

Metabolic engineering and deep learning-driven protein engineering for N-Acetylneuraminic acid biosynthesis in *Escherichia coli*

Corresponding Author: Professor Jin-Song Shi

This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

The manuscript "Multi-modular metabolic engineering and deep learning-driven protein engineering for NeuAc biosynthesis in *Escherichia coli*" presents a novel method for integrating computational and deep learning methods into a metabolic engineering framework. This framework is used to design a high-titer NeuAc-producing *Escherichia coli* strain. Deep learning methods were used to identify enzymes with specific functions and those with potentially enhanced catalytic activity. The results are impressive, significantly increasing the yield of the engineered strain. However, we have some concerns regarding missing details of the developed deep learning methods, the choice of prediction methods, and the necessity and importance of integrating some machine learning methods.

Major concerns:

1. Choice of Prediction Methods: The authors selected methods based on the DLKcat method to obtain predictions for kcat (lines 161–165) and to develop their own method for kcat/KM (lines 247–250). However, studies have shown that DLKcat cannot accurately predict the effects of mutants, and that it is no better than a naive, similarity-based approach for wild types (doi.org/10.1093/biomethods/bpae061). Furthermore, several superior approaches have been proposed in recent years that significantly outperform DLKcat (e.g., doi.org/10.1038/s41467-023-39840-4 and doi.org/10.1016/j.checat.2024.101094). Thus, a different prediction model and model architecture would have been more suitable. This should at least be discussed in the manuscript.

2. Missing validation of DLCatalysis model performance. The authors developed their own kcat/KM prediction model, DLCatalysis. The only validation they mention in the main text is using two data points, which is clearly insufficient. Moreover, it is unclear whether these points were included in the training set (lines 251–254). Typically, a dataset is divided into training, validation, and test sets before training a machine learning model. This provides unseen data with which to validate the model. In this case, this would have resulted in hundreds of unseen data points, which would have provided insight into the model's performance. Without this, it is unclear how well the model performs and how it generalizes to unseen data. Figure S2a is the only place where a validation and test set are mentioned. So if a validation and test set exist, how were they generated? Splitting a dataset naively or poorly can lead to seemingly high model performance without the model actually learning much (see doi.org/10.1093/biomethods/bpae061).

3. There is hardly any information about the developed DLCatalysis model in the main text or the Methods section. Only Supplementary Figures S1 and S2 provide hints about the training details. For example, for the dataset processing, it is only mentioned that multiple rounds of cleaning were performed (ll. 524–525). What does this mean? Further, what is the final model's exact architecture? How were the hyperparameters of the model selected? What is the model's performance? Will the authors make their code and data available?

4. Why did the authors develop their own KCat/KM prediction model? EITLEM and UniKP, for example, are two recently developed kcat/KM deep learning prediction models (<https://doi.org/10.1016/j.checat.2024.101094>;

<https://www.nature.com/articles/s41467-023-44113-1>). Why did the authors feel the need to develop their own model? And how does their model compare to those pre-existing ones?

5. The authors state that deep learning played an important role in the entire process (lines 332-337). However, given the reasons I mentioned above, I am highly sceptical that the DLKcat predictions of kcat obtained via GotEnzymes or the DLCatalysis model contributed much to the success of the approach. In fact, I would rather suspect that the traditional computational tools played a more important role in selecting appropriate enzyme candidates. I suggest that the authors either provide clearer evidence of the importance of integrating deep learning methods or offer a more nuanced discussion of their importance.

Minor concerns:

6. In the subsection "Mining and screening pathway enzymes for NeuAc biosynthesis" the authors search for enzymes with high kcat values to identify those with potentially higher catalytic efficiency. Why don't the authors use kcat/KM, which they use later in the study? This should provide a clearer picture of catalytic efficiency than kcat alone.

7. Novelty of CD-IGM algorithm: The authors describe their approach of integrating computational methods and deep learning with iterative group mutations as novel. Studies that have already implemented similar approaches exist and should be discussed (e.g. <https://doi.org/10.1016/j.scib.2023.09.039> or <https://doi.org/10.1039/D4CS00196F>).

8. Figure S2b jointly plots the performance of the training, validation, and test data. However, plotting the training data here is misleading because deep learning models easily overfit to training data, which does not indicate good model performance.

To fairly evaluate the model, I suggest dividing the test set into different categories based on the similarity of the substrates and enzymes to the training set. Then, compute performance metrics separately for these subsets.

(Remarks on code availability)

Reviewer #2

(Remarks to the Author)

The authors integrated deep learning into a metabolic engineering framework to design a high-titer N-acetylneuraminic acid (NeuAc)-producing *Escherichia coli* strain. The work identified N-acetylglucosamine 2-epimerase (AGE) as the rate-limiting enzyme, utilized deep learning to develop high-performance AGE variants, and leveraged the CDL-IGM platform to enhance their catalytic activity. Moreover, it discovered and addressed the reabsorption restriction associated with the NeuAc transporter gene *exuT*, leading to a NeuAc yield of 71.2 g/L in a 5-L fermenter, the highest concentration reported to date.

The study features substantive content with a reasonably clear logical structure, and its findings provide significant advancements in NeuAc biosynthesis. This work is well-suited for publication in a specialized journal. However, I consider the innovation insufficient for Nature Communications and recommend rejection.

Major concerns underpinning this decision are as follows:

1. The authors' identification of N-acetylglucosamine 2-epimerase (AGE) as the rate-limiting enzyme does not represent a novel finding, as definitively established by Pang et al. (J. Agric. Food Chem. 2020, 68(42), 11758–11764).

2. The study merely combines existing deep learning tools, exhibiting insufficient novelty.

3. The description of the deep learning model developed in this study is unclear, and the model is limited to one enzyme only in the NeuAc synthesis pathway, lacking extensibility.

4. The data analysis throughout the article lacks depth (e.g., in lines 376-385, significant differences in NeuAc yield and byproducts were observed between the two carbon source feeding strategies. The authors suggest glycerol provides more PEP (phosphoenolpyruvate), but no specific analysis was conducted).

(Remarks on code availability)

Reviewer #3

(Remarks to the Author)

The paper is well written and the aims, objectives and approach are clear. The figures are well presented. The methodology is clear. The results are interesting and the productivity is impressive.

I have a few queries below. While the pipeline was successful in selecting improved enzyme variants, the authors still only selected a small number, 16 in total. How can the authors be sure that these were the best variants. Did the authors consider a HTP approach to generating more variants? This could have enabled them to expand the number of mutants and expand

their pipeline. This could also have been used to improve the model further.

While the fermentation was successful it focused primarily on using glucose and glycerol as a feedstock-did the authors consider using any industrial feedstocks-such as molasses's.

What is the market for N-acetylneuraminic acid? It would be useful to understand the market size.

It would also have been useful to have a comparison between current production methods and the method detailed here-to highlight why this would be a better method for productivity.

It would be useful to understand how the N-acetylneuraminic acid would be extracted from the supernatant.

While I appreciate that this is a demonstration of the platform-considering scale it would be useful to understand the rationale for using an inducible promoter and fed batch fermentation over batch and a constitutive promoter.

It would have been useful to compare the productivity of each improved strain on an OD basis-graphical figure.

Figure 6a the legend states glycerol but the figure states glucose.

(Remarks on code availability)

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

Unfortunately, most of the authors' responses do not sufficiently address our major previous concerns. Below, we outline the remaining issues for each response in detail. While the method developed by the authors has clearly resulted in a substantial improvement in the yield of the engineered *E. coli* strain and represents an important contribution, it remains essential that the limitations and potential weaknesses of both the employed and newly developed approaches are discussed transparently. Several of the current claims are either incorrect, insufficiently supported, or require further analysis and clarification. In its current form, the manuscript does not yet meet the scientific rigor expected for publication in a high-impact journal such as *Nature Communications*.

Response to A1:

While it may have been true that at the time of development, DLKcat was among the few available tools, it is now well established that DLKcat performs very poorly for predicting mutation effects and is no better than a naïve baseline for wildtype predictions. Even if these limitations were not evident at the time of development, they are now clear and should be explicitly acknowledged and discussed in the manuscript. However, the authors still do not adequately address these limitations. This is essential for understanding the weaknesses of the current approach and for identifying directions for future improvement.

Response to A2:

Splitting folds at random creates massive overlap between training and test sequences. For example, in the split EITLEM folds, 77% of the sequences in the kcat test set also appear in the kcat training set. More importantly, for 25% of the test set, entries share both the same sequence and identical kcat values with entries in the training set; they differ only in the substrate, representing the same reaction with two substrates. Such data leakage will occur for every simply randomly split enzyme constant dataset. Careful splitting is thus crucial for a fair evaluation. These limitations need to be discussed.

Response to A4:

The large discrepancies in model performance between DLcatalysis, UniKP, and EITLEM appear implausible. The reported R2 value for DLcatalysis is nearly three times higher than that of EITLEM. However, the EITLEM architecture is more advanced and has previously shown substantially better performance than the DLKcat architecture on a closely related task when trained and evaluated on exactly the same data (see <https://doi.org/10.1016/j.checat.2024.101094>, Figure 7A). Moreover, EITLEM leverages transfer learning from kcat and KM to kcat/KM prediction, which provides an additional advantage over models that do not employ such techniques. In the original publication, EITLEM achieved a test R2 above 0.8, which is more than four times higher than the value reported here. The authors should provide a clear explanation for this substantial discrepancy, as random differences in data splits alone are unlikely to account for such variation. The same arguments apply to the unexpectedly low R2 reported for UniKP.

It is unrealistic and scientifically inappropriate to claim the development of a new state-of-the-art kcat/KM predictor without further justification and analysis, especially given the use of less sophisticated model architectures and training strategies compared to recent models.

Finally, it is important that comparing models trained on different datasets using the same test set is problematic, as each model may exhibit different forms of data leakage. A possible solution would be to evaluate each model's performance across test subsets stratified by sequence similarity to their respective training sets, as shown in <https://doi.org/10.1016/j.checat.2024.101094>, Figure 7A.

Response to Q5:

The authors provide no justification for their belief that 162 experiments would have been necessary to obtain 12 positive variants without deep learning.

Response to Q6:

If the authors chose not to use DLKcat at this stage due to its limited capabilities, it is unclear why it was still employed elsewhere in the pipeline. Additionally, we don't understand what is meant by "eliminating interference". We also still do not understand why the authors did not apply their own kcat/KM prediction models at this stage of the workflow. The criteria for deciding when to integrate these prediction models into enzyme screening and when to omit them should be clearly explained and justified.

Reviewer #2

(Remarks to the Author)

The authors have submitted a revised manuscript that thoughtfully addresses the majority of the concerns raised during the initial review. The significant enhancements, particularly the detailed description of the DLCatalysis model architecture, hyperparameter optimization, and its comparative benchmarking against state-of-the-art tools like UniKP and EITLEM, have substantially improved the manuscript's clarity and scientific rigor. The additional experimental data and deeper mechanistic analysis, such as the molecular docking and dynamics simulations explaining the enhanced activity of AGE variants, further strengthen the claims.

My primary concern regarding the novelty of identifying AGE as the bottleneck has been adequately countered by the authors, who correctly distinguish their systematic, quantitative metabolite flux analysis from prior indirect evidence. The strategic integration of existing deep learning and computational tools into a novel, efficient workflow (CD-IGM) for enzyme engineering is now more clearly framed as a valid and impactful contribution, moving the field forward through intelligent synthesis rather than solely through the creation of entirely new algorithms.

Consequently, I am pleased to reconsider my initial assessment. The work presents a comprehensive and highly successful metabolic engineering campaign, achieving a record-breaking titer of NeuAc. The integration of deep learning, while not constituting a fundamental algorithmic breakthrough, has been demonstrated to be functionally critical in drastically reducing the experimental burden and guiding effective decisions. The manuscript, in its current form, makes a valuable contribution and is now much stronger and suitable for publication in a journal of your caliber.

To further elevate the impact and general significance of this work, I have one major suggestion for the authors to consider as a clear direction for future research:

The study powerfully validates the proposed deep learning-driven pipeline for a single, critical enzyme (AGE) within one specific biosynthetic pathway. The true potential and robustness of such a computational-metabolic engineering framework would be profoundly amplified by demonstrating its extensibility. A compelling way to achieve this would be to apply the same integrated pipeline (e.g., the CLEAN-DLCatalysis screening followed by CD-IGM optimization) to a different rate-limiting enzyme in a distinct product synthesis pathway in a future study. For instance, applying this methodology to enhance a key reductase in astaxanthin biosynthesis or a cytochrome P450 in plant natural product synthesis would provide critical evidence that the framework is not a one-off solution but a generalizable platform technology. Such a demonstration would unequivocally establish its broad applicability in microbial cell factory development.

I recommend that the authors include a brief discussion on this perspective of future work and extensibility in the manuscript's conclusion or discussion section. This will help readers immediately grasp the wider implications and future trajectory of this research.

Therefore, I recommend acceptance of the manuscript pending minor revisions to incorporate the aforementioned point on future extensibility validation.

