KiMChEE: Reaction Context from Co-Substrates and Environment Improves Kinetic Prediction for Uncharacterized Enzymes

Anonymous authors
Paper under double-blind review

Abstract

A growing number of machine learning models predict enzyme kinetic parameters such as turnover number (k_{cat}) and Michaelis constant (K_m), along with their ratio (k_{cat}/K_m), from sequence and substrate information for enzyme engineering and metabolic studies. However, their accuracy degrades sharply on enzyme sequences that have never been characterized. We hypothesize that one reason for this failure lies in the input representation. Most existing methods describe a reaction by an enzyme sequence and a main substrate, discarding the remaining reactants and the assay conditions that together determine the kinetic parameter values. Consequently, records sharing an enzyme and a main substrate but differing in reaction context (e.g., different co-substrates and reaction conditions) map to identical inputs with conflicting labels. We therefore build a multimodal kinetic predictor, KiMChEE (Kinetic parameter prediction Model for Chemicals, Enzymes, and Environment). KiMChEE augments the conventional enzyme-and-substrate input with the reaction's other reactants, the co-substrates, and with the assay pH and temperature. It also pools residue-level embeddings with higher weight on substituted positions, keeping variants (e.g., engineered enzymes) separable from their wild types. An ablation isolates the contribution of the co-substrate input, showing that it improves predictions on records whose inputs would otherwise be indistinguishable despite conflicting labels while leaving the remaining records essentially unchanged. We compare methods on an out-of-distribution benchmark constructed from records absent from the public repositories (BRENDA and SABIO-RK) used to train the compared models, so that none of them can have seen these records during training. KiMChEE attains the highest R^2 on all three kinetic parameters under this benchmark, with a margin of 0.128 R^2 on k_{cat} over the second-best model. These results support our hypothesis that the reaction context, rather than an enzyme and a single substrate alone, belongs in the input for kinetic prediction on novel enzymes.

1 Introduction

Enzyme kinetic parameters, including the turnover number (k_{cat}), Michaelis constant (K_m), and catalytic efficiency (k_{cat}/K_m), link molecular catalysis to system-level biological behavior (Johnson & Goody, 2011). They are the properties that enzyme engineering studies seek to improve. Variant libraries are ranked by predicted catalytic performance, and in microbial production these parameters set the flux through a pathway step (Lee et al., 2026; Noor et al., 2016). Despite this demand, experimentally measured kinetic values are available for only a small fraction of known enzyme sequences. This gap has motivated machine learning models that predict kinetic parameters from protein sequences and chemical representations, which have benefited from pretrained protein language models and improved molecular encoders.

However, current predictors generalize poorly to enzymes and reactions unlike those seen during training, where reported accuracy drops sharply once close homologs are absent (Kroll & Lercher, 2024). We hypothesize that one reason for this failure lies in the input representation, which does

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.

not fully describe a reaction. In a multi-reactant assay, one reactant is varied while the others are held fixed, usually at saturating concentrations. Databases record only the varied reactant, which we call the *main substrate*; the others we call *co-substrates*. Existing predictors either discard the co-substrates or merge all reactants into one representation, so neither keeps the varied reactant distinct from the fixed ones. As a result, chemically distinct assays can map onto identical model inputs despite carrying different kinetic targets, leading to an input collision that the model alone cannot resolve. The model can then only learn an average of the conflicting values. For enzymes similar to the training data, this error stays hidden, but it becomes obvious for uncharacterized enzymes. Among the k_{cat} records in the dataset assembled from BRENDA (Hauenstein et al., 2026) and SABIO-RK (Wittig et al., 2018), 34.4% of records contain at least one co-substrate, and 12.8% are associated with multiple distinct co-substrate sets.

Two further issues shape the design. First, averaging protein embeddings over all residues hides one or two substitutions, so a variant looks almost identical to its wild type (NaderiAlizadeh & Singh, 2025). Second, the evaluation protocols in use overestimate how far these models generalize. Random record-level splits place homologous enzymes on both sides, so the reported error reflects memorization of homologs as much as prediction, and test sets built to keep every test enzyme below a fixed sequence identity to any training enzyme reduce that leakage. Such control does not change where the test records come from, and the difficulty is more acute in comparisons across published predictors, since each defines its own held-out partition from the same public database. A valid comparison needs measurements that none of the compared models could have seen, which no within-database split can supply (Zeng et al., 2026).

We introduce KiMChEE, the *Kinetic parameter prediction Model for Chemicals, Enzymes, and Environment*, a multimodal kinetic predictor. KiMChEE represents the main substrate and an unordered set of co-substrates as distinct input components alongside the enzyme sequence and the environment, so that records differing only in reaction context remain distinguishable. It also pools residue-level protein embeddings with higher weight on substituted positions, designed to keep variants separable from their wild types. An ablation isolates the contribution of the co-substrate input, which improves the records whose inputs would otherwise collide while leaving the remaining records essentially unchanged. We evaluate KiMChEE on an out-of-distribution benchmark curated from the primary literature and disjoint from the public databases used for training, so that none of the compared models could have encountered its records during training. On this benchmark, KiMChEE ranks first on all three kinetic parameters, as measured by R^2. The margin over the second-best model on k_{cat} is 0.128. These results support the hypothesis that the reaction context, rather than an enzyme and a single substrate alone, belongs in the input for kinetic prediction on enzymes that have never been characterized.

2 Related works

2.1 Reaction Representation in Kinetic Predictors

Most learning-based kinetic predictors condition their estimates on an enzyme sequence and a single molecular representation corresponding to the main substrate. DLKcat (Li et al., 2022) pairs a graph-based substrate encoder with a sequence-derived protein representation for k_{cat}, and DLTKcat (Qiu et al., 2023) retains this compound–protein formulation while adding bidirectional attention between substrate atoms and enzyme residues, and taking temperature as an additional input. UniKP (Yu et al., 2023) replaces task-specific encoders with a pretrained protein language model and a pretrained SMILES language model, feeding a tree-based regressor, and extends prediction to K_m and k_{cat}/K_m. Later methods preserve the enzyme–substrate formulation and strengthen the encoders. EITLEM-Kinetics (Shen et al., 2024) couples ESM-derived residue representations (Lin et al., 2023) with an attention block and transfers information iteratively among the three kinetic targets, with mutant enzymes as its stated focus; CataPro (Wang et al., 2025) combines pretrained protein representations with molecular fingerprints. In each case the reaction is represented by its main substrate alone, so the co-substrates are absent from the input and chemically distinct reactions can map onto the same enzyme–substrate pair.

Two lines of work broaden this chemical context. CatPred (Boorla & Maranas, 2025) concatenates all available reactant SMILES for k_{cat} prediction, while TurNuP (Kroll et al., 2023) combines a protein-sequence representation with a differential reaction fingerprint computed from the complete

reaction. Both add chemical information that a single substrate cannot carry, and the fingerprint additionally encodes the transformation itself, not just the molecules present. Neither, however, marks which reactant was varied; that reactant is pooled with the co-substrates. Two measurements on the same set of reactants therefore receive the same input, even when one varied the first reactant and the other varied the second.

