, et al. Retrieved sequence augmentation for protein representation learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1738–1767, 2024.

Geraldene Munsamy, Sebastian Lindner, Philipp Lorenz, and Noelia Ferruz. Zymctrl: a conditional language model for the controllable generation of artificial enzymes. In *NeurIPS machine learning in structural biology workshop*. NeurIPS, 2022.

Erik Nijkamp, Jeffrey A Ruffolo, Eli N Weinstein, Nikhil Naik, and Ali Madani. Progen2: exploring the boundaries of protein language models. *Cell systems*, 14(11):968–978, 2023.

Zhuoran Qiao, Weili Nie, Arash Vahdat, Thomas F Miller III, and Animashree Anandkumar. State-specific protein–ligand complex structure prediction with a multiscale deep generative model. *Nature Machine Intelligence*, 6(2):195–208, 2024.

Roshan M Rao, Jason Liu, Robert Verkuil, Joshua Meier, John Canny, Pieter Abbeel, Tom Sercu, and Alexander Rives. Msa transformer. In *International conference on machine learning*, pp. 8844–8856. PMLR, 2021.

Donatas Repecka, Vykintas Jauniskis, Laurynas Karpus, Elzbieta Rembeza, Irmantas Rokaitis, Jan Zrimec, Simona Poviloniene, Audrius Laurynenas, Sandra Viknander, Wissam Abuajwa, et al. Expanding functional protein sequence spaces using generative adversarial networks. *Nature Machine Intelligence*, 3(4):324–333, 2021.

Roger A Sheldon and John M Woodley. Role of biocatalysis in sustainable chemistry. *Chemical reviews*, 118(2): 801–838, 2018.

Zhenqiao Song, Yunlong Zhao, Wenxian Shi, Wengong Jin, Yang Yang, and Lei Li. Generative enzyme design guided by functionally important sites and small-molecule substrates. *arXiv preprint arXiv:2405.08205*, 2024.

Zichao Wang, Weili Nie, Zhuoran Qiao, Chaowei Xiao, Richard Baraniuk, and Anima Anandkumar. Retrieval-based controllable molecule generation. *arXiv preprint arXiv:2208.11126*, 2022.

Jason Yang, Aadyot Bhatnagar, Jeffrey A Ruffolo, and Ali Madani. Function-guided conditional generation using protein language models with adapters. *arXiv preprint arXiv:2410.03634*, 2024.

Han Yu, Huaxiang Deng, Jiahui He, Jay D Keasling, and Xiaozhou Luo. Unikp: a unified framework for the prediction of enzyme kinetic parameters. *Nature communications*, 14(1):8211, 2023.

A METHOD DETAILS

This appendix gives implementation-level details for Section 3. The notation follows the main text. All trainable modules follow the original zero-shot split: training data are used for optimization, validation data are used for checkpoint selection, and evaluation-substrate enzyme labels are not used.

A.1 BENCHMARK AND KG RESOURCES

REALUNKNOWN-389 is built from a ChEBI-derived canonical-SMILES molecule pool. Invalid and multi-component molecules are removed, and molecules are filtered by heavy-atom count and molecular weight. A molecule is excluded if it is linked to any recorded catalytic protein in the construction snapshot or if it appears in the original zero-shot train, validation, or test molecule sets. The benchmark is obtained by diversity-oriented subset selection on Morgan fingerprints. Its membership rule is

$$\mathcal{X}_{\mathrm{RU}} = \{x \in \mathcal{X}_{\mathrm{ChEBI}} : \mathrm{valid}(x),\ \mathrm{complexity}(x),\ \mathrm{enzyme_unknown}(x),\ x \notin \mathcal{X}_{\mathrm{SENZ}}\}. \quad (19)$$

Here $\mathcal{X}_{\mathrm{SENZ}} = \mathcal{X}_{\mathrm{train}} \cup \mathcal{X}_{\mathrm{val}} \cup \mathcal{X}_{\mathrm{test}}$ denotes the molecule sets of the original strict zero-shot benchmark. The KG resources include ChEBI SMILES, catalytic activity links, Pfam, EC, GO, ChEBI relations, KEGG orthology, KEGG compounds, modules, and pathways. These resources define unknownness and provide background priors. They do not provide target-specific enzyme labels for REALUNKNOWN-389.

A.2 RERANKER AND DUAL-MEMORY RETRIEVAL

The sequence-aware reranker uses the pairwise loss in Eq. (2). KGE retrieval first forms $\mathcal{C}_0(x)$ for each query. Retrieval and final reranking use the same within-pool standardization rule, applied separately to each query-dependent score vector. For a scalar score map f_x evaluated on a finite item pool $\mathcal{U}$,

$$z_{\mathcal{U}}^{[f]}(v) = \begin{cases} (v - \mu_{\mathcal{U},f})/\sigma_{\mathcal{U},f}, & \sigma_{\mathcal{U},f} > \epsilon_z^{\mathrm{pool}}, \\ v - \mu_{\mathcal{U},f}, & \sigma_{\mathcal{U},f} \le \epsilon_z^{\mathrm{pool}}. \end{cases} \quad (20)$$

The bounded background score $b_{\mathrm{bg}}(e)$ is computed from train-only annotation indicators for the protein associated with candidate e:

$$\begin{aligned} b_{\mathrm{bg}}(e) = {} & 0.30\,\mathbf{1}_{\mathrm{Pfam}}(e) + 0.25\,\mathbf{1}_{\mathrm{EC}}(e) + 0.08\,\min(d_{\mathrm{EC}}(e)/4, 1) \\ & + 0.10\,\mathbf{1}_{\mathrm{GO}}(e) + 0.18\,\mathbf{1}_{\mathrm{GO\text{-}cat}}(e) + 0.06\,\mathbf{1}_{\mathrm{KEGG}}(e), \end{aligned} \quad (21)$$

where $d_{\mathrm{EC}}(e)$ is the deepest observed EC depth. When multiple protein identifiers map to the same sequence, the sequence-level value is the mean of the available protein-level scores. The motif count used during anchor and memory selection is a raw protein-level count,

$$n_{\mathrm{motif}}^{\mathrm{raw}}(e) = |\{\text{motif/site records for the protein of } e\}|, \quad (22)$$

so it is available before the anchor-aligned frame is built. After an anchor is selected, motif records are separately projected into the query-level aligned frame to form site and shell columns.

Anchor and memory scores are given in Eq. (4). Families are ranked by a softmax aggregation of the blended retrieval scores with temperature $\tau_{\mathrm{fam}} = 0.35$: the softmax mass of all candidates in a family is summed, and the top family identifiers are retained. For each retained family, the best representative is the candidate with the largest augmented anchor score $\widetilde{s}_{\mathrm{ret}} + \eta b_{\mathrm{bg}} + \mu_{\mathrm{motif}} \mathbf{1}[n_{\mathrm{motif}}^{\mathrm{raw}} > 0]$. These representatives are merged with the strongest raw candidates, deduplicated by sequence, sorted by the same augmented anchor score, and truncated to the anchor shortlist. Donor memory selection removes the primary-anchor sequence, sorts the remaining candidates by the augmented memory score $\widetilde{s}_{\mathrm{ret}} + \eta b_{\mathrm{bg}} + \mu_{\mathrm{motif}} \min(n_{\mathrm{motif}}^{\mathrm{raw}}, 3)/3$, first keeps at most one candidate per family until the diverse-family cap or memory cap is reached, and then fills remaining slots by the same sorted order. The stored memory score ϱ_m used in donor profiles is the blended retrieval score stored for the selected memory candidate; diversity selection does not rescale it. In the donor-profile implementation, when $\mathcal{M}(x) \neq \emptyset$, memory weights are the softmax of these stored memory scores ϱ_m, with temperature $\tau_{\mathrm{mem}} = 1$ unless otherwise specified:

$$\omega_m^{\mathrm{mem}} = \frac{\exp(\varrho_m/\tau_{\mathrm{mem}})}{\sum_{m' \in \mathcal{M}(x)} \exp(\varrho_{m'}/\tau_{\mathrm{mem}})}, \qquad \sum_{m \in \mathcal{M}(x)} \omega_m^{\mathrm{mem}} = 1. \quad (23)$$

If no donor memory is available, the weighted token profile falls back to the smoothed background profile implemented by `weighted_token_profile`; the prior then reduces mainly to the source-anchor pseudo-count plus smoothing.