Specific minor point:

The comparison of model performance (R^2 and r.m.s.e. values) between DLCatalysis, UniKP, and EITLEM in Supplementary Fig. S2c-d is a crucial addition. Please ensure the figure caption explicitly states the size and nature of the test set used for this comparative benchmarking to provide full context.

Reviewer #3

(Remarks to the Author)

I would like to thank the authors for addressing my comments, I am satisfied with the responses and recommend publication.

Reviewer #4

(Remarks to the Author)

I co-reviewed this manuscript with one of the reviewers who provided the listed reports. This is part of the Nature Communications initiative to facilitate training in peer review and to provide appropriate recognition for Early Career Researchers who co-review manuscripts.

Version 2:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

The authors have made a great effort to address all of our concerns. The new experimental work, redesigned pipeline and methods have strengthened the manuscript, which we now consider suitable for publication.

(Remarks on code availability)

Reviewer #4

(Remarks to the Author)

I co-reviewed this manuscript with one of the reviewers who provided the listed reports. This is part of the Nature Communications initiative to facilitate training in peer review and to provide appropriate recognition for Early Career Researchers who co-review manuscripts.

(Remarks on code availability)

Open Access This Peer Review File is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

In cases where reviewers are anonymous, credit should be given to 'Anonymous Referee' and the source.

The images or other third party material in this Peer Review File are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

To view a copy of this license, visit <https://creativecommons.org/licenses/by/4.0/>

Reviewer comments:

Reviewer #1

(Remarks to the Author):

The manuscript "Multi-modular metabolic engineering and deep learning-driven protein engineering for NeuAc biosynthesis in *Escherichia coli*" presents a novel method for integrating computational and deep learning methods into a metabolic engineering framework. This framework is used to design a high-titer NeuAc-producing *Escherichia coli* strain. Deep learning methods were used to identify enzymes with specific functions and those with potentially enhanced catalytic activity. The results are impressive, significantly increasing the yield of the engineered strain. However, we have some concerns regarding missing details of the developed deep learning methods, the choice of prediction methods, and the necessity and importance of integrating some machine learning methods.

Major concerns:

1. Choice of Prediction Methods: The authors selected methods based on the DLKcat method to obtain predictions for kcat (lines 161–165) and to develop their own method for kcat/KM (lines 247–250). However, studies have shown that DLKcat cannot accurately predict the effects of mutants, and that it is no better than a naive, similarity-based approach for wild types (doi.org/10.1093/biomethods/bpae061). Furthermore, several superior approaches have been proposed in recent years that significantly outperform DLKcat (e.g., doi.org/10.1038/s41467-023-39840-4 and doi.org/10.1016/j.checat.2024.101094). Thus, a different prediction model and model architecture would have been more suitable. This should at least be discussed in the manuscript.

2. Missing validation of DLCatalysis model performance. The authors developed their own kcat/KM prediction model, DLCatalysis. The only validation they mention in the main text is using two data points, which is clearly insufficient. Moreover, it is unclear whether these points were included in the training set (lines 251–254).

Typically, a dataset is divided into training, validation, and test sets before training a machine learning model. This provides unseen data with which to validate the model. In this case, this would have resulted in hundreds of unseen data points, which would have provided insight into the model's performance. Without this, it is unclear how well the model performs and how it generalizes to unseen data.

Figure S2a is the only place where a validation and test set are mentioned. So if a validation and test set exist, how were they generated? Splitting a dataset naively or poorly can lead to seemingly high model performance without the model actually learning much (see doi.org/10.1093/biomethods/bpae061).

3. There is hardly any information about the developed DLCatalysis model in the main text or the Methods section. Only Supplementary Figures S1 and S2 provide hints about the training details. For example, for the dataset processing, it is only mentioned that multiple rounds of cleaning were performed (ll. 524–525). What does this mean? Further, what is the final model's exact architecture? How were the hyperparameters of the model selected? What is the model's performance? Will the authors make their code and data available?

4. Why did the authors develop their own KCat/KM prediction model? EITLEM and UniKP, for

example, are two recently developed kcat/KM deep learning prediction models (<https://doi.org/10.1016/j.checat.2024.101094>; <https://www.nature.com/articles/s41467-023-44113-1>). Why did the authors feel the need to develop their own model? And how does their model compare to those pre-existing ones?

5. The authors state that deep learning played an important role in the entire process (lines 332-337). However, given the reasons I mentioned above, I am highly sceptical that the DLKcat predictions of kcat obtained via GotEnzymes or the DLCatalysis model contributed much to the success of the approach. In fact, I would rather suspect that the traditional computational tools played a more important role in selecting appropriate enzyme candidates. I suggest that the authors either provide clearer evidence of the importance of integrating deep learning methods or offer a more nuanced discussion of their importance.

Minor concerns:

6. In the subsection “Mining and screening pathway enzymes for NeuAc biosynthesis” the authors search for enzymes with high kcat values to identify those with potentially higher catalytic efficiency. Why don't the authors use kcat/KM, which they use later in the study? This should provide a clearer picture of catalytic efficiency than kcat alone.

7. Novelty of CD-IGM algorithm: The authors describe their approach of integrating computational methods and deep learning with iterative group mutations as novel. Studies that have already implemented similar approaches exist and should be discussed (e.g. <https://doi.org/10.1016/j.scib.2023.09.039> or <https://doi.org/10.1039/D4CS00196F>).

8. Figure S2b jointly plots the performance of the training, validation, and test data. However, plotting the training data here is misleading because deep learning models easily overfit to training data, which does not indicate good model performance.

To fairly evaluate the model, I suggest dividing the test set into different categories based on the similarity of the substrates and enzymes to the training set. Then, compute performance metrics separately for these subsets.

Reviewer #2

(Remarks to the Author):

The authors integrated deep learning into a metabolic engineering framework to design a high-titer N-acetylneuraminic acid (NeuAc)-producing *Escherichia coli* strain. The work identified N-acetylglucosamine 2-epimerase (AGE) as the rate-limiting enzyme, utilized deep learning to develop high-performance AGE variants, and leveraged the CDL-IGM platform to enhance their catalytic activity. Moreover, it discovered and addressed the reabsorption restriction associated with the NeuAc transporter gene *exuT*, leading to a NeuAc yield of 71.2 g/L in a 5-L fermenter, the highest concentration reported to date.

The study features substantive content with a reasonably clear logical structure, and its findings provide significant advancements in NeuAc biosynthesis. This work is well-suited for publication in a specialized journal. However, I consider the innovation insufficient for *Nature Communications* and recommend rejection.

Major concerns underpinning this decision are as follows:

1. The authors' identification of N-acylglucosamine 2-epimerase (AGE) as the rate-limiting enzyme does not represent a novel finding, as definitively established by Pang et al. (J. Agric. Food Chem. 2020, 68(42), 11758–11764).
2. The study merely combines existing deep learning tools, exhibiting insufficient novelty.
3. The description of the deep learning model developed in this study is unclear, and the model is limited to one enzyme only in the NeuAc synthesis pathway, lacking extensibility.
4. The data analysis throughout the article lacks depth (e.g., in lines 376-385, significant differences in NeuAc yield and byproducts were observed between the two carbon source feeding strategies. The authors suggest glycerol provides more PEP (phosphoenolpyruvate), but no specific analysis was conducted).

Reviewer #3

(Remarks to the Author):

The paper is well written and the aims, objectives and approach are clear. The figures are well presented. The methodology is clear. The results are interesting and the productivity is impressive.

1. I have a few queries below. While the pipeline was successful in selecting improved enzyme variants, the authors still only selected a small number, 16 in total. How can the authors be sure that these were the best variants. Did the authors consider a HTP approach to generating more variants? This could have enabled them to expand the number of mutants and expand their pipeline. This could also have been used to improve the model further.
2. While the fermentation was successful it focused primarily on using glucose and glycerol as a feedstock—did the authors consider using any industrial feedstocks—such as molasses's.
3. What is the market for N-acetylneuraminic acid? It would be useful to understand the market size.
4. It would also have been useful to have a comparison between current production methods and the method detailed here—to highlight why this would be a better method for productivity.
5. It would be useful to understand how the N-acetylneuraminic acid would be extracted from the supernatant.
6. While I appreciate that this is a demonstration of the platform—considering scale it would be useful to understand the rationale for using an inducible promoter and fed batch fermentation over batch and a constitutive promoter.
7. It would have been useful to compare the productivity of each improved strain on an OD basis—graphical figure.
8. Figure 6a the legend states glycerol but the figure states glucose.

For your guidance, itemized answers to each reviewer's comments are appended below (answers are in blue font).

Response to reviewers' comments point by point:

Reviewer #1 (Remarks to the Author):

The manuscript "Multi-modular metabolic engineering and deep learning-driven protein engineering for NeuAc biosynthesis in *Escherichia coli*" presents a novel method for integrating computational and deep learning methods into a metabolic engineering framework. This framework is used to design a high-titer NeuAc-producing *Escherichia coli* strain. Deep learning methods were used to identify enzymes with specific functions and those with potentially enhanced catalytic activity. The results are impressive, significantly increasing the yield of the engineered strain. However, we have some concerns regarding missing details of the developed deep learning methods, the choice of prediction methods, and the necessity and importance of integrating some machine learning methods.

Q1

Choice of Prediction Methods: (1) The authors selected methods based on the DLKcat method to obtain predictions for k_{cat} (lines 161–165) and to develop their own method for k_{cat}/K_M (lines 247–250). However, studies have shown that DLKcat cannot accurately predict the effects of mutants, and that it is no better than a naive, similarity-based approach for wild types (doi.org/10.1093/biomethods/bpae061). (2) Furthermore, several superior approaches have been proposed in recent years that significantly outperform DLKcat (e.g., doi.org/10.1038/s41467-023-39840-4 and doi.org/10.1016/j.checat.2024.101094). Thus, a different prediction model and model architecture would have been more suitable. This should at least be discussed in the manuscript.

A1: We sincerely appreciate your insightful comments regarding the choice of prediction methods. Regarding the question 1, DLKcat might exhibit limitations in k_{cat} , which is essential to emphasize that enzyme performance cannot be evaluated solely by k_{cat} values. To address these limitations, we developed the DLcatalysis model for predicting $k_{cat} K_M^{-1}$ by training it on a new dataset. Furthermore, by incorporating the CLEAN model into enzyme mining, we have established a completely new workflow for enzyme mining. Critically, our experimental validation proves the pipeline's efficacy: While DLKcat alone failed to identify improved AGE variants, CLEAN-DLcatalysis discovered AGE10 with 81.2% higher $k_{cat} K_M^{-1}$ and BfM11 with 349.1% and 13.0% increase in enzyme activity, and solubility, demonstrating that our architecture overcomes base tool limitations through intelligent integration, not reliance on standalone accuracy of DLKcat. Regarding the second question, we had already completed the development of the DLcatalysis model at a time when DLKcat stood as the state-of-the-art deep learning model for predicting k_{cat} . Although numerous excellent models have emerged in recent years, we think that the value of a good model-based tool lies not in the complexity of its algorithms, but in the depth of its integration with experimental workflows, and we also believe that biologists might not only blindly pursue sophisticated algorithms and computational power, but also ingeniously integrated diverse functional models into

a new workflow and combined them with computational design to enhance the success rate of experimental outcomes. For discussing with other models, we have rewritten **Introduction** section as follows: *“In recent years, the field of enzyme kinetics prediction has witnessed significant advancements with the emergence of sophisticated deep learning frameworks. Among these, UniKP based on pretrained language models has gained prominence for its ability to predict key enzyme kinetic parameters directly from protein sequences and substrate structures. EITLEM-kinetics employs transfer learning techniques to enhance the predictive capability of the $k_{cat}K_m^{-1}$ model. The next steps should focus on refining and implementing these deep learning models to advance enzyme engineering design, using experimental feedback loops to improve prediction accuracy and boost experimental success rates.”* Please see Line 68-76 in the revised manuscript.

Q2

Missing validation of DLCatalysis model performance. The authors developed their own k_{cat}/K_M prediction model, DLCatalysis. The only validation they mention in the main text is using two data points, which is clearly insufficient. Moreover, it is unclear whether these points were included in the training set (lines 251–254).

Typically, a dataset is divided into training, validation, and test sets before training a machine learning model. This provides unseen data with which to validate the model. In this case, this would have resulted in hundreds of unseen data points, which would have provided insight into the model's performance. Without this, it is unclear how well the model performs and how it generalizes to unseen data.

Fig. S2a is the only place where a validation and test set are mentioned. So, if a validation and test set exist, how were they generated? Splitting a dataset naively or poorly can lead to seemingly high model performance without the model actually learning much (see doi.org/10.1093/biomet/bpae061).

A2: We apologize for the confusing expression. As the dataset partitioning methodology was not explicitly specified in the original manuscript, we have added a detailed description of the data splitting approach in the Methods section. Please see line 659-660 in the manuscript: *“Datasets were initially shuffled and randomly partitioned into training, validation, and test sets at an 80:10:10 ratio.”*

Q3

There is hardly any information about the developed DLCatalysis model in the main text or the Methods section. Only Supplementary Figures S1 and S2 provide hints about the training details. For example, for the dataset processing, it is only mentioned that multiple rounds of cleaning were performed (ll. 524-525). What does this mean? Further, what is the final model's exact architecture? How were the hyperparameters of the model selected? What is the model's performance? Will the authors make their code and data available?