2.2 Test-Set Design in Kinetic Predictors

Random record-level splits place highly homologous enzymes in both the training and test sets, so the resulting estimates of generalization are optimistic. Several studies therefore report performance as a function of maximum sequence identity or construct similarity-controlled test sets. TurNuP (Kroll et al., 2023) evaluates enzymes with low similarity to training proteins, CatPred (Boorla & Maranas, 2025) introduces sequence-identity-based out-of-distribution subsets, and CataPro (Wang et al., 2025) assembles benchmarks that reduce the over-representation of closely related enzyme–substrate pairs. EITLEM-Kinetics (Shen et al., 2024) targets mutant enzymes directly, using attention over residue representations to increase sensitivity to substituted positions.

These evaluations control which records land on each side of the split, but not where the records came from. Training and test records are drawn from BRENDA (Hauenstein et al., 2026) and SABIO-RK (Wittig et al., 2018) in every case, and each method defines its own subset of those repositories through its filtering and splitting choices. Each method is therefore evaluated on a partition defined by its own authors, which overstates its advantage over methods evaluated on partitions defined elsewhere. None of these benchmarks measures accuracy on kinetic values curated independently from the primary literature.

3 Method

3.1 Problem Setup and Input Collision

Throughout, E, s_{main} and the elements of $\mathcal{S}_{\text{co}}$ denote raw inputs, an amino acid sequence and SMILES strings. Lower case names a single molecule and upper case a set of them.

A kinetic record is written as

$$r = \big(E,\ s_{\text{main}},\ \mathcal{S}_{\text{co}},\ \mathbf{c}\big), \qquad \mathbf{c} = (\text{pH},\ T), \qquad y = \log_{10} k, \tag{1}$$

where $E \in \mathcal{A}^L$ is the enzyme sequence of length L over the amino-acid alphabet $\mathcal{A}$, s_{main} is the SMILES string of the reactant whose concentration was varied during the assay, and $\mathcal{S}_{\text{co}} = \{s_1, \ldots, s_m\}$ with $m = |\mathcal{S}_{\text{co}}| \in \mathbb{Z}_{\geq 0}$ is an unordered set of co-substrate SMILES held fixed during the measurement, empty when the assay involves a single reactant. The vector $\mathbf{c} \in \mathbb{R}^2$ collects the assay pH and temperature. The target k is one of k_{cat}, K_m, or k_{cat}/K_m; each is learned as a separate regression task on $y = \log_{10} k$, with no parameters shared between them.

Predictors that consume the main substrate alone apply the map, restricted here to the inputs every compared model shares, $\phi_{\text{single}}(r) = (E,\ s_{\text{main}})$, which is not injective on kinetic records:

$$\exists\, r_i \neq r_j \quad \text{such that} \quad \phi_{\text{single}}(r_i) = \phi_{\text{single}}(r_j). \tag{2}$$

Two records collide under ϕ_{single} whenever they share E and s_{main} but differ in $\mathcal{S}_{\text{co}}$, as illustrated in Figure 1. A predictor of the form $\hat{y} = g\big(\phi_{\text{single}}(r)\big)$ then returns one value for both, and the residual $|y_i - y_j|$ is irreducible whatever the capacity of g.

In the k_{cat} dataset, 12.8% of records are collided, corresponding to 6.2% of (E, s_{main}) pairs. Co-substrates are present in 34.4% of k_{cat} records and in 85.1% and 83.1% of K_m and k_{cat}/K_m records (Appendix A), so for those two parameters ϕ_{single} discards reaction context in most records. Non-empty co-substrate sets are small, with a median size of 1 and a 95th percentile of 2, although the largest across the three datasets contains 5.

3.2 Reaction-Context Encoding

Encoded vectors are written $\mathbf{h}$ from here on. The model has four input channels (Figure 2). The enzyme sequence is encoded by ProtT5 (Elnaggar et al., 2022), which returns residue-level states

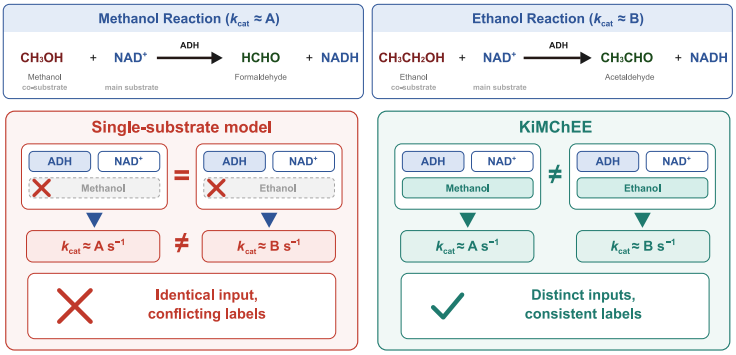

Figure 1: Two k_{cat} records for alcohol dehydrogenase in which NAD^+ is the main substrate and the co-substrate differs. A predictor that consumes the main substrate alone maps both records to the same input while their targets differ; encoding the co-substrate set separately keeps them distinct.

$f_P(E) = (\mathbf{h}_1, \ldots, \mathbf{h}_L)$ with $\mathbf{h}_i \in \mathbb{R}^{1024}$. Molecules are encoded by chemical language models, MolT5 (Edwards et al., 2022) for the main substrate and ChemBERTa (Chithrananda et al., 2020) for co-substrates, giving $\mathbf{h}_{\text{main}} = f_C^{\text{main}}(s_{\text{main}}) \in \mathbb{R}^{512}$ and $f_C^{\text{co}}(s) \in \mathbb{R}^{768}$ for a single co-substrate s. Section 3.5 gives the search from which this pairing was selected.

Co-substrates are encoded independently and the set is reduced by an average:

$$\mathbf{h}_{\text{co}} = \begin{cases} \dfrac{1}{|\mathcal{S}_{\text{co}}|} \displaystyle\sum_{s \in \mathcal{S}_{\text{co}}} f_C^{\text{co}}(s), & \mathcal{S}_{\text{co}} \neq \mathbf{0}, \\ \mathbf{0}, & \mathcal{S}_{\text{co}} = \mathbf{0}, \end{cases} \qquad \mathbf{h}_{\text{co}} \in \mathbb{R}^{768}. \tag{3}$$

The average makes $\mathbf{h}_{\text{co}}$ invariant to the ordering of $\mathcal{S}_{\text{co}}$ and defined for every $m \geq 0$, so a reaction equation, a stoichiometry, and a canonical reactant ordering are all unnecessary at inference time.

Writing $\phi_{\text{KiMChEE}}(r) = (E,\ s_{\text{main}},\ \mathcal{S}_{\text{co}},\ \mathbf{c})$, the collision of Eq. 2 does not arise:

$$\mathcal{S}_{\text{co}}^{(i)} \neq \mathcal{S}_{\text{co}}^{(j)} \implies \phi_{\text{KiMChEE}}(r_i) \neq \phi_{\text{KiMChEE}}(r_j), \tag{4}$$

so the input identifies the assayed reaction rather than the enzyme–substrate pair alone.