A.3 ALIGNED COORDINATE FRAME AND REGION CONSTRUCTION

Primary anchors, alternative anchors, donor memories, teacher candidates, generator rows, and sampled candidates for the same query are stored in one query-level aligned coordinate frame. The aligned alphabet and residue mapping are given in Eqs. (5) and (6). Region labels are column labels. Padding columns, when present in model tensors, are assigned scaffold label 0 and masked out of losses that operate only on valid target positions. Gap columns can appear in the aligned frame and can be seen by the discriminators, but test-time edits are proposed only at amino-acid columns.

The region assignment is

$$\rho_i = \begin{cases} 2, & i \in \mathcal{I}_{\text{site}}, \\ 1, & i \in \mathcal{I}_{\text{shell}}, \\ 3, & i \in \mathcal{I}_{\text{distal}}, \\ 0, & \text{otherwise.} \end{cases} \tag{24}$$

The precedence is site $>$ shell $>$ distal $>$ scaffold. Site columns are projected from a train-only motif/site library. Each motif record contains a protein identifier, one-indexed residue position, optional residue identity, optional confidence, and optional local window radius. The projection maps the ungapped residue index in a retrieved aligned protein to the corresponding alignment column. Only the top retrieved proteins by score are used for query-specific motif projection. Shell columns are local sequence-window expansions around projected sites, with site columns removed from the shell set.

Distal columns are selected from donor-supported variation outside site and shell columns. The donor profile q_i is defined for every alignment column. For tokens outside Σ'_{aln}, we set $q_i(\beta) = 0$. The implementation uses the gap-penalized variation score

$$v_i = \max\left(0, (1 - q_i(A_{a_0}(i)))\,(0.25 + H(q_i)) - 0.35 q_i(\text{gap})\right), \tag{25}$$

where $q_i(\text{gap}) \equiv q_i(-)$ denotes the donor-profile mass on the gap token and $H(q_i)$ is normalized entropy,

$$H(q_i) = -\frac{1}{\log |\Sigma'_{\text{aln}}|} \sum_{\beta \in \Sigma'_{\text{aln}}} q_i(\beta) \log(q_i(\beta) + \epsilon_{\log}).$$

Distal candidates are considered only outside site and shell columns and only at valid primary-anchor amino-acid columns, i.e., $A_{a_0}(i) \in \Sigma_{\text{aa}}$. Thus gap, mask, and padding columns are excluded from distal selection. The final distal columns are selected greedily by decreasing v_i with a fixed maximum column count and a minimum alignment-column separation constraint.

A.4 DONOR PRIOR

The donor token profile is Eq. (7). Mask and padding mass are set to zero; gap mass is allowed during aligned training. The donor profile q_i is shared across source anchors because it depends only on donor rows and their weights. The final prior is anchor-conditioned through the pseudo-count on the current anchor token. Let a_s denote the source anchor used for the current training or sampling call. During generator training $a_s = a_0$; during multi-anchor sampling a_s is the cyclically chosen source anchor.

For scaffold columns, define $h_i^{(a_s)} = \mathbf{1}[A_{a_s}(i) \in \Sigma'_{\text{aln}}]$. The scaffold prior is

$$\pi_i^{(a_s)}(\beta) = \frac{\epsilon_\pi + h_i^{(a_s)} \mathbf{1}[\beta = A_{a_s}(i)]}{|\Sigma'_{\text{aln}}|\epsilon_\pi + h_i^{(a_s)}}, \qquad \rho_i = 0, \quad \beta \in \Sigma'_{\text{aln}}. \tag{26}$$

For editable columns, the prior blends donor evidence with an anchor pseudo-count:

$$\pi_i^{(a_s)}(\beta) = \frac{q_i(\beta) + \kappa_a \mathbf{1}[\beta = A_{a_s}(i)]}{\sum_{\beta' \in \Sigma'_{\text{aln}}} (q_i(\beta') + \kappa_a \mathbf{1}[\beta' = A_{a_s}(i)])}, \qquad \rho_i \in \{1, 2, 3\}. \tag{27}$$

The reported configuration uses $\epsilon_\pi = 10^{-3}$ and $\kappa_a = 0.35$. Log-priors are floored as

$$\log \pi_i^{(a_s)}(\beta) \leftarrow \log\left(\pi_i^{(a_s)}(\beta) + \epsilon_{\log}\right). \tag{28}$$

A.5 TEACHER BANK AND PRUNING

For each training query, the teacher bank contains anchor-centered candidates from anchor base, ground-truth positive, memory wild type, site transplant, shell transplant, site-shell combination, distal single edit, distal combination, site-plus-distal edit, and mixed-donor combination classes. Each candidate stores its aligned sequence and mutation counts in site, shell, distal, and scaffold columns. Teacher scoring uses the frozen surrogate in Eq. (10). The teacher-positive set is Eq. (11).

For generator target selection, teacher positives are first filtered by the threshold in Eq. (11), sorted by their gain over the anchor, and truncated to the configured teacher top-K, giving the index set $\mathcal{P}_K^{\mathrm{top}}$. During training, if $\mathcal{P}_K^{\mathrm{top}}$ is nonempty and the Bernoulli gate with probability p_{tp} succeeds, one positive is sampled as

$$\Pr(y_j^\star = y_j \mid \mathcal{P}_K^{\mathrm{top}}) = \frac{\exp(\Delta_j^t/\tau_{\mathrm{tp}})}{\sum_{\ell \in \mathcal{P}_K^{\mathrm{top}}} \exp(\Delta_\ell^t/\tau_{\mathrm{tp}})}, \qquad j \in \mathcal{P}_K^{\mathrm{top}}. \tag{29}$$

Validation selects the top positive deterministically. If no teacher positive is available, or if the Bernoulli gate fails during training, the aligned ground truth is used when present, otherwise the anchor is used.

The implementation-level teacher proposal caps are: up to three memory-wild candidates; four site transplants; three shell transplants; four site-shell combinations; four distal singles; four distal combinations; four site-plus-distal candidates; and four mixed-donor combinations per group before deduplication. Before training the delta-catalytic discriminator, the bank is pruned to keep the anchor, the optional ground-truth positive, top overall candidates, per-bucket representatives, bottom negative-gain candidates, near-anchor examples, and a capped total bank. The released pruning defaults are $K_{\mathrm{top}} = 8$, $K_{\mathrm{bucket}} = 2$, $K_{\mathrm{bottom}} = 2$, $K_{\mathrm{near}} = 4$, near-anchor minimum absolute gap 0.03 with smallest qualifying gaps kept first, and maximum retained candidates 18. This pruning step reduces cost and stabilizes training. It does not introduce labels from evaluation substrates.

A.6 DELTA-CATALYTIC DISCRIMINATOR LOSSES

For a teacher group $\mathcal{T} = \{y_1, \ldots, y_n\}$ with anchor index j_a, raw teacher and model gains are defined in Eq. (12). The normalized teacher gain is

$$\bar{\Delta}_j^t = \mathcal{N}_{\mathrm{teach}}(\Delta^t)_j. \tag{30}$$

In the main setting $\mathcal{N}_{\mathrm{teach}}$ is z-score normalization with the same fallback convention as Eq. (20) applied to the teacher-bank vector.