A3: We apologize for confusing expression and missing information. For our data processing, we have rewritten the method in the **Method** section: *“The data processing was using previously described method. Briefly, the dataset was compiled from BRENDA and SABIO-RK databases, using custom API scripts. The collection includes substrate names, organism information, EC numbers, UniProt IDs, enzyme classifications, and catalytic efficiency ($k_{cat}K_m^{-1}$) measurements. Due to*

inconsistent reporting of experimental conditions in source databases, we excluded pH and temperature variables to maintain dataset breadth. Substrate structures were represented as canonical SMILES strings obtained through PubChem queries, with a normalization process ensuring consistent representation across naming variations. Protein sequences were retrieved either directly via UniProt IDs or indirectly using EC numbers and organism data. Wild-type enzyme sequences were analyzed in their native conformation, while mutated variants were adjusted according to documented substitution sites. Following rigorous quality control procedures, the final dataset comprised high-confidence enzyme-substrate pairs suitable for deep learning model development.” Please see Line 595-606 in the manuscript.

The detail of constructing the deep learning pipeline would be present in the **Methods** section. The added deep learning model construction, please see lines 609-669 in the manuscript: “The DLcatalysis model was trained on a newly curated dataset using DLkcat methodology¹. Briefly, deep learning approach was developed for k_{cat}/K_m^{-1} prediction by coupling a graph neural network (GNN) for substrate representation with a convolutional neural network (CNN) for protein sequence. In this framework, substrate molecules are encoded as molecular graphs with atoms constituting nodes and chemical bonds defining edges, while proteins are represented as linear sequences of amino acid residues.

Substrates exhibit limited atomic diversity, primarily comprising carbon and hydrogen atoms, with chemical bonds restricted to single and double bond types. To enhance model capacity, we generated vector representations using r-radius subgraphs induced by nodes and edges within distance r from each atom node. Substrate SMILES strings were first converted to molecular graphs using RDKit (version 2020.09.1). The graph neural network iteratively updates atom node representations through message passing, where neighboring node embeddings are transformed via nonlinear activation functions including ReLU. Two distinct message passing mechanisms propagate local structural information: node-to-node transitions aggregate atomic features, while edge-aware transitions incorporate bond information. This iterative process yields comprehensive molecular vector representations.

CNN were applied to extract low-dimensional vector representations from protein sequences through nonlinear transformations, such as the rectified linear unit (ReLU) activation function. To enable CNN-based sequence modelling, each protein sequence was decomposed into short, fixed-length segments of amino acids with $n = 1, 2$, or 3 . These segments were generated by shifting a window of length n one residue at a time along the sequence, thereby preserving local sequential patterns. To minimize the impact of rare or infrequent motifs during representation learning, only small values of n were considered. The resulting segments were then mapped to distributed vector representations, known as word embeddings. Subsequently, the CNN applied a filter operation, as defined in equation (1), to compute hidden activations from the embedded inputs and trainable weight matrices. This process produced a series of feature maps corresponding to local sequence contexts captured by the sliding window approach.

$$\mathbf{c}_i^{(t)} = f \left(\mathbf{W}_{\text{conv}} \mathbf{c}_i^{(t-1)} + \mathbf{b}_{\text{conv}} \right) \quad (1)$$

where f denotes a nonlinear activation function such as ReLU; $\mathbf{W}_{\text{conv}}$ represents the

convolutional filter weights and $\mathbf{b}_{\text{conv}}$ is the bias vector. The hidden state vectors $\mathbf{c}_i^{(t)}$ and $\mathbf{c}_i^{(t-1)}$ represent sequential protein sequence embeddings at consecutive processing steps t and $t-1$, where i denotes the spatial position within the sequence.

Hyperparameter optimization systematically evaluated key architectural configurations (Supplementary Fig. S1): convolutional layer depth (2, 3 or 4), GNN propagation steps (2, 3 or 4), learning rate (1^{-e^2} , 1^{-e^3} or 1^{-e^4}), r -radius (1, 2 or 3), and embedding dimensionality (20, 30 or 50), others are fixed. These configurations were assessed using the coefficient of determination (R^2 ; equation (2)) during hyperparameter tuning to identify the optimal architecture for predictive performance. R^2 values were computed using scikit-learn (v0.23.2). The final model was trained using the configuration yielding maximum R^2 on the validation set.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i^{\text{exp}} - y_i^{\text{pred}})^2}{\sum_{i=1}^n (y_i^{\text{exp}} - \bar{y})^2} \quad (2)$$

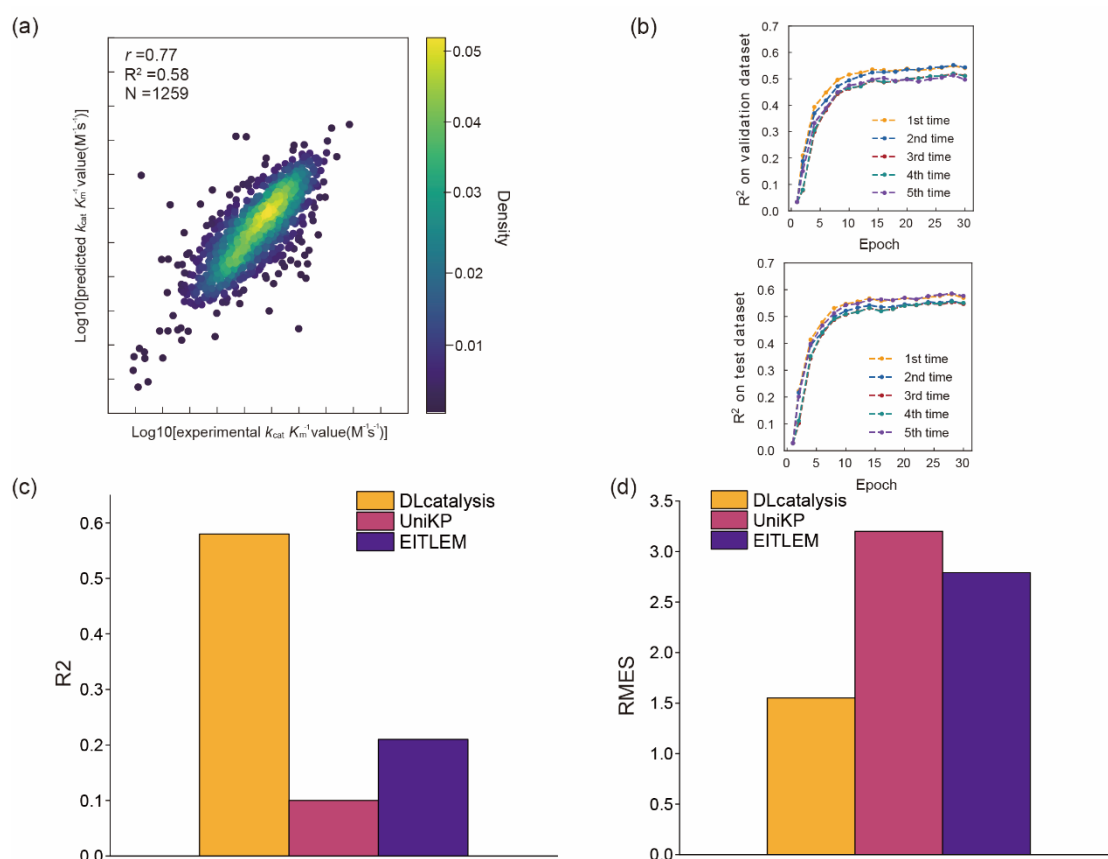
Where y_i^{exp} and y_i^{pred} denote the experimentally and predicted determined catalytic efficiency constant ($k_{\text{cat}} K_m^{-1}$) values for the i -th observation. $\bar{y}$ represents the mean of experimental $k_{\text{cat}} K_m^{-1}$ values, and n signifies the sample size of the evaluation set (validation or test dataset).

Substrate molecular vector representations and protein sequence embeddings were concatenated with catalytic rate constant ($k_{\text{cat}} K_m^{-1}$) values as regression targets to train the neural attention model. Datasets were initially shuffled and randomly partitioned into training, validation, and test sets at an 80:10:10 ratio. Model training optimized the loss function using substrate–protein pairs and corresponding $k_{\text{cat}} K_m^{-1}$ values from the training set. The optimal model was selected based on minimal root-mean-square error (r.m.s.e; equation 3) on the validation set while exhibiting minimized performance divergence between training and validation metrics. All models were implemented in PyTorch (version 1.4.0) with Python 3.7.6 under CUDA 10.1.243, while data processing utilized NumPy 1.20.2, SciPy 1.5.2, and pandas 1.1.3. Visualization was performed using Matplotlib 3.3.2 and seaborn 0.11.0.

$$\text{r.m.s.e} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i^{\text{exp}} - y_i^{\text{pred}})^2} \quad (3)$$

where y_i^{pred} is the predicted $k_{\text{cat}} K_m^{-1}$ value, y_i^{exp} is the experimental $k_{\text{cat}} K_m^{-1}$ value and n is the total number of items in the dataset (validation dataset or test dataset).”

Additionally, we have rewritten the sentence of hyperparameters parameters selection and the model performance in the **Result** section of the manuscript: “The impact of hyperparameters on deep learning performance was evaluated using learning curves. (Supplementary Fig. S1) Under the selected optimal parameters (substrate subgraph topological order $r = 2$; amino acid n -gram length = 3; vector dimensionality = 20; GNN time steps = 3; CNN layers = 3; learning rate = 1×10^{-4}), the model achieved a root mean square error (r.m.s.e.) of 1.55 for $k_{\text{cat}} K_m^{-1}$ predictions, with DLcatalysis attaining an R^2 of 0.58.(Supplementary Fig. S2)”. Please see Line 257-263.



Supplementary Fig. S2. Deep learning model performance for $k_{cat} K_m^{-1}$ prediction. (a) Performance of the final deep learning model. The correlation between predicted $k_{cat} K_m^{-1}$ values and those present in the test dataset was evaluated. The brightness of colour represents the density of data points. (b) validation dataset and test dataset by running five times of random splitting. (c) Comparison of R^2 value of our model, UniKP model, and EITLEM model with our test data. (d) Comparison of r.m.s.e value of our model, UniKP model, and EITLEM model with our test data. Please see Line 64-71 in **Supporting Information** file.

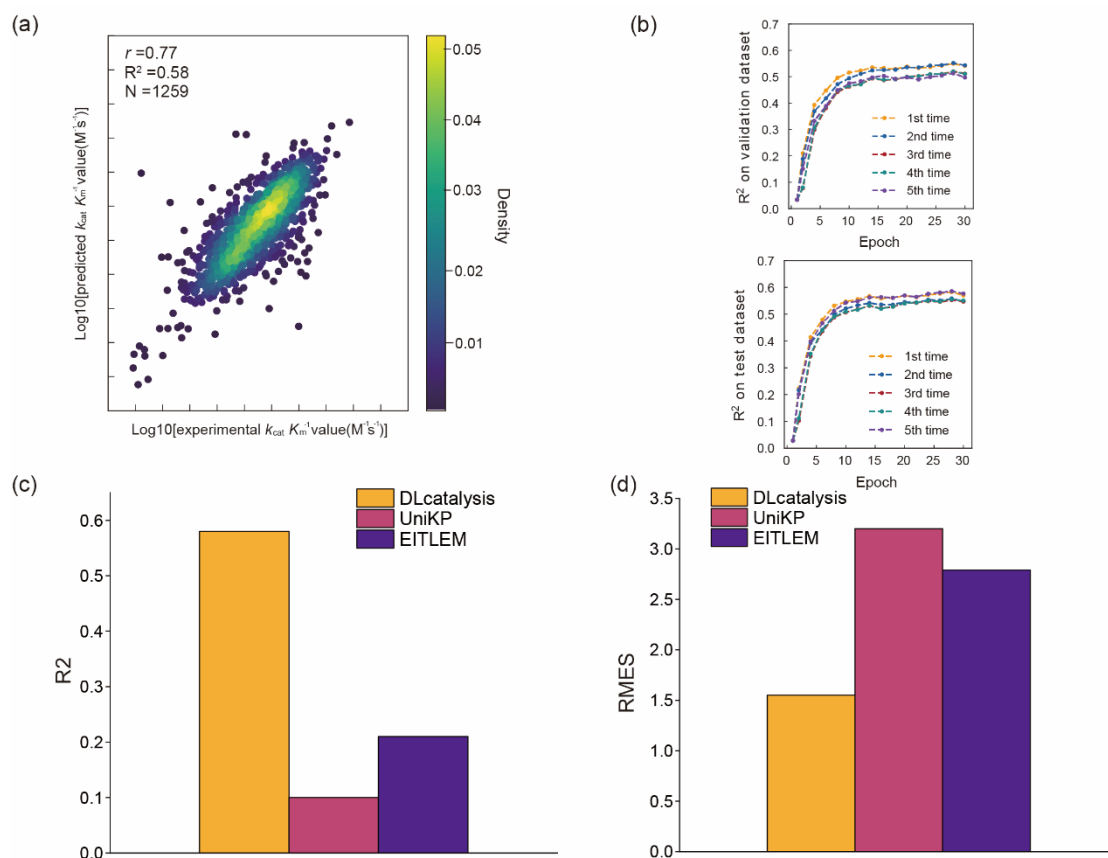
Given that DLkcat was retrained using the original implementation without code modification, and since the DLkcat framework has already been open-sourced, we consider it redundant to release the code again. Instead, we will provide the dataset for public access.