3.3 Mutation-Aware Pooling

Residue-level states are pooled into a single enzyme representation by a weighted mean that assigns extra weight to substituted positions:

$$\mathbf{h}_E = \frac{\sum_{i=1}^{L} w_i\, \mathbf{h}_i}{\sum_{i=1}^{L} w_i}, \qquad w_i = 1 + \alpha\, \mathbb{I}[i \in \mathcal{M}], \qquad \mathbf{h}_E \in \mathbb{R}^{1024}, \tag{5}$$

where $\mathcal{M} \subseteq \{1, \ldots, L\}$ collects the substituted positions annotated for the record. Wild-type records have $\mathcal{M} = \emptyset$, so Eq. 5 reduces to a uniform mean and one expression covers both subsets. Without tuning, all experiments use $\alpha = 10$.

Sequences are truncated at 1024 residues before encoding, so the pooling runs over at most 1024 positions, and annotated positions that fall beyond the truncation point, outside the sequence, or that cannot be parsed receive no extra weight.

3.4 Fusion and Read-Out

The four channel representations are concatenated:

$$\mathbf{z} = \left[\, \mathbf{h}_E \parallel \mathbf{h}_{\text{main}} \parallel \mathbf{h}_{\text{co}} \parallel \tilde{\mathbf{c}} \,\right] \in \mathbb{R}^{2306}, \qquad \hat{y} = g(\mathbf{z}), \tag{6}$$

where $\parallel$ denotes concatenation and $\tilde{\mathbf{c}}$ is the environment vector standardized to zero mean and unit variance using statistics computed on the training folds only. The fused dimension is $1024 + 512 + 768 + 2 = 2306$. The read-out g is an extremely randomized trees ensemble (Geurts et al., 2006) of 300 trees fitted on $\mathbf{z}$.

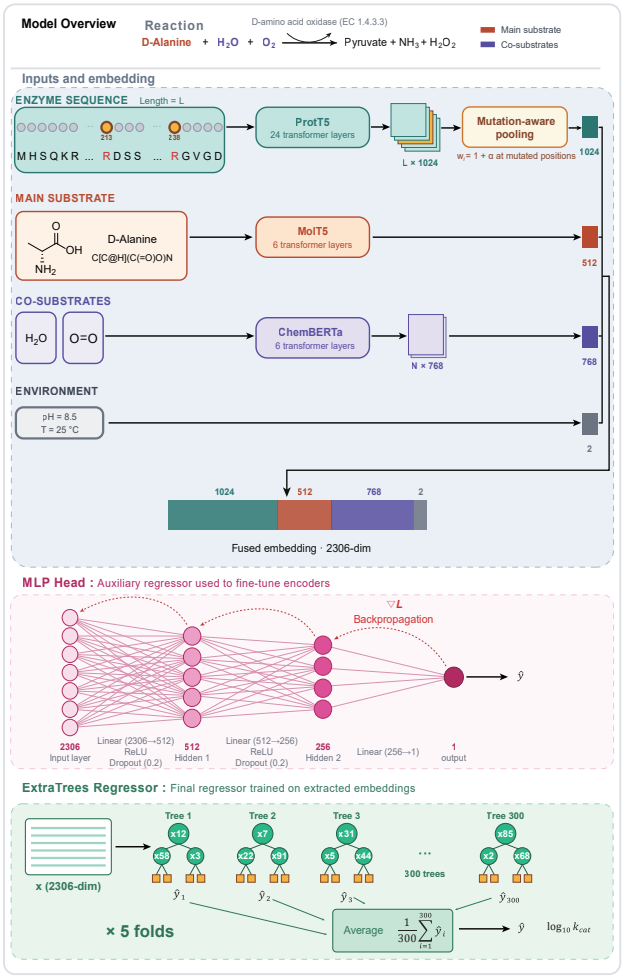

Figure 2: The pipeline shown is trained once per cross-validation fold; at inference the predictions of the five fold models are averaged.

3.5 Training and Configuration Selection

Enzyme sequences are clustered at 50% identity and whole clusters are assigned to folds, so no cluster is split across the $K = 5$ partitions. One model is trained per fold with that partition held out and the final prediction averages the five fold models.

Training proceeds in three stages (Figure 2). The first two use a multilayer perceptron head on $\mathbf{z}$, with hidden widths 512 and 256, ReLU activations and dropout 0.2, which supplies a gradient signal to the encoders in the second stage. First, the parameters of all three pretrained encoders are held fixed and only this head is trained for 3 epochs with AdamW at learning rate 10^{-4}. Second, the parameters of the last four ProtT5 transformer blocks together with its final layer norm, and of the final transformer layer of each chemical encoder, are made trainable, and the model is trained for up to 30 epochs with learning rate 10^{-4} for the head and 10^{-5} for the adapted encoder parameters, batch size 4 with gradient accumulation over 8 steps, gradient norm clipping at 1.0, and mean squared error on y; the checkpoint with the lowest validation loss is retained. Third, all encoder parameters are held fixed again, the head is discarded, and the tree ensemble is fitted on the resulting 2306-dimensional representations.

The architecture was chosen by enumerating all combinations of five binary factors: the read-out (neural head or tree ensemble), encoder adaptation (adapted as described above or kept entirely fixed), the protein encoder (ProtT5 or ESM-2), the main-substrate encoder (MolT5 or ChemBERTa),

and the co-substrate encoder (MolT5 or ChemBERTa). The resulting 32 configurations were each trained on a single 81/9/10 random split on train/validation/test of the k_{cat} dataset. The read-out was the dominant factor; every configuration with a tree ensemble outranked every configuration with a neural head ($0.665 \leq R^2 \leq 0.712$ against $0.463 \leq R^2 \leq 0.619$). Adapting the encoder parameters raised the median R^2 of the tree-ensemble configurations from 0.689 to 0.705, while the protein and chemical encoder choices produced substantially overlapping distributions. The adopted configuration, a tree ensemble on adapted ProtT5, MolT5 and ChemBERTa encoders, ranked first overall ($R^2 = 0.712$, RMSE $= 0.800$; Appendix C). The same configuration was also used for K_m and k_{cat}/K_m.

4 Experiments

4.1 Setup

Training data. Records were assembled from BRENDA and SABIO-RK and, after the filtering below, comprise 25,387 k_{cat}, 37,231 K_m, and 18,637 k_{cat}/K_m entries. A record was discarded when its assay pH or temperature deviated from the enzyme's optimum, predicted from sequence by EpHod (Gado et al., 2025) and Seq2Topt (Qiu et al., 2025) respectively, by more than the q-th quantile of the deviation distribution. At $q = 0.96$ the cut-offs are 2.04 pH units and 32.4 °C, retaining 25,387 of 27,442 k_{cat} records. Held-out R^2 is insensitive to the choice of q over $[0.90, 1.00]$ (Appendix B).

External test set. An independent set was compiled from the primary literature and restricted to measurements absent from both source databases. Two records, one for K_m and one for k_{cat}/K_m, nonetheless duplicated a training record in every field and were removed, leaving 485 k_{cat}, 502 K_m, and 470 k_{cat}/K_m records. Wild-type and mutant records number 240 and 245 for k_{cat}, 259 and 243 for K_m, and 238 and 232 for k_{cat}/K_m (Appendix D).

Baselines. Seven released predictors were evaluated using their published weights without retraining: CataPro, UniKP, CatPred, TurNuP, DLKcat, DLTKcat, and EITLEM-Kinetics. DLKcat, DLTKcat, and TurNuP predict k_{cat} only, CatPred predicts k_{cat} and K_m, and the remaining three predict all three parameters.