The listwise distillation term is

$$\mathcal{L}_{\mathrm{KL}} = \mathrm{KL}\left(\mathrm{softmax}(\bar{\Delta}^t/\tau_t) \,\|\, \mathrm{softmax}(\Delta^s/\tau_s)\right). \tag{31}$$

The discriminator candidate weights are

$$\omega_j^{\mathrm{cand}} = \max\{10^{-6},\, 1 + b_{\mathrm{site}} s_j + b_{\mathrm{shell}} h_j + b_{\mathrm{distal}} d_j + b_{\mathrm{scaffold}} \iota_j^{\mathrm{scaf}} + b_{\mathrm{anchor}} a_j + b_{\mathrm{gt}} g_j + b_{\mathrm{mem}} \iota_j^{\mathrm{mem}}\}, \tag{32}$$

where $s_j, h_j, d_j, \iota_j^{\mathrm{scaf}}$ indicate whether candidate j has at least one site, shell, distal, or scaffold mutation, and $a_j, g_j, \iota_j^{\mathrm{mem}}$ indicate anchor, ground-truth, and memory/donor candidate classes. The released defaults are listed in Appendix A.12.

The anchor-relative ranking term is

$$\begin{aligned}\mathcal{L}_{\mathrm{rank}} = {} & \frac{1}{Z_{\mathrm{rank}}} \sum_{j:\Delta_j^t > \delta_{\mathrm{rank}},\ j \neq j_a} \omega_j^{\mathrm{cand}} \left[\gamma_{\mathrm{rank}} - \left(D_{\mathrm{cat}}(x, A_{y_j}, \rho) - D_{\mathrm{cat}}(x, A_{y_{j_a}}, \rho)\right)\right]_+ \\ & + \frac{1}{Z_{\mathrm{rank}}} \sum_{j:\Delta_j^t < -\delta_{\mathrm{rank}},\ j \neq j_a} \omega_j^{\mathrm{cand}} \left[\gamma_{\mathrm{rank}} - \left(D_{\mathrm{cat}}(x, A_{y_{j_a}}, \rho) - D_{\mathrm{cat}}(x, A_{y_j}, \rho)\right)\right]_+ .\end{aligned} \tag{33}$$

Here $Z_{\mathrm{rank}} = \max(\epsilon_Z, \sum_{j \in I_+ \cup I_-} \omega_j^{\mathrm{cand}})$ with $I_+ = \{j : \Delta_j^t > \delta_{\mathrm{rank}}, j \neq j_a\}$ and $I_- = \{j : \Delta_j^t < -\delta_{\mathrm{rank}}, j \neq j_a\}$.

The direct delta-regression term used in Eq. (13) is the code-level all-candidate term, $\mathcal{L}_\Delta \equiv \mathcal{L}_\Delta^{\text{code}}$:

$$\mathcal{L}_\Delta^{\text{code}} = \frac{1}{Z_\Delta^{\text{all}}} \sum_{j=1}^{n} \omega_j^{\text{cand}} \,\text{SmoothL1}(\Delta_j^s, \bar{\Delta}_j^t), \qquad Z_\Delta^{\text{all}} = \max(\epsilon_Z, \sum_{j=1}^{n} \omega_j^{\text{cand}}). \tag{34}$$

Because $\Delta_{j_a}^s = 0$ identically, the anchor's direct-regression term has zero gradient with respect to the discriminator parameters, although it contributes to the logged scalar loss when $\bar{\Delta}_{j_a}^t \neq 0$. Equivalently, the gradient-bearing part of this term is the non-anchor calibration of raw model logit gains to the normalized teacher-gain unit.

The sign-consistency term ignores near-zero teacher deltas and the anchor:

$$\mathcal{L}_{\text{sign}} = \frac{1}{Z_{\text{sign}}} \sum_{j:\,|\Delta_j^t|>\delta_{\text{rank}},\, j\neq j_a} \omega_j^{\text{cand}} \,\text{BCEWithLogits}\left(\Delta_j^s, \mathbf{1}[\Delta_j^t > 0]\right), \tag{35}$$

where $Z_{\text{sign}} = \max(\epsilon_Z, \sum_{j:\,|\Delta_j^t|>\delta_{\text{rank}},\, j\neq j_a} \omega_j^{\text{cand}})$.

The geometry term z-scores both model gains and teacher-normalized gains inside the same teacher group. For a vector v over the teacher bank, define

$$z_{\mathcal{T}}(v)_j = \begin{cases} (v_j - \mu_{\mathcal{T},v}) / \max(\sigma_{\mathcal{T},v}, \epsilon_{\text{std}}^{\text{teach}}), & \sigma_{\mathcal{T},v} > \epsilon_z^{\text{teach}}, \\ v_j - \mu_{\mathcal{T},v}, & \sigma_{\mathcal{T},v} \leq \epsilon_z^{\text{teach}}. \end{cases} \tag{36}$$

Then

$$\widehat{\Delta}^s = z_{\mathcal{T}}(\Delta^s), \qquad \widehat{\Delta}^t = z_{\mathcal{T}}(\bar{\Delta}^t), \tag{37}$$

and

$$\mathcal{L}_{\text{reg}} = \frac{1}{Z_{\text{reg}}} \sum_{j=1}^{n} \omega_j^{\text{cand}} \left(\widehat{\Delta}_j^s - \widehat{\Delta}_j^t\right)^2, \qquad Z_{\text{reg}} = \max(\epsilon_Z, \sum_{j=1}^{n} \omega_j^{\text{cand}}). \tag{38}$$

When $\mathcal{N}_{\text{teach}}$ is z-score normalization, $z_{\mathcal{T}}(\bar{\Delta}^t)$ is numerically close to $\bar{\Delta}^t$. The notation is kept because the implementation applies the same geometry-normalization routine to teacher and model vectors.

For $j^\star \in \arg\max_j \Delta_j^t$ with deterministic first-index tie-breaking, the top-1 term is

$$\mathcal{L}_{\text{top1}} = \frac{1}{Z_{\text{top1}}} \sum_{j\neq j^\star} \omega_j^{\text{cand}} \left[\gamma_{\text{top1}} - \left(D_{\text{cat}}(x, A_{y_{j^\star}}, \rho) - D_{\text{cat}}(x, A_{y_j}, \rho)\right)\right]_+, \tag{39}$$

where $Z_{\text{top1}} = \max(\epsilon_Z, \sum_{j\neq j^\star} \omega_j^{\text{cand}})$. This expression follows the implementation. If the anchor is the highest-scoring teacher candidate, the term keeps the anchor above the remaining candidates. If the group has only one candidate, the term is zero. The full discriminator objective is Eq. (13). Empty masked index sets give zero loss, and denominators are clamped away from zero.

A.7 Generator objectives and corruption process

The corruption process in Eq. (14) follows the implementation in `make_xt_substitution`. For each batch element, $t \sim \text{Uniform}(0, 1)$ and $p(t) = p_{\min} + (p_{\max} - p_{\min})t$. Each valid non-padding, non-gap column is independently selected for corruption with probability $p(t)$. Selected tokens are replaced by samples from the empirical background amino-acid distribution estimated from training aligned targets. Padding, mask, and gap tokens are not sampled as replacement tokens, and gap columns are frozen. The corrupted index set is denoted I_{corr}.