Q4

Why did the authors develop their own KCat/KM prediction model? EITLEM and UniKP, for example, are two recently developed kcat/KM deep learning prediction models (<https://doi.org/10.1016/j.checat.2024.101094>; <https://www.nature.com/articles/s41467-023-44113-1>). Why did the authors feel the need to develop their own model? And how does their model compare to those pre-existing ones?

A4: Thanks for your valuable suggestion. We finished the DLcatalysis for k_{cat}/K_m prediction earlier, and EITLEM and UniKP were not yet available at that time. EITLEM and UniKP currently represent the state-of-the-art methods for predicting $k_{cat} K_m^{-1}$ values. However, the EITLEM has sequence length limitation (< 1024 residues) necessitated the exclusion of enzymes with sequences exceeding this threshold. Furthermore, due to its computational complexity, EITLEM operates at a

significantly slower speed, rendering it unsuitable for large-scale prediction tasks where computational efficiency is critical, such as identifying enzymes with maximum activity within protein families, which incurs prohibitively high time costs. For UniKP, it's also a robust model, while the training data just 910 items. For discussing with other models, we have added the comparison of model's performance in **Result** section as follows: *“Under the selected optimal parameters (substrate subgraph topological order $r = 2$; amino acid n -gram length = 3; vector dimensionality = 20; GNN time steps = 3; CNN layers = 3; learning rate = 1×10^{-4}), the model achieved a root mean square error (r.m.s.e.) of 1.55 for $k_{cat} K_m^{-1}$ predictions, with DLcatalysis attaining an R^2 value of 0.58. (Supplementary Fig. S2). UniKP² and EITLEM³ are currently supposed to represent the state-of-the-art methods for predicting $k_{cat} K_m^{-1}$ values. When benchmarked against these models, DLcatalysis demonstrated superior performance, as UniKP and EITLEM models achieved R^2 values of 0.10 and 0.21, with corresponding r.m.s.e. values of 3.20 and 2.79, respectively (Supplementary Fig. S2c-d). Thus, DLcatalysis offers an advantage based on model performance in predicting $k_{cat} K_m^{-1}$.”* Please see Line 259-268 in the manuscript.



Supplementary Fig. S2. Deep learning model performance for $k_{cat} K_m^{-1}$ prediction. (a) Performance of the final deep learning model. The correlation between predicted $k_{cat} K_m^{-1}$ values and those present in the test dataset was evaluated. The brightness of colour represents the density of data points. (b) validation dataset and test dataset by running five times of random splitting. (c) Comparison of R^2 value of our model, UniKP model, and EITLEM model with our test data. (d) Comparison of r.m.s.e value of our model, UniKP model, and EITLEM model with our test data. Please see Line 64-71 in Supporting Information file.

The authors state that deep learning played an important role in the entire process (lines 332-337). However, given the reasons I mentioned above, I am highly sceptical that the DLKcat predictions of kcat obtained via GotEnzymes or the DLCatalysis model contributed much to the success of the approach. In fact, I would rather suspect that the traditional computational tools played a more important role in selecting appropriate enzyme candidates. I suggest that the authors either provide clearer evidence of the importance of integrating deep learning methods or offer a more nuanced discussion of their importance.

A5: Thank you for your comments. The primary value of deep learning models in experimental guidance lies in enhancing efficiency at specific stages, rather than determining experimental success or failure. Without deep learning, we could also complete experiments through empirical approaches, but this would incur significantly higher resource consumption and labor costs, which is universally undesirable. In this study, deep learning successfully identified superior enzymes and optimized variants, with our design strategy prioritizing traditional computational methods as the foundation while leveraging deep learning as a complementary tool. This approach yielded 12 positive variants from 70 mutagenesis candidates, effectively resolving our bottleneck. Crucially, relying solely on traditional computation would have required approximately 162 experimental iterations, doubling both financial and labor expenditures, while likely achieving comparable outcomes. Thus, deep learning's indispensable contribution was strategic resource optimization, not theoretical superiority. We added the following discussion in **Discussion** section: *"As demonstrated in our workflow, DLCatalysis reduced mutagenesis screening from an estimated 162 variants (required by traditional methods) to 70 candidates, yielding 12 positive variants with 349.1% higher catalytic efficiency. Crucially, this 56.7% reduction in experimental burden directly translates to accelerated timeline and resource allocation for critical bottleneck resolution. We emphasize that deep learning's value lies not in replacing domain expertise, but in converting computational predictions into experimental priorities. This efficiency paradigm is particularly vital for data-scarce targets where traditional methods reach diminishing returns, ultimately making high-impact enzyme engineering accessible within realistic resource constraints."* Please see Line 516-524 in the manuscript.

Q6

In the subsection "Mining and screening pathway enzymes for NeuAc biosynthesis" the authors search for enzymes with high kcat values to identify those with potentially higher catalytic efficiency. Why don't the authors use kcat/KM, which they use later in the study? This should provide a clearer picture of catalytic efficiency than kcat alone.

A6: Thank you for your comments. Our approach is driven by experimental observations rather than theoretical preferences. Specifically, GlcNAc accumulation without significant ManNAc accumulation in the NeuAc pathway indicated that the bottleneck stems from low catalytic efficiency of AGE rather than other pathway enzymes. Recognizing DLkcat's limitations in activity prediction, we retrained the model on $k_{cat} K_m^{-1}$ datasets to develop DLCatalysis, and integrated the CLEAN model to eliminate interference, yielding BfAGE that successfully enhanced NeuAc production.

Q7

Novelty of CD-IGM algorithm: The authors describe their approach of integrating computational

methods and deep learning with iterative group mutations as novel. Studies that have already implemented similar approaches exist and should be discussed (e.g. <https://doi.org/10.1016/j.scib.2023.09.039> or <https://doi.org/10.1039/D4CS00196F>).

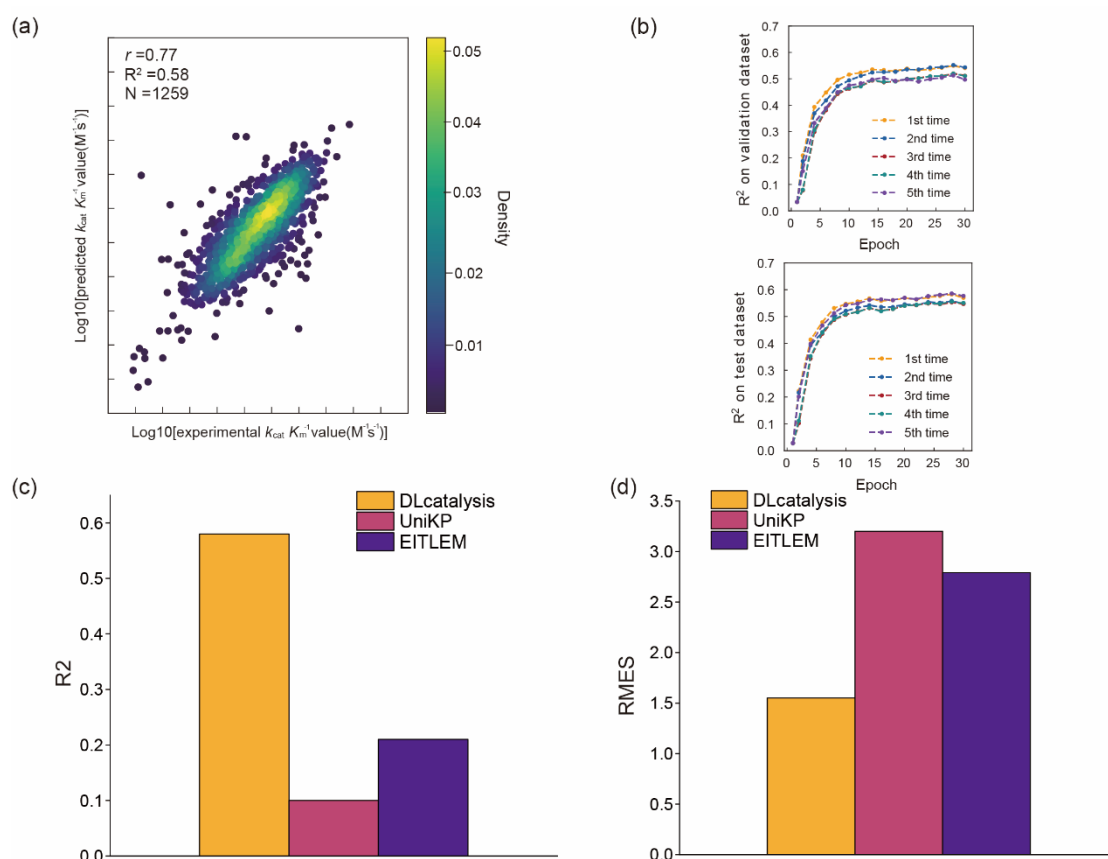
A7: Thank you for your suggestion. CD-IGM is an enzyme design framework that integrates deep learning with computational design. Its core innovation lies in combination of deep learning model, computational design, and semi-rational design based on 3D structure of protein, as amino acid residues in different spatial regions perform distinct functional roles. This workflow greatly improved the efficiency of protein design. By leveraging deep learning and computational design, CD-IGM accelerates the identification of hotspots for precision engineering. We have rewritten the following discussion in **Discission** section: *“As AI technology applications in enzyme engineering gain increasing prominence, successful cases of deep learning combined with computational design for guiding enzyme design are rapidly multiplying. Liu et al.⁴ developed the MECE platform for glycoside hydrolase design by integrating DeepGH with traditional conservation analysis. Ran et al.⁵ introduced the EnzyKR workflow that combines molecular docking with deep learning. To simultaneously address the weak catalytic efficiency and low soluble expression of AGE^{Bf}, this study employs PROSS and DeMaSk to identify potential key sites in AGE^{Bf}, followed by secondary screening using DLcatalysis to reduce experimental workload. The effective sites are then grouped based on the protein's three-dimensional structure, and through iterative group optimization, the mutant AGE^{BfM11} was obtained with a 349.1% increase in enzymatic activity and a 13% improvement in soluble expression.”* Please see Line 506-516 in the manuscript.

Q8

Fig. S2b jointly plots the performance of the training, validation, and test data. However, plotting the training data here is misleading because deep learning models easily overfit to training data, which does not indicate good model performance.

To fairly evaluate the model, I suggest dividing the test set into different categories based on the similarity of the substrates and enzymes to the training set. Then, compute performance metrics separately for these subsets.

A8: Thank you for your suggestion. For model performance based on test dataset, we revised the Supplementary Fig. S2a in the **Supplement Information** file.



Supplementary Fig. S2. Deep learning model performance for $k_{cat} K_m^{-1}$ prediction. (a) Performance of the final deep learning model. The correlation between predicted $k_{cat} K_m^{-1}$ values and those present in the test dataset was evaluated. The brightness of colour represents the density of data points. (b) validation dataset and test dataset by running five times of random splitting. (c) Comparison of R^2 value of our model, UniKP model, and EITLEM model with our test data. (d) Comparison of r.m.s.e value of our model, UniKP model, and EITLEM model with our test data. Please see Line 64-71 in Supporting Information file.

Reviewer #2 (Remarks to the Author):

The authors integrated deep learning into a metabolic engineering framework to design a high-titer N-acetylneuraminic acid (NeuAc)-producing Escherichia coli strain. The work identified N-acetylglucosamine 2-epimerase (AGE) as the rate-limiting enzyme, utilized deep learning to develop high-performance AGE variants, and leveraged the CDL-IGM platform to enhance their catalytic activity. Moreover, it discovered and addressed the reabsorption restriction associated with the NeuAc transporter gene exuT, leading to a NeuAc yield of 71.2 g/L in a 5-L fermenter, the highest concentration reported to date.

The study features substantive content with a reasonably clear logical structure, and its findings provide significant advancements in NeuAc biosynthesis. This work is well-suited for publication in a specialized journal. However, I consider the innovation insufficient for Nature Communications and recommend rejection.

Q1

The authors' identification of N-acetylglucosamine 2-epimerase (AGE) as the rate-limiting enzyme does not represent a novel finding, as definitively established by Pang et al. (J. Agric. Food Chem. 2020, 68(42), 11758–11764).

A1: Thank you for your detailed comments for our manuscript. Although the given study AGE and N-acetylneuraminic synthase (NeuB) as putative rate-limiting enzymes, critical re-examination of the cited literature reveals that the original work merely documented slr1975 from *Synechocystis* sp. PCC 6803 as exhibiting constrained enzymatic activity due to low ATP affinity and pyruvate-mediated inhibition⁶. Subsequently limitation was addressed by replacing it with AGE from *Anabaena* sp. CH1. Crucially, the study did not conclusively establish AGE as a genuine rate-limiting enzyme in the de novo biosynthetic pathway of N-acetylneuraminic acid (NeuAc). In contrast, this study systematically identifies rate-limiting steps through quantitative analysis of intermediate metabolite accumulation dynamics, which is a more rigorous and universally applicable framework for metabolic engineering strategies.