Metrics. Performance is reported as R^2, Pearson correlation, Spearman correlation, and RMSE, all computed on $y = \log_{10} k$. Confidence intervals and p values come, unless stated otherwise, from a bootstrap over the evaluated records, resampling with replacement to the size of the set being scored and taking the 2.5th and 97.5th percentiles of the resulting distribution. Comparisons between two models resample the same records for both, so the difference is paired. The ablation in Section 4.2 uses 2,500 resamples and the external benchmark 2,000.

4.2 Channel Ablation

Each ablation removes one input channel from the fused representation and retrains the pipeline on a single random 90/10 partition of the k_{cat} dataset (Table 1). Removing the enzyme channel costs the most, 0.377 R^2, the main substrate costs 0.089, and the co-substrate and environment channels 0.016 and 0.011, so all four channels contribute.

Table 1: Channel ablation on the k_{cat} held-out partition. Each row removes one input channel and retrains. ΔR^2 and p are from a paired bootstrap against the full model.

Variant	R^2 (95% CI)	ΔR^2	p
Full model	0.739 $[0.708, 0.769]$	—	—
− Enzyme	0.362 $[0.303, 0.417]$	-0.377	< 0.001
− Main substrate	0.650 $[0.611, 0.687]$	-0.089	< 0.001
− Co-substrate	0.723 $[0.687, 0.756]$	-0.016	0.018
− Environment	0.728 $[0.697, 0.756]$	-0.011	< 0.001

To locate where the co-substrate channel contributes, the cost of removing it was measured separately on collided and non-collided records and compared with that of the main-substrate and environment channel. A held-out record is collided when its enzyme and main substrate (E, s_{main}) occurs in the dataset with more than one co-substrate set; ϕ_{single} then maps it to the same input as a different reaction (Eq. 2). Of the 2,539 records in the same held-out partition, 345 are collided and 2,194 are not. Removing the co-substrate channel lowers R^2 by 0.122 on the collided records but only by 0.001 on the rest, where the interval contains zero (Table 2). By contrast, removing the main-substrate or environment channel reduces R^2 on both groups. The co-substrate channel thus helps only where the single-substrate input fails to identify the reaction. It does so by supplying reaction information the conventional input omits.

Table 2: Cost of removing each substrate or environment channel on the k_{cat} held-out partition, split by whether the record collides under ϕ_{single}. Entries are ΔR^2 with 95% paired-bootstrap intervals.

Removed channel	Collided ($n = 345$)	Not collided ($n = 2{,}194$)
Co-substrate	-0.122 $[-0.264, -0.020]$	-0.001 $[-0.004, +0.002]$
Main substrate	-0.147 $[-0.214, -0.092]$	-0.081 $[-0.107, -0.058]$
Environment	-0.021 $[-0.036, -0.009]$	-0.010 $[-0.017, -0.004]$

4.3 Out-of-Distribution Benchmark

Published predictors are trained on different subsets of BRENDA and SABIO-RK under different split rules and filtering criteria, so evaluating them on any one model's held-out partition favors that model. A fair comparison therefore requires a test set that none of them was trained on. Because all of them draw on the same two repositories, records absent from both meet this requirement regardless of how each model was split. PubMed articles were first screened automatically for candidate measurements, and articles already cited by either repository were excluded. Because the automated extraction was not reliable enough to use directly, the final set was manually curated from those articles, taking each value, assay condition, and enzyme identity from the article itself.

KiMChEE attains the highest R^2 on each of the three parameters among the seven released predictors (Table 3). The margin over the second-best model is 0.128 on k_{cat}, 0.064 on k_{cat}/K_m, and 0.021 on K_m. KiMChEE also gives the highest Pearson and Spearman correlations on k_{cat} and K_m. On k_{cat}/K_m, it places second on both, behind EITLEM-Kinetics by 0.012 in Pearson and 0.014 in Spearman, although EITLEM-Kinetics attains $R^2 = -0.182$ on that parameter.

Three baselines cannot predict a few records (n in Table 3). CatPred and TurNuP reject sequences with non-standard amino acids and a substrate containing no atom other than hydrogen (e.g., H_2), which account for the same five k_{cat} records in both models, and CatPred loses five K_m records for the same reason. DLKcat cannot parse SMILES containing more than one molecule and loses one record whose substrate is written as a salt with its counterions.

4.4 Wild-Type and Mutant Records

When the external set is scored separately on wild-type and mutant records (Table 4), KiMChEE attains the highest R^2 on the mutant subset for all three parameters. On wild-type records, it is highest for k_{cat} and k_{cat}/K_m, and second to CatPred on K_m.

A paired bootstrap against EITLEM-Kinetics, the one baseline designed specifically for mutant enzymes, gives $\Delta R^2 = +0.378$ (95% CI $[+0.247, +0.526]$, $p < 0.001$) for k_{cat} on the mutant subset. For K_m, the difference is $+0.157$ (95% CI $[-0.004, +0.315]$, $p = 0.056$) and for k_{cat}/K_m $+0.056$ (95% CI $[-0.098, +0.231]$, $p = 0.471$). Only the k_{cat} difference is statistically significant; the K_m and k_{cat}/K_m intervals include zero.

4.5 Direction of Mutation Effects

Matched wild-type and mutant pairs (identical accession, s_{main}, $\mathcal{S}_{\text{co}}$ and $\mathbf{c}$) whose measured values differ by more than 0.1 $\log_{10}$ units were retained, giving 172, 157, and 175 pairs for k_{cat}, K_m, and k_{cat}/K_m, respectively. For each pair, the predicted direction of change from wild type to mutant

Table 3: Performance on the out-of-distribution external test set, $\log_{10}$ units. n is the number of records the model returned a prediction for.

Parameter	Model	n	R^2	PCC	SCC	RMSE
k_{cat}	KiMChEE	485	**0.242**	**0.500**	**0.504**	**1.219**
	UniKP	485	0.114	0.371	0.332	1.318
	CataPro	485	0.105	0.392	0.325	1.325
	TurNuP	480	0.044	0.362	0.298	1.374
	CatPred	480	−0.069	0.369	0.393	1.452
	EITLEM-Kinetics	485	−0.140	0.329	0.260	1.496
	DLKcat	484	−0.198	0.157	0.102	1.530
	DLTKcat	485	−2.234	0.480	0.469	2.519
K_m	KiMChEE	502	**0.443**	**0.695**	**0.711**	**0.850**
	CatPred	497	0.422	0.679	0.670	0.857
	EITLEM-Kinetics	502	0.281	0.582	0.576	0.966
	CataPro	502	−0.063	0.425	0.409	1.174
	UniKP	502	−0.983	0.219	0.199	1.604
k_{cat}/K_m	KiMChEE	470	**0.052**	0.261	0.240	**1.554**
	CataPro	470	−0.012	0.251	0.212	1.606
	UniKP	470	−0.027	0.029	0.018	1.617
	EITLEM-Kinetics	470	−0.182	**0.273**	**0.254**	1.735

Table 4: Performance on the external test set, scored separately on wild-type and mutant records. R^2 with 95% bootstrap confidence intervals.