Let y_{tar} be the reconstruction target defined in Section 3.3, and let $I_{\text{rec}} = \{i : A_{y_{\text{tar}}}(i) \neq \text{pad}\}$. Reconstruction uses

$$\mathcal{L}_{\text{rec}} = -\frac{1}{Z_{\text{rec}}} \sum_{i\in I_{\text{rec}}} \omega_i^{\text{rec}} \log p_\theta(Y_i = A_{y_{\text{tar}}}(i) \mid \widetilde{A}_y, A_{a_0}, \mathcal{M}(x), x, t). \tag{40}$$

The token weights are

$$\omega_i^{\text{rec}} = 1 + \omega_{\text{corr}} \mathbf{1}[i \in I_{\text{corr}}] + \omega_{\text{edit}} \mathbf{1}[A_{y_{\text{tar}}}(i) \neq A_{a_0}(i)], \qquad Z_{\text{rec}} = \max(1, \sum_{i\in I_{\text{rec}}} \omega_i^{\text{rec}}). \tag{41}$$

A differentiable sample is produced with Gumbel–Softmax applied to the full token-logit vector at each alignment column:

$$\widehat{A}_{i,\cdot} = \mathrm{GumbelSoftmax}(\ell_{i,\cdot}; \tau_{\mathrm{GS}}), \qquad \widehat{A}_{i,\beta} \geq 0, \qquad \sum_{\beta \in \Sigma_{\mathrm{aln}}} \widehat{A}_{i,\beta} = 1. \tag{42}$$

The sequence-plausibility proxy term is

$$\mathcal{L}_{\mathrm{plaus}} = \mathrm{BCEWithLogits}(D_{\mathrm{plaus}}(\widehat{A}), 1). \tag{43}$$

The released generator code uses the baseline-free catalytic guidance

$$\mathcal{L}_{\mathrm{cat}}^{(G)} = -D_{\mathrm{cat}}(x, \widehat{A}, \rho). \tag{44}$$

Adding the constant stop-gradient baseline $\mathrm{sg}\, D_{\mathrm{cat}}(x, A_{a_0}, \rho)$ gives an equivalent generator gradient for a fixed anchor, but the scalar implemented loss is Eq. (44). The anchor-retention term is applied on scaffold columns:

$$\mathcal{L}_{\mathrm{anchor}} = -\frac{1}{Z_{\mathrm{anchor}}} \sum_{i:\rho_i=0,\, A_{a_0}(i) \in \Sigma_{\mathrm{aa}}} \log p_\theta(Y_i = A_{a_0}(i) \mid \widetilde{A}_y, A_{a_0}, \mathcal{M}(x), x, t). \tag{45}$$

For the anchor loss, Z_{anchor} is the number of scaffold amino-acid columns in the sum, clamped below by 1 as in the implementation. The donor-memory term uses the direction $\mathrm{KL}(\pi_i \| p_\theta)$ up to a constant:

$$\mathcal{L}_{\mathrm{mem}} = -\frac{1}{Z_{\mathrm{mem}}} \sum_{i \in I_{\mathrm{rec}}} \omega_{\rho_i}^{\mathrm{memCE}} \sum_{\beta \in \Sigma'_{\mathrm{aln}}} \pi_i^{(a_0)}(\beta) \log p_\theta(Y_i = \beta \mid \widetilde{A}_y, A_{a_0}, \mathcal{M}(x), x, t). \tag{46}$$

Here $\omega_2^{\mathrm{memCE}}, \omega_1^{\mathrm{memCE}}, \omega_3^{\mathrm{memCE}}$ are the site, shell, and distal memory weights, while scaffold columns receive zero memory-alignment weight. The normalizer is $Z_{\mathrm{mem}} = \max(1, \sum_{i \in I_{\mathrm{rec}}} \omega_{\rho_i}^{\mathrm{memCE}})$, matching the implementation clamp. The full generator objective is Eq. (15). If a masked set is empty, its loss is zero; denominators are clamped away from zero.

A.8 Sequence-plausibility discriminator

The discriminator denoted by D_{plaus} is a sequence-plausibility discriminator used as a differentiable proxy during generator training. It is not trained from pLDDT, 3D structures, or foldability labels. In the anchor pipeline, it is a sequence-only StructureDiscTransformer loaded from the pretrained Route-2 checkpoint and frozen before generator training.

The checkpoint is trained from CSV files with `seq` and `label` columns using binary cross-entropy with logits. Positive rows are observed training enzyme sequences. Negative rows are retrieval-style hard negatives produced by the discriminator dataset builder: for a substrate, a sampled sequence is used as a negative only if it is not a known positive sequence for the same substrate in the same split. These negatives are compatibility negatives and may still be foldable proteins. Thus D_{plaus} is interpreted as a learned enzyme-sequence plausibility prior, not as a calibrated structural-quality predictor.

The training CSV usually contains ungapped natural enzyme sequences, which are tokenized and padded by the shared vocabulary. During anchor-generator training and sampling, the same model is applied to aligned rows that may include gaps. Gap, mask, and padding are part of the vocabulary. For soft samples, both D_{plaus} and D_{cat} can receive expected token embeddings in the shared aligned vocabulary:

$$E_i = \sum_{\beta \in \Sigma_{\mathrm{aln}}} \widehat{A}_{i,\beta}\, \mathrm{Emb}(\beta). \tag{47}$$

No ESMFold or pLDDT computation is used inside this loss. pLDDT is used only for post-hoc evaluation.

A.9 Sampling algorithm

Sampling starts from an aligned source anchor and applies the two edit stages defined in the main text. At a step in stage r, the implementation first selects a subset of high-priority positions using model

and prior change scores. For an active amino-acid column i with current token $A(i)$, the position priority is

$$s_{\mathrm{pos}}(i) = 0.65\,(1 - p_\theta(Y_i = A(i) \mid A, A_{a_s}, \mathcal{M}(x), x, t_{\mathrm{edit}})) + 0.35\left(1 - \pi_i^{(a_s)}(A(i))\right). \quad (48)$$

The top configured positions by s_{pos} are retained. At a retained position, proposed amino-acid tokens are the union of the generator top-token set and the donor-supported set $\mathcal{V}_i^{\mathrm{mem}} = \{A_m(i) : m \in \mathcal{M}(x),\ A_m(i) \in \Sigma_{\mathrm{aa}}\}$, excluding the current token. Thus the realized event set is a subset of the full combinatorial set:

$$\Omega(A) \subseteq \{(i, \beta) : i \in \mathcal{I}_{\mathrm{edit}}^{(r)},\ A(i) \in \Sigma_{\mathrm{aa}},\ \beta \in \Sigma_{\mathrm{aa}},\ \beta \neq A(i)\}. \quad (49)$$

Events are scored by Eq. (16). Feasible edits satisfy

$$D_{\mathrm{plaus}}(A_{i\to\beta}) - D_{\mathrm{plaus}}(A) \geq \delta_{\mathrm{str}}^{\min}. \quad (50)$$

If the active column set is empty, the stage is skipped. If the realized event set $\Omega(A)$ is empty at a step, the implementation skips that step and continues. Previously edited columns are not removed from $\Omega(A)$, so a column may be revisited later. If feasible edits exist, infeasible edits are masked. If no feasible edit exists under the threshold, the unmasked event set is used as a fallback. The selected edit is the maximum-score edit with probability p_{greedy} and is otherwise sampled from the event-level softmax

$$\Pr((i, \beta) \mid A, r) = \frac{\exp(\widetilde{S}_r(i, \beta; A, a_s))}{\sum_{(i', \beta') \in \Omega_{\mathrm{mask}}(A)} \exp(\widetilde{S}_r(i', \beta'; A, a_s))}, \quad (51)$$

where $\widetilde{S}_r$ denotes the structurally masked edit score and $\Omega_{\mathrm{mask}}(A)$ is the feasible event set when feasible events exist, otherwise the full realized event set. The implementation does not use a separate event-softmax temperature; the code argument τ is the token-level decoding temperature τ_{tok} used to form generator probabilities before Eq. (16). In Stage 2, the implementation additionally skips an update if the selected masked score is lower than the mean unmasked event score by more than numerical tolerance. The implementation uses a total raw trial budget rather than a per-anchor budget. Anchors are visited cyclically, so the raw candidate count before deduplication is B_{raw}, and anchor sources are approximately balanced.