Q2

The study merely combines existing deep learning tools, exhibiting insufficient novelty.

A2: We appreciate your comment. The future research process will be assisted by multi models' system⁷, where we have developed a system that either combines two deep learning models or integrates deep learning models, advanced computational design and semi-rational design based on 3D structure for enzyme engineering. Moreover, existing deep learning frameworks themselves are often built by combining established algorithms. Following this same paradigm, we retrained the pre-existing DLKcat model to develop DLcatalysis, a specialized tool for predicting $k_{cat} K_m^{-1}$ values. To further enhance the success rate of enzyme mining, we integrated the enzyme classification model CLEAN, successfully identifying the superior enzyme eBfAGE. Subsequently, to improve both catalytic efficiency and soluble expression of BfAGE, we combined DLcatalysis, PROSS and DeMaSk, with rationally partitioning protein structures into regions, yielding the optimized mutant BfM11. This engineered variant significantly increased metabolic flux from GlcNAc to NeuAc. Notably, high-impact studies increasingly adopt existed deep learning model to create the new framework, such as using existed AlphaFold 2 to establish a structure-guided workflow for enzyme mining⁸ or combining existed software (Rosetta) and existed deep learning model (LigandMPNN) for de novo enzyme design⁹. This demonstrates that innovation in applied research lies not solely in creating entirely new tools, but in the strategic integration of existing methodologies to solve unmet challenges, a valid and impactful paradigm in modern computational biology.

Q3

The description of the deep learning model developed in this study is unclear, and the model is limited to one enzyme only in the NeuAc synthesis pathway, lacking extensibility.

A3: Thank you for your comments. The detail of constructing the deep learning pipeline would be present in the **Methods** section. The added development of deep learning model please see lines 609-669: "The DLcatalysis model was trained on a newly curated dataset using DLkcat methodology¹. Briefly, deep learning approach was developed for $k_{cat} K_m^{-1}$ prediction by coupling a graph neural network (GNN) for substrate representation with a convolutional neural network (CNN) for protein sequence. In this framework, substrate molecules are encoded as molecular

graphs with atoms constituting nodes and chemical bonds defining edges, while proteins are represented as linear sequences of amino acid residues.

Substrates exhibit limited atomic diversity, primarily comprising carbon and hydrogen atoms, with chemical bonds restricted to single and double bond types. To enhance model capacity, we generated vector representations using r-radius subgraphs induced by nodes and edges within distance r from each atom node. Substrate SMILES strings were first converted to molecular graphs using RDKit (version 2020.09.1). The graph neural network iteratively updates atom node representations through message passing, where neighboring node embeddings are transformed via nonlinear activation functions including ReLU. Two distinct message passing mechanisms propagate local structural information: node-to-node transitions aggregate atomic features, while edge-aware transitions incorporate bond information. This iterative process yields comprehensive molecular vector representations.

CNN were applied to extract low-dimensional vector representations from protein sequences through nonlinear transformations, such as the rectified linear unit (ReLU) activation function. To enable CNN-based sequence modeling, each protein sequence was decomposed into short, fixed-length segments of amino acids with $n = 1, 2$, or 3 . These segments were generated by shifting a window of length n one residue at a time along the sequence, thereby preserving local sequential patterns. To minimize the impact of rare or infrequent motifs during representation learning, only small values of n were considered. The resulting segments were then mapped to distributed vector representations, known as word embeddings. Subsequently, the CNN applied a filter operation, as defined in equation (1), to compute hidden activations from the embedded inputs and trainable weight matrices. This process produced a series of feature maps corresponding to local sequence contexts captured by the sliding window approach.

$$\mathbf{c}_i^{(t)} = f \left(\mathbf{W}_{\text{conv}} \mathbf{c}_i^{(t-1)} + \mathbf{b}_{\text{conv}} \right) \quad (1)$$

where f denotes a nonlinear activation function such as ReLU; $\mathbf{W}_{\text{conv}}$ represents the convolutional filter weights and $\mathbf{b}_{\text{conv}}$ is the bias vector. The hidden state vectors $\mathbf{c}_i^{(t)}$ and $\mathbf{c}_i^{(t-1)}$ represent sequential protein sequence embeddings at consecutive processing steps t and $t-1$, where i denotes the spatial position within the sequence.

Hyperparameter optimization systematically evaluated key architectural configurations: convolutional layer depth (2, 3 or 4), GNN propagation steps (2, 3 or 4), learning rate (10^{-2} , 10^{-3} or 10^{-4}), r -radius (1, 2 or 3), and embedding dimensionality (20, 30 or 50), others are fixed. These configurations were assessed using the coefficient of determination (R^2 ; equation (2)) during hyperparameter tuning to identify the optimal architecture for predictive performance. R^2 values were computed using scikit-learn (v0.23.2). The final model was trained using the configuration yielding maximum R^2 on the validation set.

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i^{\text{exp}} - y_i^{\text{pred}})^2}{\sum_{i=1}^n (y_i^{\text{exp}} - \bar{y})^2} \quad (2)$$

Where y_i^{exp} and y_i^{pred} denote the experimentally and predicted determined catalytic efficiency constant ($k_{\text{cat}} K_m^{-1}$) values for the i -th observation. $\bar{y}$ represents the mean of experimental $k_{\text{cat}} K_m^{-1}$ values, and n signifies the sample size of the evaluation set (validation or test dataset).

Substrate molecular vector representations and protein sequence embeddings were concatenated with catalytic rate constant ($k_{cat} K_m^{-1}$) values as regression targets to train the neural attention model. Datasets were initially shuffled and randomly partitioned into training, validation, and test sets at an 80:10:10 ratio. Model training optimized the loss function using substrate–protein pairs and corresponding $k_{cat} K_m^{-1}$ values from the training set. The optimal model was selected based on minimal root-mean-square error (r.m.s.e; equation 3) on the validation set while exhibiting minimized performance divergence between training and validation metrics. All models were implemented in PyTorch (version 1.4.0) with Python 3.7.6 under CUDA 10.1.243, while data processing utilized NumPy 1.20.2, SciPy 1.5.2, and pandas 1.1.3. Visualization was performed using Matplotlib 3.3.2 and seaborn 0.11.0.

$$\text{r.m.s.e} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i^{\text{exp}} - y_i^{\text{pred}})^2} \quad (3)$$

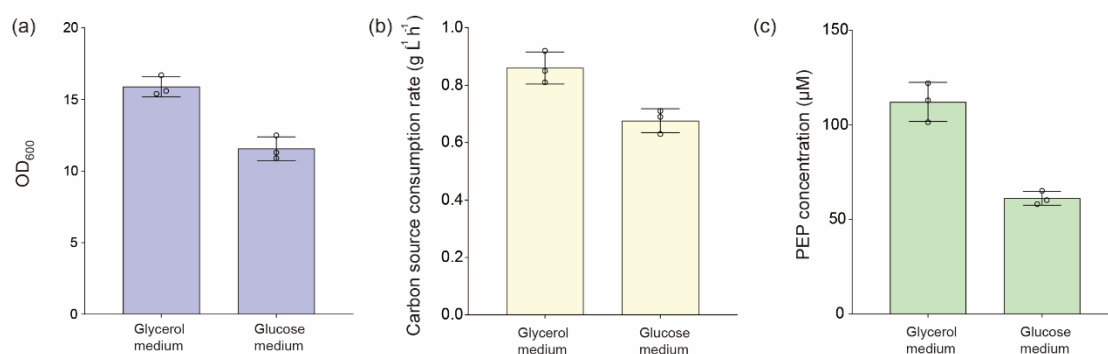
where y_i^{pred} is the predicted $k_{cat} K_m^{-1}$ value, y_i^{exp} is the experimental $k_{cat} K_m^{-1}$ value and n is the total number of items in the dataset (validation dataset or test dataset)."

Regarding the suggestion of lacking extensibility, we believe that encountering and resolving experimental challenges constitutes the core of scientific research. During our experiments, we observed suboptimal enzymatic activity in AGE and noted that the state-of-the-art DLkcat model failed to identify superior AGE variants. Consequently, we retrained DLkcat to develop DLCatalysis, a model specifically designed to predict k_{cat}/K_m values. Experimental validation confirmed that DLCatalysis successfully identified enhanced AGE variants and accurately predicted improved AGE mutants. In contrast, UniKP was limited to mine and mutational prediction of Tyrosine ammonia lyase, while EITLEM lacked experimental validation of model performance.

Q4

The data analysis throughout the article lacks depth (e.g., in lines 376-385, significant differences in NeuAc yield and byproducts were observed between the two carbon source feeding strategies. The authors suggest glycerol provides more PEP (phosphoenolpyruvate), but no specific analysis was conducted).

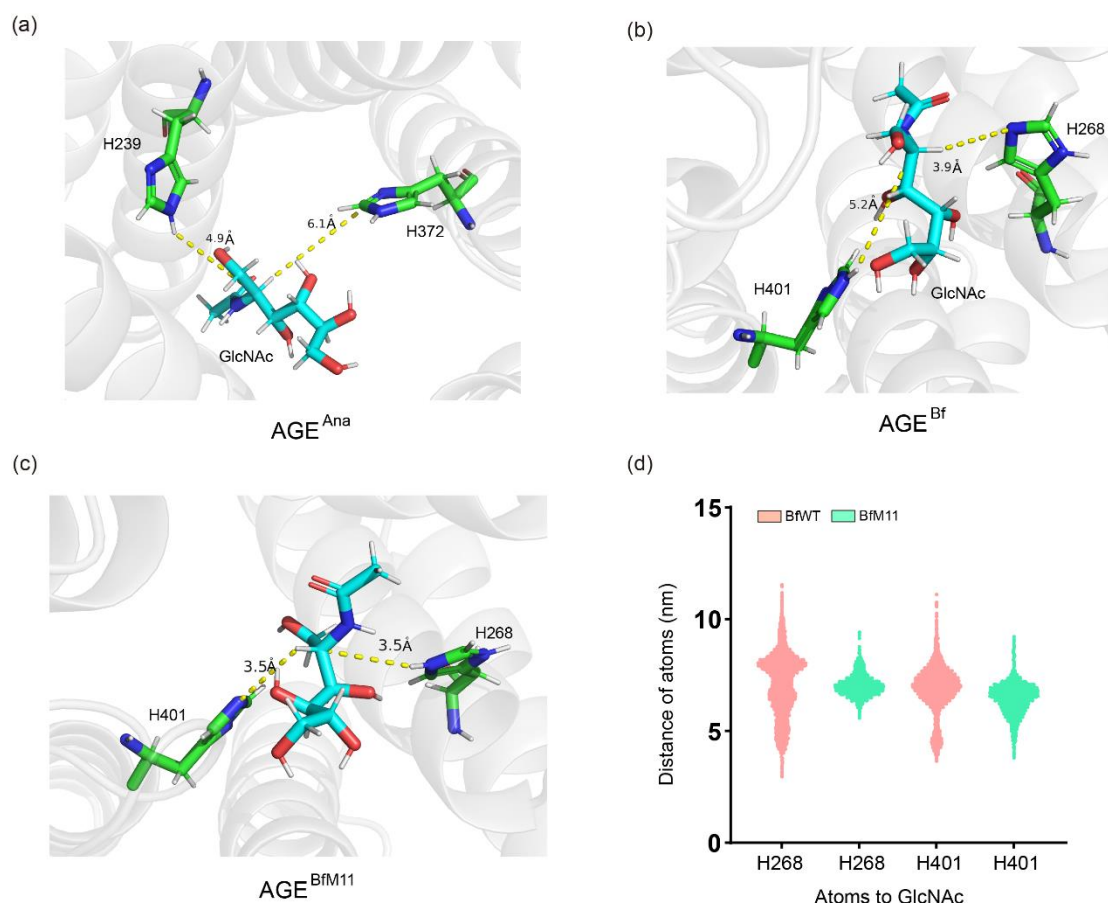
A4: Thank you for your valuable comments and kind suggestions. We have revised the whole manuscript to improve the depth of data analysis. For example, to explain why glycerol outperforms glucose as a carbon source in our NeuAc biosynthesis system, we added the following content in **Result** section: "To further investigate the effects of glycerol and glucose on strain growth and PEP content, EMSA20 was cultured in media containing either carbon source. After 24 h cultivation, EMSA20 exhibited OD₆₀₀ of 15.9, a carbon source consumption rate of 0.87 g L⁻¹ h⁻¹, and a PEP content of 112.2 μM in glycerol medium, whereas in glucose medium these parameters were 13.75, 0.74 g L⁻¹ h⁻¹, and 61.2 μM, respectively (Supplementary Fig. S7). These data demonstrate that glycerol cultivation enables faster carbon source utilization, supporting enhanced growth and higher PEP accumulation compared to glucose cultivation. This metabolic advantage might stem from ptsG (phosphotransferase system) knockout, which specifically impairs glucose utilization in EMSA20 while preserving glycerol metabolic efficiency." Please see Line 423-432 in the manuscript.