Parameter	Model	Wild type	Mutant
k_{cat}	KiMChEE	$\mathbf{0.065}$ $[-0.074, 0.202]$	$\mathbf{0.362}$ $[0.296, 0.418]$
	CataPro	-0.125 $[-0.320, 0.030]$	0.263 $[0.165, 0.346]$
	CatPred	-0.500 $[-0.978, -0.133]$	0.240 $[0.140, 0.327]$
	UniKP	0.017 $[-0.066, 0.080]$	0.167 $[0.067, 0.242]$
	TurNuP	-0.123 $[-0.272, -0.011]$	0.148 $[0.058, 0.225]$
	EITLEM-Kinetics	-0.335 $[-0.644, -0.073]$	-0.017 $[-0.170, 0.111]$
	DLKcat	-0.229 $[-0.468, -0.031]$	-0.204 $[-0.382, -0.065]$
	DLTKcat	-1.624 $[-2.355, -1.099]$	-2.803 $[-3.782, -2.010]$
K_m	KiMChEE	0.513 $[0.430, 0.583]$	$\mathbf{0.338}$ $[0.192, 0.453]$
	CatPred	$\mathbf{0.532}$ $[0.431, 0.607]$	0.245 $[0.058, 0.375]$
	EITLEM-Kinetics	0.347 $[0.218, 0.460]$	0.182 $[0.015, 0.314]$
	CataPro	-0.076 $[-0.234, 0.055]$	-0.045 $[-0.315, 0.127]$
	UniKP	-0.782 $[-1.062, -0.556]$	-1.287 $[-1.883, -0.896]$
k_{cat}/K_m	KiMChEE	$\mathbf{0.026}$ $[-0.084, 0.107]$	$\mathbf{0.081}$ $[-0.025, 0.168]$
	CataPro	-0.077 $[-0.191, 0.015]$	0.070 $[-0.074, 0.181]$
	UniKP	-0.069 $[-0.127, -0.024]$	0.025 $[-0.060, 0.089]$
	EITLEM-Kinetics	-0.337 $[-0.578, -0.145]$	0.025 $[-0.196, 0.194]$

(increase or decrease) is compared with the measured direction. The fraction of pairs on which the two agree is then tested against chance (0.5) with a two-sided binomial test (Table 5). On k_{cat}, CataPro and CatPred exceed chance, KiMChEE does not, and TurNuP falls significantly below it. On K_m, no model exceeds chance. On k_{cat}/K_m, KiMChEE and CataPro exceed chance.

Table 5: Sign agreement on matched wild-type and mutant pairs differing by more than 0.1 $\log_{10}$ units. p is from a two-sided binomial test against 0.5. CatPred and TurNuP are scored on fewer pairs because they cannot predict every record, and dashes mark parameters a model does not predict.

	k_{cat}		K_m		k_{cat}/K_m	
Model	n	Agreement (p)	n	Agreement (p)	n	Agreement (p)
KiMChEE	172	0.488 (0.819)	157	0.433 (0.110)	175	**0.634** (0.0005)
CataPro	172	0.593 (0.018)	157	0.522 (0.632)	175	0.594 (0.015)
CatPred	168	0.583 (0.037)	153	0.556 (0.196)	—	—
UniKP	172	0.576 (0.056)	157	0.535 (0.425)	175	0.497 (1.000)
EITLEM-Kinetics	172	0.500 (1.000)	157	0.459 (0.338)	175	0.549 (0.226)
DLKcat	172	0.494 (0.939)	—	—	—	—
DLTKcat	172	0.477 (0.594)	—	—	—	—
TurNuP	168	0.400 (0.011)	—	—	—	—

5 Discussion

Several studies have reported that kinetic predictors generalize poorly, with accuracy falling when test enzymes share low sequence identity with the training data (Kroll & Lercher, 2024; Kroll et al., 2023; Boorla & Maranas, 2025), a finding confirmed by an independent benchmark (Zeng et al., 2026). The out-of-distribution set used here, built from records absent from the source databases rather than from a split of them, reproduces this finding; four of seven released predictors fall below zero R^2 on k_{cat}, and on k_{cat}/K_m none of the three baselines that predict it exceeds zero. KiMChEE ranks first on R^2 for all three parameters, so the reaction-context representation reduces this limitation.

This reduction is consistent with what the input represents. A kinetic measurement is a property of a catalyzed reaction, and the main substrate alone does not specify one. Adding the co-substrates as a separate channel lets the input identify the reaction, and the ablation confirms that the model uses this information. Removing the channel lowers R^2 by a statistically significant 0.016 on the held-out partition, and this drop is concentrated on collided records (0.122, compared with 0.001 on the rest).

The model has two limitations. The first concerns variants. KiMChEE gives the highest R^2 on mutant records for all three parameters, but when asked whether a substitution raises or lowers k_{cat}, it is right no more often than chance, although it does better than chance on k_{cat}/K_m (Table 5). Yet this direction is what enzyme engineering needs most. The weakness is not specific to KiMChEE; on K_m no model tested exceeds chance, and on k_{cat} only two do. It remains an open problem for kinetic predictors in general. The second is absolute accuracy. An R^2 of 0.242 on k_{cat} is the highest on this set but still too low to be trusted zero-shot for an arbitrary enzyme.

Given these limitations, rather than scoring a designed variant once, KiMChEE is better suited to a lab-in-the-loop setting, in which it ranks a library, a few variants are measured, and those measurements retrain it for the next round. By resolving inputs that a single substrate leaves ambiguous, the reaction-context representation gives such a loop a stronger starting point.

6 Conclusion

KiMChEE adds the co-substrates as a separate input channel alongside the enzyme sequence, the main substrate, and the assay conditions, so that records sharing an enzyme and a main substrate no longer collapse to the same input. The ablation shows that the co-substrate channel improves accuracy mainly on these records. Because co-substrates are already recorded in the source databases, the channel requires no new measurements or reaction equations. On an external test set absent from BRENDA and SABIO-RK, KiMChEE outperforms seven released predictors, with the highest R^2 for k_{cat}, K_m, and k_{cat}/K_m, and the highest R^2 on mutant records for all three. Absolute accuracy still leaves room for improvement, and neither KiMChEE nor most baselines predict whether a substitution raises or lowers k_{cat} better than chance. Until these gaps close, the predictions are best used to rank candidates, and retraining on the measurements they guide is a natural next step.

AI USE STATEMENT

We used a large language model to screen candidate articles from the primary literature for reported kinetic measurements and to reformat the extracted values into the tabular schema of the external benchmark. Every retained record was then checked by hand against its source article, and any record whose value, unit, assay condition, or enzyme identity could not be confirmed was discarded. We did not use generative AI to propose or refine the hypotheses of this work, to design the methodology or the experiments, to formulate or prove mathematical claims, to generate synthetic data, or to interpret results.

We also used generative AI to edit the manuscript for readability, to search for relevant literature, to write and revise code for data processing and analysis, and to prepare figures. All AI-assisted output was reviewed by the authors: code was executed and checked against expected results, figures were checked against the underlying data, and every cited reference was read in the original. We take responsibility for the final content of this paper.