A.10 Final reranking and diversity

Within a substrate-specific candidate pool $\mathcal{Y}(x)$, each reranking feature is standardized separately using Eq. (20). The current implementation uses the absolute catalytic discriminator logit inside this within-substrate standardization, as in Eq. (17). Before scoring, raw candidates are deduplicated by the stripped amino-acid sequence; if the same unaligned sequence is produced by multiple source anchors, the first generated trace in cyclic trial order is retained. Thus $a(y)$ denotes the retained source anchor trace and $A_y^{a(y)}$ denotes its stored aligned row. All source anchors for the same query are embedded in one query-level aligned frame, so the same region vector ρ is used in Eq. (17).

The diversity similarity follows the implementation and uses padded aligned sequence identity. Model tensors are normalized to length L, but the stored aligned strings passed to the diversity helper are treated as strings and may be shorter after serialization or trimming; pairwise padding is therefore retained in the mathematical description. Let A_y and $A_{y'}$ be the stored aligned rows before pairwise padding. Define $L' = \max(|A_y|, |A_{y'}|)$ and let $\bar{A}_y, \bar{A}_{y'}$ be the two rows padded with gap symbols to length L'. Then

$$\mathrm{sim}_{\mathrm{aln}}(y, y') = \frac{\sum_{i=1}^{L'} \mathbf{1}[\bar{A}_y(i) = \bar{A}_{y'}(i)]}{L'}. \quad (52)$$

When the selected set is empty, the maximum similarity penalty in Eq. (18) is zero. Padded aligned identity can overestimate similarity when two candidates share long gap regions; it is therefore used only as a duplicate-suppression heuristic, not as a calibrated diversity metric.

A.11 Training schedule and checkpoint selection

Table 2 gives the training details used for all learned components. All modules use AdamW and gradient clipping at norm 1.0. The inductive KGE retriever uses seed 0 because it builds an internal holdout when the external validation split has no enzyme-bank overlap; the remaining neural modules use seed 2026. Evaluation substrates from REALUNKNOWN-389 are never used for optimization, early stopping, or checkpoint selection.

Algorithm 1 Multi-anchor two-stage sampling

Require: substrate x; aligned anchor pool $\mathcal{A}(x)$; donor memory $\mathcal{M}(x)$; edit sets $\mathcal{I}_{\text{edit}}^{(1)}, \mathcal{I}_{\text{edit}}^{(2)}$; source-anchor prior $\pi^{(a_s)}$; generator G_θ; discriminators $D_{\text{cat}}, D_{\text{plaus}}$; total raw trial budget B_{raw}; final count K

Ensure: final set $\mathcal{S}$

$\mathcal{Y}_{\text{raw}}(x) \leftarrow \emptyset$
for $\ell = 1$ to B_{raw} **do**
  choose source anchor $a_s \leftarrow \mathcal{A}(x)[(\ell - 1) \bmod |\mathcal{A}(x)|]$
  $A \leftarrow A_{a_s}$
  build $\pi^{(a_s)}$ using Eqs. (26)–(27)
  for $r \in \{1, 2\}$ **do**
    choose active set $\mathcal{I}_{\text{act}} \leftarrow \mathcal{I}_{\text{edit}}^{(r)}$
    if $\mathcal{I}_{\text{act}} = \emptyset$ **then**
      continue to the next stage
    end if
    for $t_{\text{step}} = 0$ to $T_r - 1$ **do**
      form the restricted event set $\Omega(A)$ using Eq. (49)
      if $\Omega(A) = \emptyset$ **then**
        continue to the next edit step
      end if
      set $t_{\text{edit}} = 1 - t_{\text{step}}/T_r$
      score each edit using Eq. (16)
      mask structurally infeasible edits using Eq. (50) when feasible edits exist
      select one edit by the greedy-or-softmax rule in Eq. (51) and update A if the stage-specific acceptance rule allows it
    end for
  end for
  add $(\text{strip}_{\Sigma_{\text{aa}}}(A), A, a_s)$ to $\mathcal{Y}_{\text{raw}}(x)$
end for
deduplicate $\mathcal{Y}_{\text{raw}}(x)$ by unaligned sequence, keeping the first generated trace for each sequence
score candidates in $\mathcal{Y}_{\text{raw}}(x)$ using Eq. (17)
initialize $\mathcal{S} \leftarrow \emptyset$
greedily select up to K candidates from $\mathcal{Y}_{\text{raw}}(x)$ using Eq. (18)
return $\mathcal{S}$

A.12 Implementation hyperparameter map

Table 3 summarizes code-level defaults in the released pipeline. Reported operating points may override reranking weights; these overrides are fixed before evaluation and are not tuned on RealUnknown-389 labels.

B Details of anchor-centered teacher-bank candidates

Beyond the anchor itself, our teacher bank contains optional aligned ground-truth positives, donor wild-type candidates, and several anchor-centered edited candidates. Each candidate is stored together with its sequence and the numbers of site, shell, distal, and scaffold mutations.

Let site_diff, shell_diff, and distal_diff denote the donor-specific positions at which the donor differs from the anchor within the site, shell, and distal regions, respectively. The edited candidate types are constructed as follows.

site_transplant. For a single donor, we sample up to two positions from site_diff and transplant the donor residues at those positions onto the anchor. If the donor differs from the anchor at only one site position, only that position is used.

shell_transplant. For a single donor, we sample up to two positions from shell_diff and transplant the donor residues at those shell positions onto the anchor.

Table 2: Training schedule and checkpoint selection for the learned components.

Module	Epochs / batch	Learning setup	Seed	Validation checkpoint metric
Inductive KGE retriever	10 / 128	AdamW, lr 10^{-3}, 8 sampled negatives	0	sampled validation MRR with internal holdout fallback when needed
Sequence-aware reranker	8 / 16	AdamW, lr 2×10^{-4}, 6 negatives per positive	2026	validation pair accuracy minus 0.1 times validation pair loss
Sequence-plausibility discriminator	8 / 64	AdamW, lr 10^{-4}, binary cross-entropy	2026	validation AUC, with validation loss fallback if AUC is undefined
Delta-catalytic discriminator	12 / 8	AdamW, lr 10^{-4}, teacher bank groups pruned to at most 18 candidates	2026	validation top-1 accuracy plus 0.85 pair accuracy minus 0.05 loss
Anchor-conditioned generator	30 / 1	AdamW, lr 3×10^{-4}, teacher-positive targets with probability 0.80	2026	lowest validation generator objective with frozen D_{cat} and D_{plaus}

Table 3: Code-level defaults for retrieval, alignment, teacher-bank construction, and delta-catalytic discrimination in the released anchor-edit pipeline.