“Supplementary Fig. S7. Cultivation results of strain EMSA20 in glycerol- and glucose-based media. (a) The OD₆₀₀ of strain EMSA20 in different culture medium. (b) The carbon source consumption rate of strain EMSA20 in different culture medium. (c) The PEP concentration of strain EMSA20 in different culture medium.” Please see Line 91-94 in **Supporting Information** file.

We further analyzed the enzymatic activity differences among AnaAGE, BfWT, and BfM11 by comparing their catalytic distances. The analysis of AnaAGE and BfWT was added in the **Results** section: *“To understand the structural basis for the higher catalytic activity of AGE^{Bf} compared to AGE^{Ana}, molecular docking simulations were performed to analyze the binding modes of the GlcNAc in both enzymes. The isomerization of GlcNAc to ManNAc is known to proceed via an S_N1-type mechanism, involving stereoinversion at the C2 position¹⁰. This reaction is facilitated by two conserved histidine residues that act cooperatively: one serves as a base to deprotonate the C2 hydroxyl group, generating a planar oxocarbenium ion intermediate, while the other donates a proton to the opposite face of the intermediate, completing the epimerization. In AGE^{Ana}, these critical residues are H239 and H372; sequence and structural alignment identify the corresponding catalytic pair in AGE^{Bf} as H268 and H401. Analysis of the docked complexes revealed notable differences in the spatial arrangement of the catalytic His relative to the reactive C2 atom of GlcNAc. In AGE^{Ana} (Supplementary Fig. S3a), the distances between the imidazole rings of H239 and H372 and the C2 atom are 4.9 Å and 6.1 Å, respectively, slightly beyond the optimal range for efficient proton transfer. In contrast, in AGE^{Bf} (Supplementary Fig. S3b), the equivalent distances from H268 and H401 to C2 are reduced to 3.9 Å and 5.2 Å, indicating a closer and more geometrically favourable positioning of the substrate within the active site.”* Please see Line 287-302 in the manuscript.

For BfM11, the analysis was added in the **Results** section: *“Molecular docking of AGE^{BfM11} with GlcNAc revealed that the distance between the C2 atom of the substrate and the catalytic residues H268 and H401 is shorter than that in the AGE^{Bf} (Supplementary Fig. S3c). Furthermore, 100 ns molecular dynamics simulations showed that the catalytic distances in BfM11 are consistently lower than those observed in AGE^{Bf} throughout the simulation trajectory (Supplementary Fig. S3d). These structural and dynamic features contribute to enhanced proton transfer efficiency and provide a mechanistic basis for the higher catalytic activity of the AGE^{BfM11} variant.”* Please see Line 362-368 in the manuscript.



“Supplementary Fig. S3. Catalytic distance of AnaAGE, BfWT, and BfM11. (a) The active site of AGE^{Ana} with GlcNAc showing the catalytic distance. (b) The active site of AGE^{Bf} with GlcNAc showing the catalytic distance. (c) The active site of AGE^{BfM11} with GlcNAc showing the catalytic distance. (d) After 100 ns of molecular dynamics simulation, the distances from H268 and H401 to the C2 atom of GlcNAc were analyzed for AGE^{Bf} and AGE^{BfM11}, respectively.” Please see Line 73-77 in the **Supporting Information** file.

Reviewer #3 (Remarks to the Author):

The paper is well written and the aims, objectives and approach are clear. The Figures are well presented. The methodology is clear. The results are interesting and the productivity is impressive.

Q1

I have a few queries below. While the pipeline was successful in selecting improved enzyme variants, the authors still only selected a small number, 16 in total. How can the authors be sure that these were the best variants? Did the authors consider a HTP approach to generating more variants? This could have enabled them to expand the number of mutants and expand their pipeline. This could also have been used to improve the model further.

A1: Thank you for your kind suggestions. Given the vast mutational design space of proteins, we employed deep learning and computational design to significantly reduce the search space and enhance efficiency. During fermentation, accumulation of ManNAc indicated that AGE activity had already met metabolic demands, suggesting that further enzyme enhancement would not resolve productivity limitations due to the systemic nature of metabolic engineering. While high-throughput

screening remains a powerful approach, the absence of transcription factors specifically responsive to NeuAc precluded its application. Utilizing the NeuAc-responsive nanR transcription factor would introduce false-positive cells from NeuAc transport, rendering high-throughput mutant screening unfeasible^{6, 11}. Experimental validation confirmed the exceptional efficacy of our integrated deep learning and computational design strategy.

Q2

While the fermentation was successful it focused primarily on using glucose and glycerol as a feedstock-did the authors consider using any industrial feedstocks-such as molasses's.

A2: Thank you for your valuable suggestion. Glucose and glycerol serve as the carbon sources for *E. coli* fermentation, as evidenced by their predominant adoption in industrial-scale processes and recent high-impact studies on NeuAc biosynthesis^{6, 12, 13, 14, 15, 16, 17, 18}. Although molasses is a low-cost carbon source as a sugar industry byproduct, its primary composition of sucrose renders it incompatible with the non-sucrose-utilizing *E. coli* MG1655 strain employed in this study¹⁹. Consequently, molasses was excluded from carbon source selection. Future work will address this limitation by engineering sucrose metabolic pathways into the host strain to reduce production costs.

Q3

What is the market for N-acetylneuraminic acid? It would be useful to understand the market size.

A3: Thank you for your suggestion. Following your suggestions, we added the sentence to describe the market for N-acetylneuraminic in the **Introduction** section: "The global N-acetylneuraminic acid (NeuAc) market valued at USD 500 million in 2024 is projected to reach USD 1.0 billion by 2033 with a compound annual growth rate of 8.5% from 2026 to 2033²⁰." Please see Line 51-53 in the manuscript.

Q4

It would also have been useful to have a comparison between current production methods and the method detailed here-to highlight why this would be a better method for productivity.

A4: Thank you for your suggestion. This comparison has been incorporated and rewritten it into the **Discussion** section: "Recent advances in NeuAc biosynthesis have leveraged established metabolic engineering strategies to achieve 46.92 g L⁻¹ and 0.82 g L⁻¹h⁻¹ productivity in 5-L bioreactor fermentation¹⁴. However, by employing modular engineering to construct an initial NeuAc-producing strain, we utilized deep learning to identify a superior and previously unreported enzyme combination. Through enzyme engineering, we improved the catalytic efficiency of rate-limiting enzymes and discovered a mechanism to prevent intracellular accumulation of NeuAc, thereby increasing NeuAc productivity. Although the overall NeuAc productivity was 0.66 g L⁻¹h⁻¹, the titer and productivity reached 53.6 g L⁻¹ and 0.90 g L⁻¹h⁻¹ at 60 h fermentation, respectively. These results demonstrate that our approach outperforms current state-of-the-art methodologies." Please see Line 551-560 in the manuscript.

Q5

It would be useful to understand how the N-acetylneuraminic acid would be extracted from the supernatant.

A5: Thank you for your comment. NeuAc titers in this study were quantified directly from

fermentation broth without extraction. NeuAc was extracted using a previously described method.²¹ Briefly, the reaction solution was collected and centrifuged at 8000 g for 10 min to remove the cell pellet. The supernatant was adjusted to pH 4.5 with glacial acetic acid and then loaded onto an anion exchange column (500 mL) pre-equilibrated with 2M formic acid. After removing the unreacted GlcNAc and ManNAc using deionized water, NeuAc was eluted with a linear gradient of 0-1 M formic acid at a flow rate of 10 mL min⁻¹. The desired fraction was concentrated to approximately 400 mM and then crystallized by adding 5 volumes of glacial acetic acid. After standing for 2 days at 4 °C, the precipitate was recovered using vacuum filtration and washed with fresh glacial acetic acid before drying at 40 °C to constant weight.

Q6

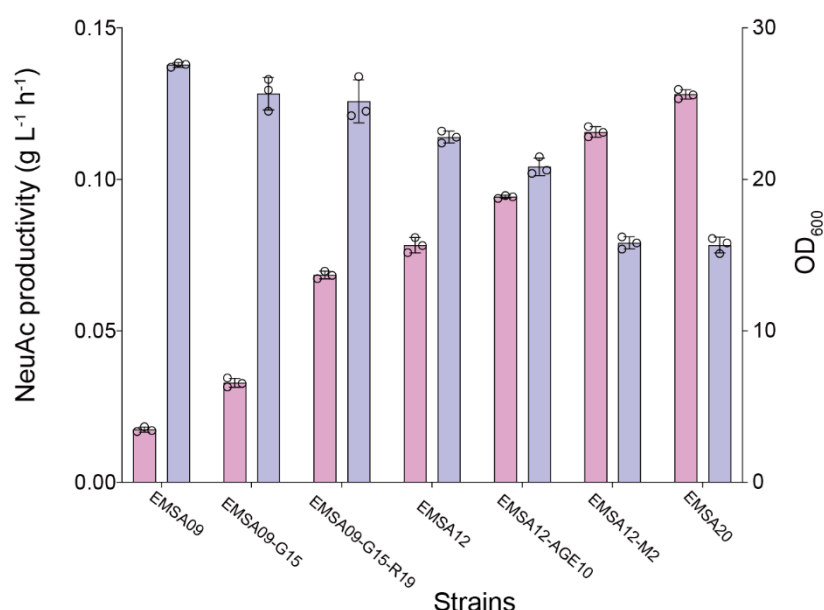
While I appreciate that this is a demonstration of the platform-considering scale it would be useful to understand the rationale for using an inducible promoter and fed batch fermentation over batch and a constitutive promoter.

A6: Thank you for your valuable suggestion. We have compared the different fermentation modes in the **Discussion** section: *“Due to the fact that NeuAc biosynthesis consumes critical growth-essential intermediates such as acetyl-CoA, glutamine, and phosphoenolpyruvate (PEP), we determined that activating the NeuAc pathway during the growth phase would be detrimental to cellular metabolism. Therefore, we selected the widely adopted IPTG-inducible expression system to temporally separate growth and production phases. This choice is further substantiated by prior attempts using constitutive expression systems for NeuAc production in Bacillus subtilis¹³ and E. coli¹⁶, which consistently yielded lower titers compared to inducible systems¹⁴. While our current study employs this inducible strategy, future efforts will focus on developing auto-inducible systems featuring low basal expression, broad induction thresholds, and high sensitivity for optimized NeuAc biosynthesis. In recent years, more E. coli-based industrial processes have adopted fed-batch technology²², primarily due to its ability to achieve significantly higher product titers. While batch fermentation theoretically shortens the production cycle time, the requirement for high initial carbon source concentrations inevitably leads to the accumulation of growth-inhibitory byproducts, such as acetate and lactate, during fermentation process²³. To maximize yield, we therefore implemented a fed-batch strategy for NeuAc biosynthesis.”* Please see Line 535-549 in the manuscript.

Q7

It would have been useful to compare the productivity of each improved strain on an OD basis-graphical Figure.

A7: Thank you for your suggestion. We added the updates in **Discussion** section: *“Furthermore, the optimal strains at each stage exhibited a significant increase in productivity, while the OD₆₀₀ showed a declining trend (Supplementary Fig. S7), indicating that the metabolic flux, originally biased towards growth, was redirected towards production through the modifications.”* Please see Line 530-533 in the manuscript.



[“Supplementary Fig. S7. The comparison of the productivity of each improved strain.”](#)Please see Line 98 in **Supporting Information** file.

Q8

Fig. 6a the legend states glycerol but the Fig. states glucose.

A8: Thank you very much for your careful comments. We have revised the Fig. 6a., please see Line 1095 in the manuscript.

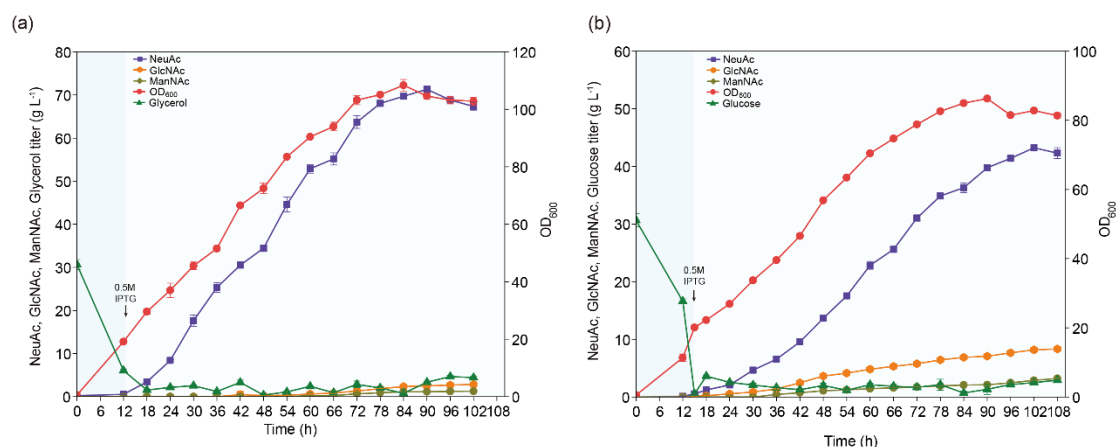


Fig. 6. Fed-batch fermentation of NeuAc biosynthesis in a 5-L fermenter. (a) The time curve of NeuAc production in a 5-L fermenter using glycerol as the carbon source. (b) The time curve of NeuAc production in a 5-L fermenter using glucose as the carbon source.