ETHICS STATEMENT

This work uses previously published in vitro kinetic measurements and involves no human subjects, no personally identifiable information, and no experiments on living organisms.

REPRODUCIBILITY STATEMENT

The model and its four input channels are specified in Section 3, with the training schedule in Section 3.5 and the configuration search in Appendix C. The assembly of the training data and the assay-condition filter are described in Section 4.1 and Appendices A and B; the construction of the external test set is described in Section 4.3 and Appendix D. The supplementary material contains the external test set, the training data with their fold assignments, the predictions of every model evaluated on it, and the training and evaluation code; hyperparameters and random seeds are set in the training notebook. The pretrained encoders are public checkpoints, named in Section 3.2. The code that assembles the training data, the adapted encoder weights, and the code that runs the baselines are not included; the filtering rules are described in Section 4.1 and Appendix B.

REFERENCES

Veda Sheersh Boorla and Costas D. Maranas. CatPred: A comprehensive framework for deep learning in vitro enzyme kinetic parameters. *Nature Communications*, 16(1):2072, February 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-57215-9.

Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction, October 2020.

Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between Molecules and Natural Language. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 375–413, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.26.

Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. ProtTrans: Toward Understanding the Language of Life Through Self-Supervised Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7112–7127, October 2022. ISSN 0162-8828, 2160-9292, 1939-3539. doi: 10.1109/TPAMI.2021.3095381.

Japheth E. Gado, Matthew Knotts, Ada Y. Shaw, Debora Marks, Nicholas P. Gauthier, Chris Sander, and Gregg T. Beckham. Machine learning prediction of enzyme optimum pH. *Nature Machine Intelligence*, 7(5):716–729, April 2025. ISSN 2522-5839. doi: 10.1038/s42256-025-01026-6.

Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. *Machine Learning*, 63(1):3–42, 2006. doi: 10.1007/s10994-006-6226-1.

Julia Hauenstein, Lisa Jeske, Antje Jäde, Mathias Krull, Katrin Dümmer, Julia Koblitz, Anja Tietz, Dieter Jahn, Lorenz Christian Reimer, and Boyke Bunk. BRENDA in 2026: A Global Core Biodata Resource for functional enzyme and metabolic data within the DSMZ Digital Diversity. *Nucleic Acids Research*, 54(D1):D527–D534, January 2026. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkaf1113.

Kenneth A. Johnson and Roger S. Goody. The Original Michaelis Constant: Translation of the 1913 Michaelis–Menten Paper. *Biochemistry*, 50(39):8264–8269, October 2011. ISSN 0006-2960, 1520-4995. doi: 10.1021/bi201284u.

Alexander Kroll and Martin J. Lercher. DLKcat cannot predict meaningful k_{cat} values for mutants and unfamiliar enzymes. *Biology Methods and Protocols*, 9(1):bpae061, 2024. doi: 10.1093/biomethods/bpae061.

Alexander Kroll, Yvan Rousset, Xiao-Pan Hu, Nina A. Liebrand, and Martin J. Lercher. Turnover number predictions for kinetically uncharacterized enzymes using machine and deep learning. *Nature Communications*, 14(1):4139, July 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-39840-4.

Byung Tae Lee, Byeongsub Lee, Joon Young Kwon, Tilmann Weber, and Hyun Uk Kim. AI and mechanistic modeling for characterizing biosynthetic pathways of natural products. *Natural Product Reports*, 43(5):787–802, 2026. ISSN 0265-0568, 1460-4752. doi: 10.1039/D5NP00059A.

Feiran Li, Le Yuan, Hongzhong Lu, Gang Li, Yu Chen, Martin K. M. Engqvist, Eduard J. Kerkhoven, and Jens Nielsen. Deep learning-based kcat prediction enables improved enzyme-constrained model reconstruction. *Nature Catalysis*, 5(8):662–672, June 2022. ISSN 2520-1158. doi: 10.1038/s41929-022-00798-z.

Zeming Lin, Halil Akin, Roshan Rao, Brian Hie, Zhongkai Zhu, Wenting Lu, Nikita Smetanin, Robert Verkuil, Ori Kabeli, Yaniv Shmueli, Allan Dos Santos Costa, Maryam Fazel-Zarandi, Tom Sercu, Salvatore Candido, and Alexander Rives. Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637):1123–1130, March 2023. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.ade2574.

Navid NaderiAlizadeh and Rohit Singh. Aggregating residue-level protein language model embeddings with optimal transport. *Bioinformatics Advances*, 5(1):vbaf060, 2025. ISSN 2635-0041. doi: 10.1093/bioadv/vbaf060.

Elad Noor, Avi Flamholz, Arren Bar-Even, Dan Davidi, Ron Milo, and Wolfram Liebermeister. The Protein Cost of Metabolic Fluxes: Prediction from Enzymatic Rate Laws and Cost Minimization. *PLOS Computational Biology*, 12(11):e1005167, November 2016. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1005167.

Sizhe Qiu, Simiao Zhao, and Aidong Yang. DLTKcat: Deep learning-based prediction of temperature-dependent enzyme turnover rates. *Briefings in Bioinformatics*, 25(1):bbad506, November 2023. ISSN 1467-5463, 1477-4054. doi: 10.1093/bib/bbad506.

Sizhe Qiu, Bozhen Hu, Jing Zhao, Weiren Xu, and Aidong Yang. Seq2Topt: A sequence-based deep learning predictor of enzyme optimal temperature. *Briefings in Bioinformatics*, 26(2):bbaf114, March 2025. ISSN 1467-5463, 1477-4054. doi: 10.1093/bib/bbaf114.

Xiaowei Shen, Ziheng Cui, Jianyu Long, Shiding Zhang, Biqiang Chen, and Tianwei Tan. EITLEM-Kinetics: A deep-learning framework for kinetic parameter prediction of mutant enzymes. *Chem Catalysis*, 4(9):101094, September 2024. ISSN 2667-1093. doi: 10.1016/j.checat.2024.101094.

Zechen Wang, Dongqi Xie, Dong Wu, Xiaozhou Luo, Sheng Wang, Yangyang Li, Yanmei Yang, Weifeng Li, and Liangzhen Zheng. Robust enzyme discovery and engineering with deep learning using CataPro. *Nature Communications*, 16(1):2736, March 2025. ISSN 2041-1723. doi: 10.1038/s41467-025-58038-4.

Ulrike Wittig, Maja Rey, Andreas Weidemann, Renate Kania, and Wolfgang Müller. SABIO-RK: An updated resource for manually curated biochemical reaction kinetics. *Nucleic Acids Research*, 46(D1):D656–D660, January 2018. ISSN 0305-1048, 1362-4962. doi: 10.1093/nar/gkx1065.

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, December 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-44113-1.

Zishuo Zeng, Jiao Jin, Rufang Xu, and Xiaozhou Luo. Benchmarking and experimental validation of machine learning strategies for enzyme engineering. 2026. doi: 10.64898/2026.03.29.715152. Preprint.

A Composition of the training data

This section expands on the training data described in Section 4.1. Records were assembled separately for each parameter from BRENDA and SABIO-RK, giving 25,387 k_{cat}, 37,231 K_m, and 18,637 k_{cat}/K_m entries. Figure A1 shows the distribution of $y = \log_{10} k$ for each. All three are unimodal and approximately symmetric on the log scale, spanning roughly ten orders of magnitude, which is the range a regression on y has to cover.