Component	Symbols / arguments	Code defaults
Dual-memory retrieval	scan top-k, anchor top-k, memory count	256, 4, 10
Retrieval blending	α, η, μ_{motif}	0.65, 0.08, 0.05
Family diversity	anchor family top-k, memory diverse families	6, 6
Alignment / regions	L, motif top proteins, min sites, shell radius	512, 6, 1, 3
Motif weights	site, shell	1.0, 0.45
Distal selection	distal cap, min separation	24, 4
Donor prior	ϵ_π, κ_a, τ_{mem}	10^{-3}, 0.35, 1.0
Teacher proposals	memory/site/shell/site-shell/distal-single/distal-combo/site+distal/mixed caps	3/4/3/4; 4/4/4/4
Teacher pruning	top/per-bucket/bottom/near/max	8/2/2/4/18
Dcat temperatures	τ_t, τ_s	0.7, 1.0
Dcat losses	$\lambda_{\mathrm{KL}}, \lambda_{\mathrm{rank}}, \lambda_{\Delta}, \lambda_{\mathrm{sign}}, \lambda_{\mathrm{reg}}, \lambda_{\mathrm{top1}}$	0.5, 1.0, 0.6, 0.2, 0.1, 0.8
Dcat margins	δ_{rank}, γ_{rank}, γ_{top1}	0.10, 0.15, 0.15
Dcat candidate weights	site/shell/distal/ scaffold/anchor/GT/memory	$0.20/0.05/0.10/-0.05$; $0.05/0.00/0.05$

site_shell_combo. For a single donor, we sample up to two site positions from site_diff and up to two shell positions from shell_diff, take the union of the sampled positions, and transplant the donor residues at the resulting position set onto the anchor.

distal_single. For a single donor, we sample exactly one position from distal_diff and transplant the donor residue at that distal position onto the anchor.

distal_combo. For a single donor, we sample two or three positions from distal_diff. A candidate is kept only if at least two unique distal positions are selected. The donor residues at those positions are then transplanted onto the anchor.

Table 4: Code-level defaults for generator training, sampling, reranking, and numerical floors. $\epsilon_z^{\text{teach}}$ is the teacher-vector fallback threshold, $\epsilon_{\text{std}}^{\text{teach}}$ is the teacher-vector denominator clamp, and ϵ_Z is the discriminator weighted-mean denominator clamp. Operating-point experiments override final reranking weights using fixed configurations.

Component	Symbols / arguments	Code defaults
Generator target	p_{tp}, teacher top-k, τ_{tp}	$0.80, 4, 0.5$
Generator corruption	$p_{\min}, p_{\max}$, changed weight, edit weight	$0.12, 0.45, 2.0, 2.5$
Generator losses	$\lambda^G_{\text{str}}, \lambda^G_{\text{cat}}, \lambda^G_{\text{anchor}}, \lambda^G_{\text{mem}}$	$1.0, 0.8, 0.20, 0.25$
Generator Gumbel–Softmax	τ_{GS}	1.0
Memory CE weights	$\omega_2^{\text{memCE}}/\omega_1^{\text{memCE}}/\omega_3^{\text{memCE}}$	$1.0/0.45/0.65$
Sampling	$B_{\text{raw}}, T_1, T_2, \tau_{\text{tok}}, p_{\text{greedy}}$	10×3, 18, 18, 0.9, 0.20
Edit proposal	top positions stage 1/2, top tokens	$8/6, 5$
Edit score	$\lambda_{\text{prior}}, \gamma_{\text{cat}}^{(1/2)}, \gamma_{\text{str}}^{(1/2)}$	$0.35, 1.2/0.8, 0.6/1.0$
Sequence-plausibility filter	$\delta_{\text{str}}^{\min}$	-0.20
Final rerank	$\lambda^R_{\text{cat}}, \lambda^R_{\text{str}}, \lambda^R_{\text{uni}}$	$1.0, 0.8, 0.0$ by code default
Diversity	$\lambda_{\text{div}}, \beta_{\text{anc}}, K$	$0.35, 0.10, 10$
Teacher-positive threshold	δ_{tp}	0.03
Numerical floors	$\epsilon_z^{\text{pool}}, \epsilon_z^{\text{teach}}, \epsilon_{\text{std}}^{\text{teach}}, \epsilon_Z, \epsilon_{\log}$	$10^{-8}, 10^{-8}, 10^{-6}, 10^{-8}, 10^{-8}$

Table 5: Anchor-centered teacher-bank candidate types used in our implementation.

Candidate type	Source	Edited positions
anchor_base	anchor	none
gt_positive	aligned GT	donor-independent full target
memory_wild	single donor	donor wild-type sequence
site_transplant	single donor	up to 2 site positions
shell_transplant	single donor	up to 2 shell positions
site_shell_combo	single donor	up to 2 site + up to 2 shell positions
distal_single	single donor	1 distal position
distal_combo	single donor	2–3 distal positions
site_plus_distal	single donor	up to 2 site + up to 2 distal positions
mixed_donor_combo	two donors	site-oriented + distal-oriented edits

site_plus_distal. For a single donor, we sample up to two site positions from site_diff and up to two distal positions from distal_diff, take the union of the sampled positions, and transplant the donor residues at the resulting position set onto the anchor. A candidate is kept only if at least two unique positions are selected.

mixed_donor_combo. We first sample two different donors. From the first donor, we sample up to two site positions; from the second donor, we sample up to two distal positions. We then take the union of these sampled positions. The candidate is constructed by first transplanting residues from the first donor onto the anchor, and then transplanting residues from the second donor onto the intermediate sequence over the same union of positions. A candidate is kept only if at least two unique positions are selected.

Additional candidate types. In addition to the edited candidates above, the teacher bank also contains: (i) anchor_base, i.e., the unedited anchor; (ii) gt_positive, i.e., the aligned ground-truth sequence when it is available and include_gt is enabled; and (iii) memory_wild, i.e., donor wild-type sequences included directly without anchor-centered editing.

C CONSTRUCTION OF REALUNKNOWN-389

REALUNKNOWN-389 is the primary benchmark used throughout this paper. In the main text, we define it as a curated benchmark of 389 currently enzyme-unknown substrates, i.e., substrates that have no currently known enzyme in the resource snapshot used for benchmark construction. Here we provide the full construction procedure.

We begin from a ChEBI-derived molecule pool with canonical SMILES. Molecules are first filtered to remove invalid or multi-component entries. We then restrict molecular complexity to molecules with 4–80 heavy atoms and molecular weight between 50 and 1000 Da. From this filtered pool, we exclude any molecule that is linked to a recorded catalytic protein in the construction snapshot, and we further exclude molecules that already appear in the train/validation/test molecule sets of the original strict zero-shot benchmark. Therefore, membership in REALUNKNOWN-389 is determined only by chemistry and current unknownness; no target-specific catalytic protein or target-specific retrieval signal is used during benchmark construction.

To construct the final benchmark, we select 389 targets by diversity-oriented subset selection on Morgan fingerprints. In our benchmark-building pipeline, this procedure is implemented by src.build_context_unknown_benchmarks, which takes the original train split together with excluded validation/test molecules and the processed KG-derived resources as input, and outputs real_unknown_all.csv, real_unknown_389.csv, and real_unknown_389_senz.csv. The benchmark-building script uses a target size of 389 and novelty thresholds of 0.25 and 0.50 for novelty analysis.

The KG preprocessing stage provides the biochemical resources used during benchmark construction, including chebi_smiles.tsv, catalytic_activity_CHEBI.tsv, protein_family_Pfam.tsv, has_EC.tsv, is_GO.tsv, chebi_relations.tsv, genome_annotation_KEGG.tsv, chebi_KEGG_compound.tsv, KEGG_ko_pathway.tsv, KEGG_ko_module.tsv, and KEGG_module_pathway.tsv. Although the same construction pipeline can also produce context-supported variants such as ctx_unknown_389, the present paper uses only REALUNKNOWN-389for all main experiments.

For post-hoc analysis, each target is additionally assigned a novelty stratum according to its similarity to the training-molecule bank. The benchmark-building pipeline uses novelty_low_thr=0.25 and novelty_high_thr=0.50, and the resulting benchmark CSVs include sim_stratum for low / medium / high novelty analysis. These strata are used only for analysis and do not affect benchmark membership.

D ADDITIONAL EXPERIMENTAL EVIDENCE

This appendix collects the experiments that support the three parts of our design: retrieval supplies a strong starting pool, learned editing expands that pool efficiently, and the anchor–donor decomposition provides useful substrate-dependent evidence. We report the exact paired estimates behind the compact figures and separate complete-sequence recovery from local-match diagnostics.