Reference

- Li F, *et al.* Deep learning-based kcat prediction enables improved enzyme-constrained model reconstruction. *Nature Catalysis* **5**, 662-672 (2022).
- Yu H, Deng H, He J, Keasling JD, Luo X. UniKP: a unified framework for the prediction of enzyme kinetic parameters. *Nat Commun* **14**, 8211 (2023).
- Shen X, Cui Z, Long J, Zhang S, Chen B, Tan T. EITLEM-Kinetics: A deep-learning framework for kinetic parameter prediction of mutant enzymes. *Chem Catalysis* **4**, (2024).
- Liu H, *et al.* MECE: a method for enhancing the catalytic efficiency of glycoside hydrolase

- based on deep neural networks and molecular evolution. *Science Bulletin* **68**, 2793-2805 (2023).
5. Zhou J, Huang M. Navigating the landscape of enzyme design: from molecular simulations to machine learning. *Chem Soc Rev* **53**, 8202-8239 (2024).
 6. Pang Q, *et al.* In vivo evolutionary engineering of riboswitch with high-threshold for N-acetylneuraminic acid production. *Metabolic Engineering* **59**, 36-43 (2020).
 7. Gao S, *et al.* Empowering biomedical discovery with AI agents. *Cell* **187**, 6125-6151 (2024).
 8. Huang J, *et al.* Discovery of deaminase functions by structure-based protein clustering. *Cell* **186**, 3182-3195 e3114 (2023).
 9. Hou K, *et al.* De novo design of porphyrin-containing proteins as efficient and stereoselective catalysts. *Science* **388**, 665-670 (2025).
 10. Samuel J, Tanner ME. Mechanistic aspects of enzymatic carbohydrate epimerization. *Nat Prod Rep* **19**, 261-277 (2002).
 11. Peters G, *et al.* Development of N-acetylneuraminic acid responsive biosensors based on the transcriptional regulator NanR. *Biotechnol Bioeng* **115**, 1855-1865 (2018).
 12. Peng Y, Ma L, Xu P, Tao F. High-Performance Production of N-Acetyl-d-Neuraminic Acid with Whole Cells of Fast-Growing *Vibrio natriegens* via a Thermal Strategy. *Journal of Agricultural and Food Chemistry* **71**, 20198-20209 (2023).
 13. Zhang X, *et al.* Engineering of Synthetic Multiplexed Pathways for High-Level N-Acetylneuraminic Acid Bioproduction. *Journal of Agricultural and Food Chemistry* **69**, 14868-14877 (2021).
 14. Zhao M, Zhu Y, Wang H, Zhang J, Xu W, Mu W. Efficient Production of N-Acetylneuraminic Acid in *Escherichia coli* Based on the UDP-N-Acetylglucosamine Biosynthetic Pathway. *Journal of Agricultural and Food Chemistry* **71**, 10701-10709 (2023).
 15. Pang Q, Han H, Xu Y, Liu X, Qi Q, Wang Q. Exploring Amino Sugar and Phosphoenolpyruvate Metabolism to Improve *Escherichia coli* N-Acetylneuraminic Acid Production. *Journal of Agricultural and Food Chemistry* **68**, 11758-11764 (2020).
 16. Liu C, Lv X, Li J, Liu L, Du G, Liu Y. Metabolic Engineering of *Escherichia coli* for Increased Bioproduction of N-Acetylneuraminic Acid. *Journal of Agricultural and Food Chemistry* **70**, 15859-15868 (2022).
 17. Tian R, *et al.* Titrating bacterial growth and chemical biosynthesis for efficient N-acetylglucosamine and N-acetylneuraminic acid bioproduction. *Nat Commun* **11**, 5078 (2020).
 18. Kang J, *et al.* Engineering of an N-acetylneuraminic acid synthetic pathway in *Escherichia coli*. *Metabolic Engineering* **14**, 623-629 (2012).
 19. Mohamed ET, *et al.* Generation of an *E. coli* platform strain for improved sucrose utilization using adaptive laboratory evolution. *Microbial Cell Factories* **18**, 116 (2019).
 20. Reports VM. Global N-Acetylneuraminic Acid Market Size By End-User (Pharmaceutical Industry, Food & Beverage Industry), By Application (Healthcare Applications, Dietary Supplements), By Source (Animal-Based Sources, Plant-Based Sources), By Form (Powdered Form, Liquid Form), By Bus Design (Single Deck Buses, Double Deck Buses), By Geographic Scope And Forecast.) (2025).
 21. Chen X, *et al.* Development of an *Escherichia coli*-based biocatalytic system for the efficient synthesis of N-acetyl-D-neuraminic acid. *Metabolic Engineering* **47**, 374-382 (2018).
 22. Kopp J, Slouka C, Spadiut O, Herwig C. The Rocky Road From Fed-Batch to Continuous Processing With *E. coli*. *Front Bioeng Biotechnol* **7**, 328 (2019).

23. Luli GW, Strohl WR. Comparison of growth, acetate production, and acetate inhibition of *Escherichia coli* strains in batch and fed-batch fermentations. *Applied and Environmental Microbiology* **56**, 1004-1011 (1990).

Reviewer comments:

Reviewer #1

(Remarks to the Author):

Unfortunately, most of the authors' responses do not sufficiently address our major previous concerns. Below, we outline the remaining issues for each response in detail. While the method developed by the authors has clearly resulted in a substantial improvement in the yield of the engineered E. coli strain and represents an important contribution, it remains essential that the limitations and potential weaknesses of both the employed and newly developed approaches are discussed transparently. Several of the current claims are either incorrect, insufficiently supported, or require further analysis and clarification. In its current form, the manuscript does not yet meet the scientific rigor expected for publication in a high-impact journal such as Nature Communications.

Response to A1:

While it may have been true that at the time of development, DLKcat was among the few available tools, it is now well established that DLKcat performs very poorly for predicting mutation effects and is no better than a naïve baseline for wildtype predictions. Even if these limitations were not evident at the time of development, they are now clear and should be explicitly acknowledged and discussed in the manuscript. However, the authors still do not adequately address these limitations. This is essential for understanding the weaknesses of the current approach and for identifying directions for future improvement.

Response to A2:

Splitting folds at random creates massive overlap between training and test sequences. For example, in the split EITLEM folds, 77% of the sequences in the kcat test set also appear in the kcat training set. More importantly, for 25% of the test set, entries share both the same sequence and identical kcat values with entries in the training set; they differ only in the substrate, representing the same reaction with two substrates. Such data leakage will occur for every simply randomly split enzyme constant dataset. Careful splitting is thus crucial for a fair evaluation. These limitations need to be discussed.

Response to A4:

The large discrepancies in model performance between DLcatalysis, UniKP, and EITLEM appear implausible. The reported R2 value for DLcatalysis is nearly three times higher than that of EITLEM. However, the EITLEM architecture is more advanced and has previously shown substantially better performance than the DLKcat architecture on a closely related task when trained and evaluated on exactly the same data (see <https://doi.org/10.1016/j.checat.2024.101094>, Figure 7A). Moreover, EITLEM leverages transfer learning from kcat and KM to kcat/KM prediction, which provides an additional advantage over models that do not employ such techniques. In the original publication, EITLEM achieved a test R2 above 0.8, which is more than four times higher than the value reported here. The authors should provide a clear explanation for this substantial discrepancy, as random differences in data splits alone are unlikely to account for such variation. The same arguments apply to the unexpectedly low R2 reported for UniKP.

It is unrealistic and scientifically inappropriate to claim the development of a new state-of-the-art kcat/KM predictor without further justification and analysis, especially given the use of less sophisticated model architectures and training strategies compared to recent models.

Finally, it is important that comparing models trained on different datasets using the same test set is problematic, as each model may exhibit different forms of data leakage. A possible solution would be to evaluate each model's performance across test subsets stratified by sequence similarity to their respective training sets, as shown in <https://doi.org/10.1016/j.checat.2024.101094>, Figure 7A.

Response to Q5:

The authors provide no justification for their belief that 162 experiments would have been necessary to obtain 12 positive variants without deep learning.

Response to Q6:

If the authors chose not to use DLKcat at this stage due to its limited capabilities, it is unclear why it was still employed elsewhere in the pipeline. Additionally, we don't understand what is meant by "eliminating interference". We also still do not understand why the authors did not apply their own

kcat/KM prediction models at this stage of the workflow. The criteria for deciding when to integrate these prediction models into enzyme screening and when to omit them should be clearly explained and justified.

Reviewer #2 (Remarks to the Author)

The authors have submitted a revised manuscript that thoughtfully addresses the majority of the concerns raised during the initial review. The significant enhancements, particularly the detailed description of the DLCatalysis model architecture, hyperparameter optimization, and its comparative benchmarking against state-of-the-art tools like UniKP and EITLEM, have substantially improved the manuscript's clarity and scientific rigor. The additional experimental data and deeper mechanistic analysis, such as the molecular docking and dynamics simulations explaining the enhanced activity of AGE variants, further strengthen the claims.

My primary concern regarding the novelty of identifying AGE as the bottleneck has been adequately countered by the authors, who correctly distinguish their systematic, quantitative metabolite flux analysis from prior indirect evidence. The strategic integration of existing deep learning and computational tools into a novel, efficient workflow (CD-IGM) for enzyme engineering is now more clearly framed as a valid and impactful contribution, moving the field forward through intelligent synthesis rather than solely through the creation of entirely new algorithms.

Consequently, I am pleased to reconsider my initial assessment. The work presents a comprehensive and highly successful metabolic engineering campaign, achieving a record-breaking titer of NeuAc. The integration of deep learning, while not constituting a fundamental algorithmic breakthrough, has been demonstrated to be functionally critical in drastically reducing the experimental burden and guiding effective decisions. The manuscript, in its current form, makes a valuable contribution and is now much stronger and suitable for publication in a journal of your caliber.

To further elevate the impact and general significance of this work, I have one major suggestion for the authors to consider as a clear direction for future research:

The study powerfully validates the proposed deep learning-driven pipeline for a single, critical enzyme (AGE) within one specific biosynthetic pathway. The true potential and robustness of such a computational-metabolic engineering framework would be profoundly amplified by demonstrating its extensibility. A compelling way to achieve this would be to apply the same integrated pipeline (e.g., the CLEAN-DLCatalysis screening followed by CD-IGM optimization) to a different rate-limiting enzyme in a distinct product synthesis pathway in a future study. For instance, applying this methodology to enhance a key reductase in astaxanthin biosynthesis or a cytochrome P450 in plant natural product synthesis would provide critical evidence that the framework is not a one-off solution but a generalizable platform technology. Such a demonstration would unequivocally establish its broad applicability in microbial cell factory development.

I recommend that the authors include a brief discussion on this perspective of future work and extensibility in the manuscript's conclusion or discussion section. This will help readers immediately grasp the wider implications and future trajectory of this research.

Therefore, I recommend acceptance of the manuscript pending minor revisions to incorporate the aforementioned point on future extensibility validation.

Specific minor point:

The comparison of model performance (R^2 and r.m.s.e. values) between DLCatalysis, UniKP, and EITLEM in Supplementary Fig. S2c-d is a crucial addition. Please ensure the figure caption explicitly states the size and nature of the test set used for this comparative benchmarking to provide full context.

Reviewer #3 (Remarks to the Author)

I would like to thank the authors for addressing my comments, I am satisfied with the responses and recommend publication.

For your guidance, itemized answers to each reviewer's comments are appended below (answers are in blue font).

Response to reviewers' comments point by point:

Response to Reviewer #1

We are very grateful for the time and care you have devoted to our manuscript across two rounds of review, and we owe you a sincere apology for the confusion caused by the previous version, in particular by the way our deep-learning component was presented and applied. Your critique made clear that the continued reliance on DLkcat at any stage of the workflow was not defensible given its known limitations on wild-type and mutation effects, and that the previous benchmark, based on randomly split folds, was contaminated by extensive train-test sequence overlap and therefore could not support the comparative claims we drew from it. Taking these points to heart, we decided in this revision to start over rather than patch the previous predictor: we rebuilt the kinetic model from scratch and removed DLkcat entirely from the pipeline; we re-evaluated the new model against the published predictors on identity-clustered splits, so that the comparison is no longer biased by sequence overlap; and we carried out a fresh round of wet-lab validation across the key enzymes of the NeuAc pathway, so that the predictor is judged not by benchmark numbers alone but by its contribution to actual strain engineering. The redesign and the new experimental work are the main reasons why this revision has taken longer than we had hoped, and we are grateful for your patience. Our detailed point-by-point responses to each of your comments are provided below.