Table A1 quantifies the collision described in Section 3.1. A group of records sharing an enzyme and a main substrate is collided when its members differ in co-substrate set, so that ϕ_{single} maps them to one input. The record share exceeds the pair share for every parameter because a collided pair carries several measurements by construction. The collision is most frequent for k_{cat} and least frequent for k_{cat}/K_m, so the fraction of the training data on which the co-substrate channel can act through collision resolution alone is bounded by these shares.

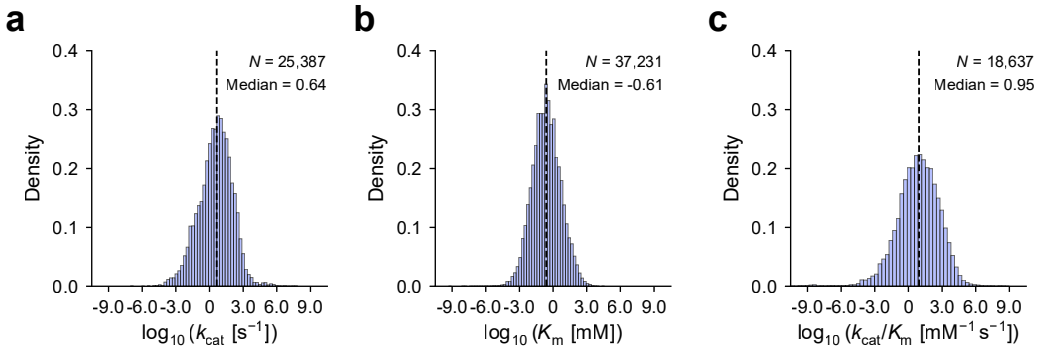

Figure A1: Distribution of $\log_{10} k$ in the training data for (a) k_{cat}, (b) K_m, and (c) k_{cat}/K_m. Dashed lines mark the median.

Table A1: Records and (enzyme, main-substrate) pairs that ϕ_{single} fails to separate, as count (percentage of the corresponding total).

	k_{cat}	K_m	k_{cat}/K_m
Collided records	3,251 (12.8%)	3,974 (10.7%)	1,050 (5.6%)
Collided pairs	1,382 (6.2%)	1,613 (5.0%)	462 (2.7%)
Total records	25,387	37,231	18,637
Total pairs	22,290	32,260	17,111

B Assay condition filtering

Section 4.1 discards a record when its assay pH or temperature departs from the enzyme's predicted optimum by more than the q-th quantile of the deviation distribution, with the optima predicted from sequence by EpHod and Seq2Topt. Figure B1 shows both deviation distributions for the k_{cat} dataset together with the cut-offs at $q = 0.96$. Both are concentrated near zero and fall off with a long tail, so the threshold acts on that tail rather than on the body of the distribution. Of 27,442 records, 957

exceed the pH cut-off alone, 969 the temperature cut-off alone, and 129 both, leaving the 25,387 used for training.

The threshold itself carries little weight. Figure B2 sweeps q over $[0.90, 1.00]$, where $q = 1.00$ applies no filtering, and reports held-out R^2 across the folds at each setting. The distributions overlap throughout: every median lies within 0.01 of the unfiltered mean of 0.734, and no setting separates from no filtering once the fold-to-fold spread is taken into account. We adopt $q = 0.96$, whose mean is the highest of those tested, while noting that the filter is not the source of the accuracy reported in Section 4. Its role is to remove records whose reported conditions are implausible for the enzyme in question, not to tune the result.

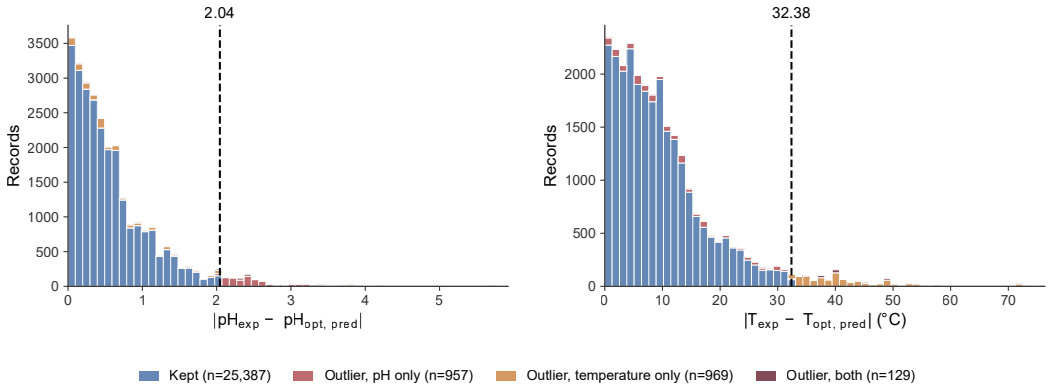

Figure B1: Deviation of the assay conditions from the enzyme's predicted optimum, for the k_{cat} records before filtering. Left, absolute difference between the reported pH and the optimum predicted by EpHod; right, absolute difference between the reported temperature and the optimum predicted by Seq2Topt. Dashed lines mark the cut-offs at $q = 0.96$. Bars are stacked by which criterion a record fails.

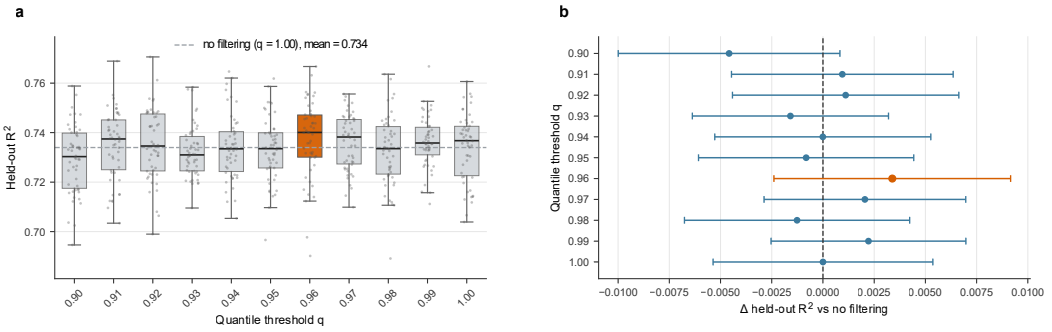

Figure B2: Sensitivity of held-out performance to the filtering threshold. (a) Held-out R^2 at each quantile threshold q; points are individual folds, boxes span the interquartile range, and the dashed line marks the mean without filtering ($q = 1.00$). The adopted setting, $q = 0.96$, is highlighted. (b) Difference in held-out R^2 against no filtering, with 95% confidence intervals.

C Configuration search

The architecture described in Section 3.5 was selected by enumerating all combinations of five binary factors: the protein encoder (ProtT5 or ESM-2), the main-substrate encoder (MolT5 or ChemBERTa), the co-substrate encoder (MolT5 or ChemBERTa), the read-out (a multilayer perceptron head or a tree ensemble on the fused representation), and whether the pretrained encoders are adapted as described in Section 3.5 or left entirely fixed. The last two factors are shown together in Figure C1 as four read-out schemes: *Hybrid* and *ExtraTrees* denote a tree ensemble with

and without encoder adaptation, and *MLP-Unfreeze* and *MLP-Frozen* the corresponding pair for the neural head. Each of the 32 configurations was trained once on a single 81/9/10 split of the k_{cat} data, so the comparison reflects one partition and one seed.