D.1 EVALUATION PROTOCOL

For the labeled development bridge, every candidate is compared with each associated evaluator-only natural sequence using a fixed BLOSUM62 local aligner with gap-open score -10 and gap-extension score -0.5. Let m be the number of identical aligned residues, a the number of aligned columns, L_c and L_h the candidate and reference lengths, and s_c and s_h their aligned residue spans. We report

$$I_{\text{local}} = \frac{m}{\max(a,1)}, \qquad R_{\text{full}} = \frac{m}{\max(L_c, L_h, 1)}, \qquad C_{\min} = \min\left(\frac{s_c}{\max(L_c,1)}, \frac{s_h}{\max(L_h,1)}\right). \tag{53}$$

Full recovery is the target-wise best-of-set value of R_{full}. Candidate membership and order are frozen before evaluator-only associations or references are opened. Unless stated otherwise, intervals use 10,000 paired target bootstrap replicates. This bridge is exact-pair- and ligand-separated; it is not a protein- or family-disjoint final test.

Table 6: Exact retrieval-complementarity and proposal-efficiency results. Effects are target-level paired differences. The union is compared only with its own cache-derived parent.

Quantity or contrast	Estimate [95% CI]	W/T/L	Accounting
Historical retrieval top-10 full	0.153053	–	requested/actual $K = 10$
Union retrieval parent full	0.149459 $[0.141050, 0.158019]$	–	1,273 slots, $K \leq 10$
Union − own parent, full	$+0.006211$ $[0.004048, 0.008716]$	49/79/0	union $K \leq 20$
Union − own parent, coverage	$+0.023723$ $[0.013534, 0.034623]$	–	availability endpoint
Learned − direct, normalized AUC	$+0.003816$ $[0.000239, 0.007507]$	75/0/53	$B = 10, 20, 50$
Learned $B = 20$ − direct $B = 50$	$+0.011790$ $[0.006365, 0.017605]$	78/1/49	cross-budget contrast

D.2 Learned editing expands a retrieval-first candidate pool

The historical same-cache retrieval top-10 policy remains the fixed-K core: it returns exactly ten candidates and reaches mean full recovery 0.153053. We therefore use learned editing as an expansion mechanism rather than as a replacement for retrieval. The union experiment uses its own cache-derived parent and asks whether the added learned candidates contribute new recoverable sequence evidence.

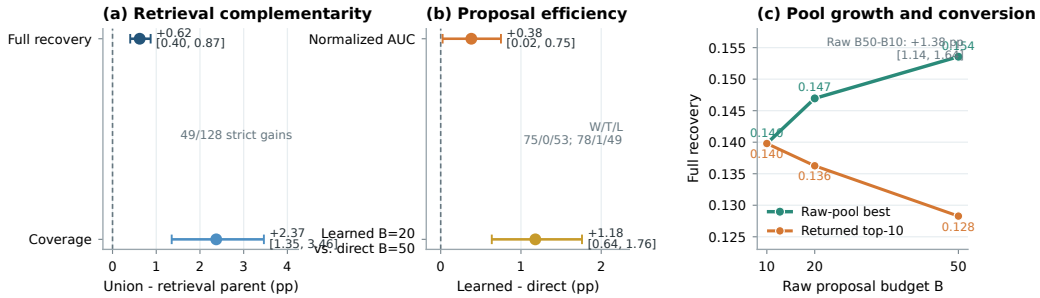

Figure 6: Retrieval-first candidate expansion. (a) Adding learned candidates improves both full recovery and aligned coverage over the union's own retrieval parent. (b) The learned whole-proposal policy has a positive budget-curve summary and reaches a stronger operating point with fewer proposals than direct donor-prior search. (c) Strict prefixes of the same learned streams show that useful candidates continue to accumulate, while the fixed top-10 selector converts a smaller fraction of the larger pool. Points and bars are paired estimates; whiskers are 95% intervals.

In the strict common-stream analysis, increasing the raw proposal budget from $B = 10$ to $B = 50$ raises raw-pool full recovery by $+0.013770$ $[0.011352, 0.016370]$, local identity by $+0.065270$ $[0.050450, 0.082313]$, and coverage by $+0.058163$ $[0.046373, 0.071338]$. The full-recovery gain occurs on 105 targets, with 23 ties and no losses. These results show that editing keeps discovering useful candidates as the proposal pool grows.

D.3 Role separation supplies useful conditional evidence

We isolate the donor role on 114/128 targets selected by a reference-free support rule. Full and donor-off arms use the same anchor, editable positions, checkpoints, sampling seeds, raw budget $B = 20$, requested $K = 10$, and selector. Distinct donor context raises returned full recovery by $+0.020471$ $[0.014875, 0.026478]$ and raw-pool best recovery by $+0.024669$ $[0.018956, 0.030713]$. It also increases the realized output from 4.539 to 9.057 candidates per target without outcome-aware top-up.

To test whether the learned scores use the substrate condition, we next hold the known enzyme sequence, corrupted state, time, anchor–donor context, and mask fixed, and replace only the substrate. Across 310 recorded-substrate clusters, three frozen checkpoints, and five masks, the recorded

substrate is preferred to a chemistry-matched unrecorded substrate by both scoring heads. The NLL head reaches accuracy 0.6013 $[0.5614, 0.6407]$ with margin $+0.03984$ $[0.02446, 0.05612]$; the pair-energy head reaches 0.6059 $[0.5642, 0.6470]$ with margin $+1.4861$ $[0.9995, 1.9948]$. Both directions hold in all 15 checkpoint–mask cells, whereas random-substrate controls remain near chance.

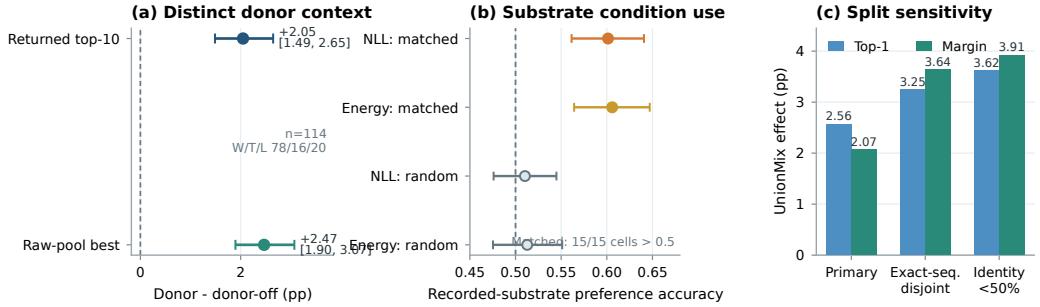

Figure 7: Mechanism and robustness evidence. (a) Distinct donor context improves both returned and raw-pool full recovery under matched proposal conditions. (b) Both frozen scoring heads prefer the recorded substrate to a chemistry-matched alternative on fixed known sequences; random-substrate controls remain near chance. (c) The favorable internal selection-score direction is preserved after exact-sequence exclusion and below 50% identity filtering.

D.4 The favorable direction survives stricter sequence views

The primary internal selection view contains 4,245 rows with no exact pair, canonical ligand, or 512-column projection overlap; its UnionMix top-1 and margin effects are $+0.02562$ and $+0.02068$. Removing all exact fit-sequence reuse leaves 2,309 rows and increases the effects to $+0.032459$ and $+0.036364$. Restricting further to identity below 50% leaves 2,159 rows and yields $+0.036188$ and $+0.039141$. These are internal development-selection endpoints rather than full-recovery results, but the consistent direction shows that the scoring gain does not depend on exact sequence reuse.

Model-free calibration gives a second view of the retrieval signal. An unchanged anchor recovers none of the five masked residues. Donor-prior argmax reaches macro/micro accuracy 0.0847/0.0618, retrieved donor top-1 reaches 0.1460/0.1076, and a top-4 donor vote reaches 0.1521/0.1148. The evaluation-only top-4 oracle is 0.2505/0.1808, indicating additional headroom for using retrieved residue evidence.

D.5 Complete-sequence recovery on the common-74 intersection

On the fixed 74-target intersection with SENZ, AnchorZyme improves full recovery by $+0.105140$ $[0.094582, 0.115655]$ and aligned coverage by $+0.451816$ $[0.411757, 0.491359]$. Both effects favor AnchorZyme on all 74 targets. This is the comparison most closely aligned with our goal: producing complete, scaffold-grounded sequences rather than short high-identity fragments.

Table 7: Paper-faithful comparison on the fixed common-74 intersection. Reporting all three endpoints separates complete-sequence recovery from a short-span local-match objective.

Endpoint	Favored policy	Effect [95% CI]	W/T/L
Full recovery	AnchorZyme	$+0.105140$ $[0.094582, 0.115655]$	74/0/0
Coverage	AnchorZyme	$+0.451816$ $[0.411757, 0.491359]$	74/0/0
Local identity	SENZ	$+0.459760$ $[0.412172, 0.506462]$	71/1/2

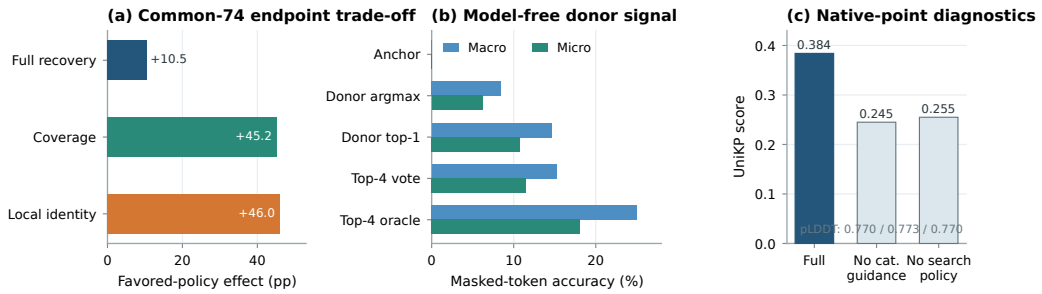

Figure 8: Application diagnostics and endpoint trade-offs. (a) On common-74, ANCHORZYME is stronger in complete-sequence recovery and aligned coverage, while SENZ is stronger in identity within its shorter aligned span. (b) Retrieved donors carry measurable masked-residue signal even without the learned generator. (c) Catalytic guidance and search account for the UniKP gain at the frozen REALUNKNOWN-389 operating point while structural confidence remains stable.

D.6 TWO OPERATING BOUNDARIES

Large proposal pools require a stronger selector. The strict-prefix experiment shows a clear candidate-generation gain, but the frozen top-10 selector does not convert all of it. Raw-pool full recovery rises from 0.139793 at $B = 10$ to 0.153563 at $B = 50$, whereas returned top-10 recovery changes from 0.139793 to 0.128279. Oracle retention consequently falls from 1.000 to 0.262. Thus the generator is most useful as a candidate-pool expander, while larger deployment budgets call for a selector trained specifically for high-volume conversion.

Complete-sequence and short-span objectives favor different systems. On common-74, ANCHORZYME wins every target in full recovery and coverage, but SENZ has higher local identity on 71/74 targets. This is a metric trade-off rather than a contradiction: local identity rewards a strong match inside a shorter aligned span, while our discovery setting prioritizes complete scaffold-grounded proposals. For applications that need only a short locally similar segment, the SENZ-style operating point can be preferable; for full candidate sequences, the recovery and coverage results favor ANCHORZYME.

Evidence scope. The REALUNKNOWN-389 panel measures frozen computational screening diagnostics, not catalytic activity. The union and raw-pool analyses measure candidate availability. The fixed-known-sequence experiment measures directional substrate use in the score, and the split analysis measures an internal selection endpoint. These roles define the claims made in the main paper and identify the next experimental step: independent biochemical testing of the prioritized full-length candidates.

E OPERATING-POINT SENSITIVITY AND ABLATION ANALYSIS

Operating-point sensitivity. Figure 9 shows a controllable activity–structure frontier. mainbest improves UniKP over tradeoff_base while preserving pLDDT, whereas other fixed policies move along the expected activity–reliability trade-off. This behavior is important in a discovery setting: users may prefer a conservative point for downstream validation or a more aggressive point for exploring difficult substrates. The result therefore shows that ANCHORZYME provides a tunable generation policy rather than a single brittle decoding configuration.

Ablation study. Table 8 identifies the main sources of improvement. Removing catalytic guidance or the search policy causes the largest UniKP drops ($0.384 \rightarrow 0.245$ and $0.384 \rightarrow 0.255$), showing that the delta-catalytic signal and two-stage editing are essential for turning retrieved evidence into activity gains. Removing plausibility guidance keeps UniKP high (0.386) but yields the lowest pLDDT (0.761), which indicates that activity-oriented edits alone can drift away from structurally reliable regions. Removing teacher-positive supervision gives a smaller but consistent decrease (0.371), supporting its role as a calibration signal for which retrieved candidates should be imitated. Thus the catalytic branch drives activity improvement, teacher positives calibrate edits, and sequence-plausibility guidance preserves reliability.

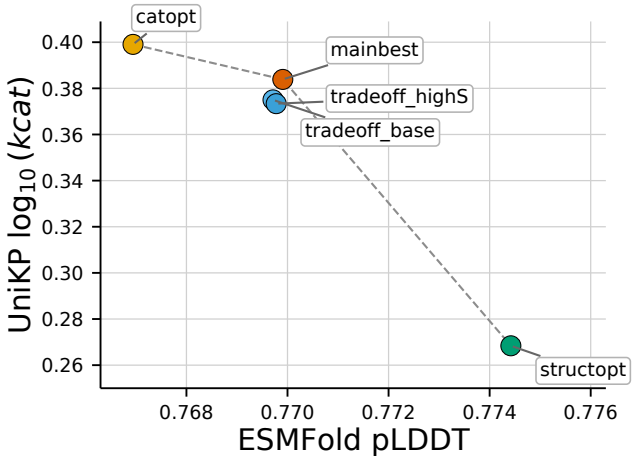

Figure 9: Operating-point sensitivity on REALUNKNOWN-389. Frozen test-time policies trace an activity–structure frontier; mainbest improves catalytic quality over tradeoff_base while preserving structural confidence.

Table 8: Ablation results under the strict zero-shot protocol. UniKP and normalized pLDDT are the primary metrics; the other columns report distributional diagnostics.

Method	**UniKP ↑**	**pLDDT ↑**	**Cluster diff. ↓**	**Repeat-AA WD ↓**
FULL (ANCHORZYME)	0.384	0.770	0.541	61.131
w/o catalytic guidance	0.245	**0.773**	0.376	61.127
w/o plausibility guidance	**0.386**	0.761	0.522	61.130
w/o teacher-positive sup.	0.371	**0.773**	0.475	61.131
w/o search policy	0.255	0.770	**0.073**	**61.040**

F COMPUTATION RESOURCES

All experiments were conducted on a server with 125 GiB memory, 1 Intel Xeon w7-3455 CPU, and 1 NVIDIA RTX PRO 6000 Blackwell Workstation Edition GPU with 96 GB memory. Although the server is equipped with 2 GPUs, only GPU 0 was used in our experiments. All used data in the experiment requires storage of less than 500 GB. The total training time of the models is less than 40 hours.