The read-out separates the configurations completely. Every tree-ensemble configuration ranks above every neural-head configuration on both metrics, with no overlap between the two groups (Figure C1a, b). Within the tree-ensemble group, adapting the encoders shifts the median R^2 upward, from 0.689 when the encoders are held fixed to 0.705 when they are adapted; the same adaptation applied to the neural head produces no comparable shift.

The three encoder choices behave differently. Their distributions overlap across most of the range on both R^2 and RMSE (Figure C1c, d), and no pairing is clearly preferable to the alternatives once the read-out is fixed. The adopted configuration, a tree ensemble on adapted ProtT5, MolT5 and ChemBERTa encoders, ranks first overall at $R^2 = 0.712$ and RMSE $= 0.800$, but the next several configurations fall within 0.01 R^2 of it, which is smaller than the spread this search can resolve from a single split. The encoder selection should therefore be read as a choice among near-equivalent options rather than as an identified optimum. The same configuration was used for K_m and k_{cat}/K_m without repeating the search.

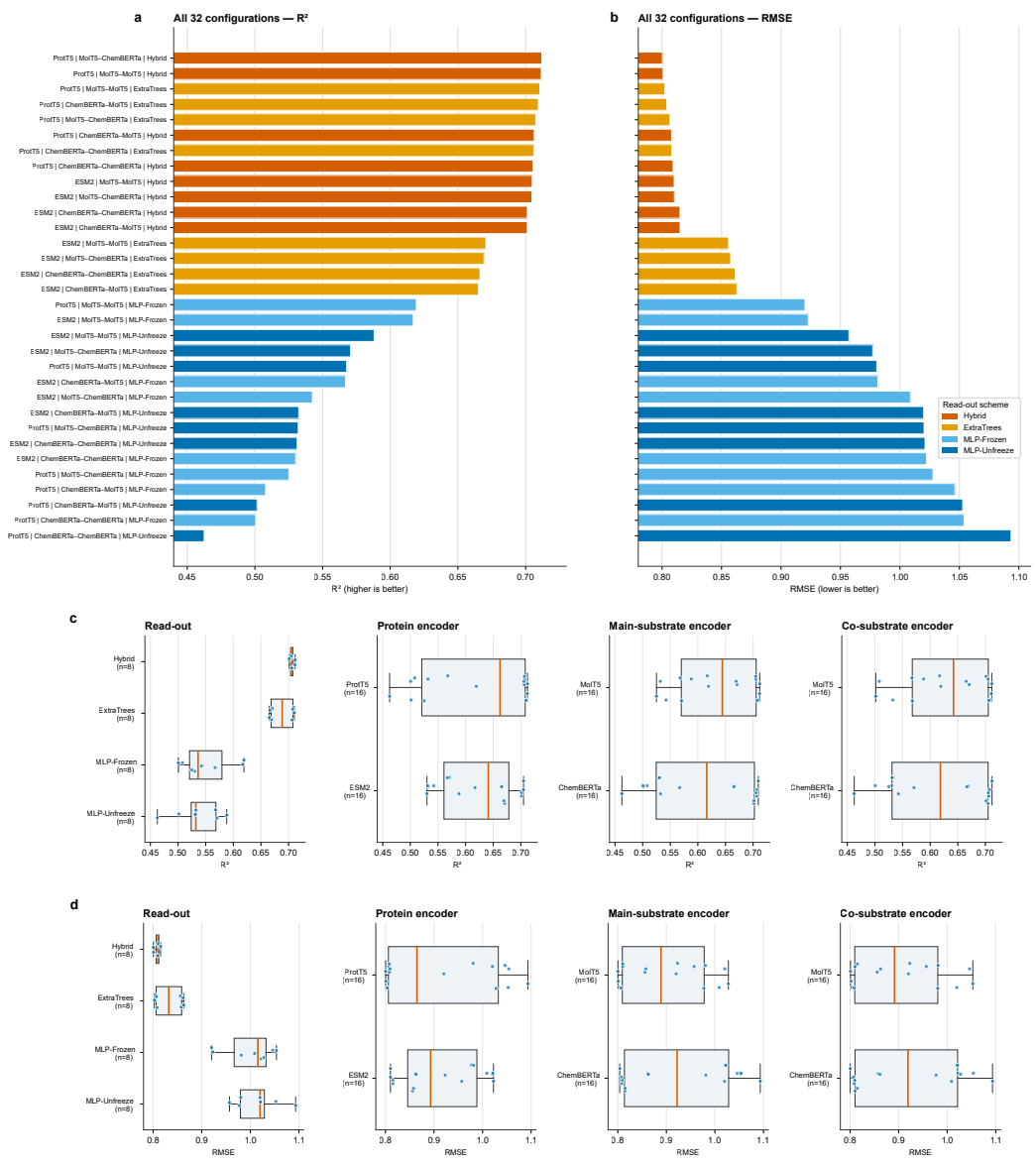

Figure C1: Configuration search over the 32 combinations of protein encoder, main-substrate encoder, co-substrate encoder, read-out, and encoder adaptation, each trained on a single 81/9/10 split of the k_{cat} data. (a, b) All configurations ranked by held-out R^2 and RMSE and coloured by read-out scheme. Labels give the protein encoder, the main-substrate and co-substrate encoders, and the read-out scheme, in that order. (c, d) The same results grouped by each factor separately; each point is one configuration, boxes span the interquartile range and the line marks the median.

D Sequence Homology Between the External Test Set and the Training Data

When the external set was assembled, every article cited by either repository was excluded, and after manual curation the entries that duplicated a training measurement were removed as well. No record therefore matches a training record on $(E, s_{\mathrm{main}}, \mathcal{S}_{\mathrm{co}}, \mathbf{c})$ and on y. Sequences are a separate matter: the same enzyme is often characterized in more than one article, so an external record can still carry a sequence that the training data contains. To see how far that goes, each unique external sequence was

searched against the unique sequences of the corresponding training data with MMseqs2. Identity was taken as the number of identical aligned residues over the length of the shorter sequence, and the maximum over all hits is reported.

A sequence that also appears in training is carried by 37 of the 485 k_{cat} records (7.6%), 40 of 502 K_m records (8.0%) and 18 of 470 k_{cat}/K_m records (3.8%). Of these, 11, 15 and 2 records share both E and s_{main} with a training record. Table D1 gives the distribution of maximum identity. The three sets differ in how they are spread: the K_m records sit closest to the training data, with a median maximum identity of 64%, while k_{cat} and k_{cat}/K_m are more dispersed, at 45% and 42%.

Table D1: External test records by maximum sequence identity to the training data of the same parameter.

Maximum identity	k_{cat}	K_m	k_{cat}/K_m
$< 30\%$	159	80	164
30–50%	119	128	123
50–80%	93	104	106
80–99%	34	92	36
$\geq 99\%$	80	98	41
Total	485	502	470