Q1

Comment:

While it may have been true that at the time of development, DLKcat was among the few available tools, it is now well established that DLKcat performs very poorly for predicting mutation effects and is no better than a naïve baseline for wildtype predictions. Even if these limitations were not evident at the time of development, they are now clear and should be explicitly acknowledged and discussed in the manuscript. However, the authors still do not adequately address these limitations. This is essential for understanding the weaknesses of the current approach and for identifying directions for future improvement.

Response:

We thank the reviewer for this important comment. We agree that DLKcat has clear limitations, particularly for mutation-effect prediction and for catalytic-efficiency prediction beyond similarity-driven wild-type cases. After recognizing these limitations, we substantially revised the computational part of the manuscript and developed a new model, DLCatalysis, instead of continuing to rely on DLKcat as the main predictive framework.

DLCatalysis was designed as a direct kcat/Km prediction model for enzyme–substrate pairs. Briefly, the model uses residue-level enzyme embeddings, multimodal substrate descriptors, hierarchical EC-number embeddings, bidirectional enzyme–substrate cross-attention, and a three-branch prediction structure that separates enzyme baseline, substrate contribution, and enzyme–substrate interaction effects. In addition, we introduced pairwise ranking losses to improve the model’s ability to prioritize enzymes and mutants, which is the key requirement in our experimental workflow.

We also benchmarked DLCatalysis against representative kinetic-parameter prediction models, including DLkcat, UniKP, EITLEM-Kinetics, and CataPro, using the same curated dataset and the same **CD-HIT 40% identity-clustered ten-fold cross-validation splits**. Under this unified benchmark, DLCatalysis showed the best performance among the compared models, with a 10-fold CV PCC of 0.419, compared with UniKP (0.391), EITLEM-Kinetics (0.363), CataPro (0.346), and DLkcat (0.194). Importantly, unlike EITLEM-Kinetics and CataPro, which rely on staged strategies involving separate kcat and Km modelling and subsequent transfer-learning or correction steps for kcat/Km prediction, DLCatalysis predicts kcat/Km directly in a single end-to-end framework. This design simplifies the workflow, avoids additional intermediate modelling steps, and reduces computational cost while maintaining better performance in our benchmark.

To clarify this revision, we have added the following description in the revised manuscript. Please see Lines 229-288 in the revised manuscript.

“Discovering the rate-limiting enzyme by deep learning pipeline

The DLkcat-based screening described above identified an optimal enzyme combination for NeuAc biosynthesis but failed to resolve the persistent GlcNAc accumulation. Although GotEnzymes ranked AGE candidates, none of the selected enzymes outperformed the reference AGEAna in converting GlcNAc to ManNAc, indicating that kcat alone is insufficient for identifying enzymes with superior catalytic performance under physiological conditions. In our engineered strain, intracellular GlcNAc remains well below enzyme saturation, a regime where catalytic flux depends on kcat Km-1 rather than kcat. This outcome led us to systematically evaluate other models. UniKP predicts kcat Km-1 using ProtTrans embeddings with an extra

trees regressor³⁴, but its k_{cat} K_m -1 evaluation relied on a limited test set without sequence-clustered splits. EITLEM-Kinetics applies iterative transfer learning with ESM-1v embeddings but did not cluster sequences before data partitioning, allowing homologous proteins to leak between training and test sets³⁵. CataPro addressed leakage through CD-HIT clustering at 40% identity³⁶ but predicts k_{cat} K_m -1 indirectly, training separate k_{cat} and K_m models whose outputs are divided and refined by a correction network.

The catalytic efficiency ($k_{cat} K_m^{-1}$) of an enzyme–substrate pair reflects three sources of variation: an enzyme-level component (some enzymes are intrinsically efficient catalysts across many substrates), a substrate-level component (some chemotypes are broadly more amenable to enzymatic turnover), and an enzyme–substrate-specific interaction component (how well a given active site accommodates a given substrate). In representative prior $k_{cat} K_m^{-1}$ predictors³⁴⁻³⁶, these three sources are absorbed into a single regression head over a concatenated enzyme–substrate embedding, and the relative contribution of marginal effects versus pair-specific interaction cannot be directly inspected. Motivated by additive two-way decompositions of paired observational data³⁶, DLCatalysis instead provides an explicit additive inductive bias for this structure (Fig. 3a): residue-level ProtT5³⁷ embeddings of the enzyme, a substrate representation pooled from Morgan fingerprints, MolT5 language embeddings³⁸, and GROVER graph features³⁹, and a hierarchical EC-number encoding are routed to three prediction heads, an enzyme-baseline branch, a substrate-baseline branch, and an enzyme–substrate interaction branch, whose contributions are combined through a learned per-sample gate. This decomposition is intended to encourage the model to attribute its predictions to enzyme-level, substrate-level, and pair-specific sources of variation rather than learning a fully entangled mapping; the decomposition is structural and is not enforced by an explicit training constraint. Beyond conventional absolute-value regression, DLCatalysis is trained with two complementary pairwise ranking objectives that reflect how kinetic predictors are used in practice. An enzyme-wise ranking loss compares predictions for alternative substrates of the same enzyme; a substrate-wise ranking loss compares predictions for alternative enzymes of the same substrate. Together with the regression term, these objectives are intended to encourage the model to learn not only the absolute magnitude of $k_{cat} K_m^{-1}$ but also its relative ordering within enzyme families and within substrate classes, which are the two ordering relations on which downstream tasks such as substrate screening, enzyme mining, and saturation-mutagenesis ranking ultimately depend. To benchmark DLCatalysis, we retrained UniKP, EITLEM, CataPro, and DLkcat on our curated dataset using the same ten-fold cross-validation splits. DLCatalysis achieved PCC of 0.419 after hyperparameter optimization (Supplementary Fig.S1a-j). Component ablation (Supplementary Fig.S2) supports keeping all six components in the final architecture. Removing any of the three molecular representations lowers the 10-fold CV PCC, with Morgan substructure fingerprints accounting for the largest marginal contribution (Δ PCC 0.035), followed by MolT5 (0.022) and GROVER (0.010); these three representations cover complementary chemical scales of

substructure patterns, learned sequence-level semantics, and learned graph-level features. The three prediction heads show a parallel pattern. The interaction branch y_{int} contributes most (ΔPCC 0.015), with the enzyme branch y_{seq} (0.012) and the substrate branch y_{sub} (0.009) following, consistent with cross-attention capturing residue-to-atom coupling that the enzyme or substrate branch alone does not encode. The ranking is consistent across the 10 folds, indicating these differences reflect systematic component contributions rather than fold-specific variation. On the unified benchmark (Supplementary Fig. S3a-b), DLcatalysis reached a 10-fold CV PCC of 0.419, ahead of UniKP (0.391), EITLEM (0.363), CataPro (0.346), and DLkcat (0.194). The advantage is consistent with modelling $k_{\text{cat}} K_m^{-1}$ as a single target, rather than the staged pipelines used by representative prior models such as CataPro and EITLEM that first predict k_{cat} and K_m independently and then combine them.”

For the method, we revised the in Line 593-654 of manuscript.

“Data preparation and deep learning model construction

The k_{cat}/K_m dataset compiled by CataPro³⁶, which collected experimentally measured k_{cat} and K_m values from BRENDA⁵² and SABIO-RK⁵³, was used as the starting point. Enzyme sequences were matched via UniProt identifiers and canonical substrate SMILES were retrieved from PubChem. Several additional curation steps were applied. Entries whose substrates were catalytic metal ions (e.g., Zn^{2+} , Mn^{2+}) were removed, as these cofactors do not undergo chemical transformation and lack meaningful $k_{\text{cat}} K_m^{-1}$ values. Sequences exceeding 1,000 residues were truncated, non-standard residues (U, Z, O, B) were replaced with X. Anomalous k_{cat}/K_m values caused by inconsistent K_m unit annotations in the source databases were identified and corrected. After duplicate removal, the dataset was supplemented with experimentally determined $k_{\text{cat}} K_m^{-1}$ values for four AGE enzymes reported in the literature but absent from BRENDA and SABIO-RK. The final dataset comprised 24,810 enzyme-substrate pairs spanning 11,845 unique sequences and 3,117 substrates. All sequences were clustered at 40% identity using CD-HIT v4.8.1, and clusters were assigned to ten cross-validation folds such that no test enzyme shared >40% identity with any training enzyme.

Enzyme and substrate encoding

Enzymes were encoded by per-residue embeddings from ProtT5-XL-UniRef50. Substrates were described by three orthogonal embeddings: Morgan circular fingerprints from RDKit, MolT5-base (smiles2caption) sentence embeddings, and GROVER fingerprints from a pretrained chemical graph encoder. The four-level Enzyme Commission number was encoded by four trainable embedding tables (one per level), concatenated and projected into a reaction-context vector.

DL Catalysis architecture

DL Catalysis predicts $k_{cat} K_m^{-1}$ using a three-branch residual formulation:

$$\hat{y} = \mu_{\text{base}} + g(\mu_{\text{sub}} + \mu_{\text{int}}) \quad (2)$$

where μ_{base} is the enzyme baseline, μ_{sub} the substrate-dependent term, μ_{int} the enzyme–substrate interaction term, and $g \in [0,1]$ a sample-wise dynamic gate. The interaction branch employs a two-layer bidirectional cross-attention module between enzyme residue tokens and substrate feature tokens, in which substrate tokens first undergo self-attention and then the two streams attend to each other; pooled enzyme and substrate vectors are formed by learned attention pooling and masked averaging respectively, and concatenated together with the EC embedding to form the interaction representation. μ_{base} , μ_{sub} , and μ_{int} are produced by independent two-layer MLP heads, and the dynamic gate is computed from the pooled enzyme feature, pooled substrate feature, and EC embedding through a two-layer network with sigmoid output.

Training objective and protocol

DL Catalysis was trained with a composite objective that combines a Log-Cosh regression term, a batch-level Pearson-correlation regulariser, and two pairwise margin-ranking terms:

$$\mathcal{L} = \mathcal{L}_{\text{Log-Cosh}} + \lambda_{\text{PCC}} \mathcal{L}_{\text{PCC}} + \lambda_{\text{same-enzyme}} \mathcal{L}_{\text{same-enzyme}} + \lambda_{\text{same-substrate}} \mathcal{L}_{\text{same-substrate}} \quad (3)$$

The same-enzyme ranking term $\mathcal{L}_{\text{same-enzyme}}$ was applied to within-batch pairs sharing the same enzyme but different substrates, whereas the same-substrate ranking term $\mathcal{L}_{\text{same-substrate}}$ was applied to pairs sharing the same substrate but different enzymes.

Training was implemented in PyTorch Lightning with AdamW and 16-bit mixed precision. Hyperparameters were selected by Bayesian hyperparameter optimisation using Optuna.

Evaluation metrics

Three evaluation metrics were used: Pearson correlation coefficient (PCC) and root-mean-square error (RMSE):

$$\text{PCC} = \frac{\sum_{i=1}^n (y_i - \bar{y})(\hat{y}_i - \bar{\hat{y}})}{\sqrt{\sum_{i=1}^n (y_i - \bar{y})^2} \sqrt{\sum_{i=1}^n (\hat{y}_i - \bar{\hat{y}})^2}} \quad (4)$$

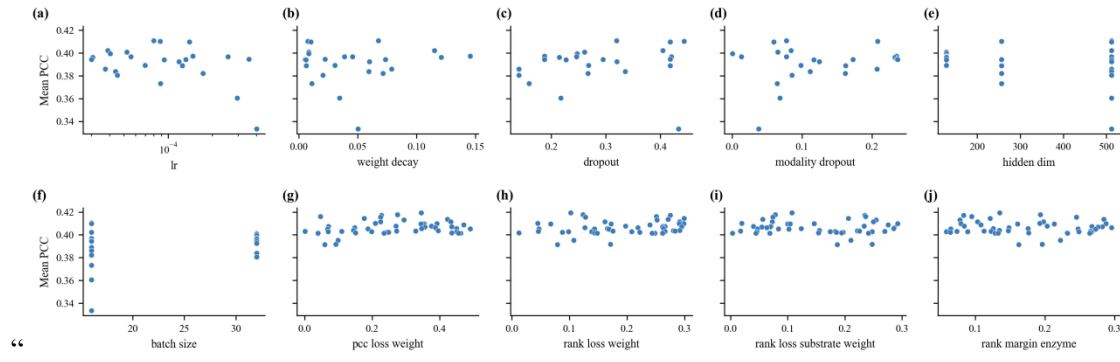
$$\sigma_{\text{rank}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (\bar{r}_{\text{pred}} - \bar{r}_{\text{exp}})^2} \quad (5)$$

Here, $\bar{r}_{\text{pred}}$ and $\bar{r}_{\text{exp}}$ are the predicted and experimental $\sigma_{\text{rank}}_{10}(\bar{r}_{\text{pred}}/\bar{r}_{\text{exp}})$, respectively. $\bar{r}_{\text{pred}}$ and $\bar{r}_{\text{exp}}$ represent the mean predicted and mean experimental values. r_{pred} and r_{exp} denote the corresponding mid-rank-tied ranks of $\bar{r}_{\text{pred}}$ and $\bar{r}_{\text{exp}}$.

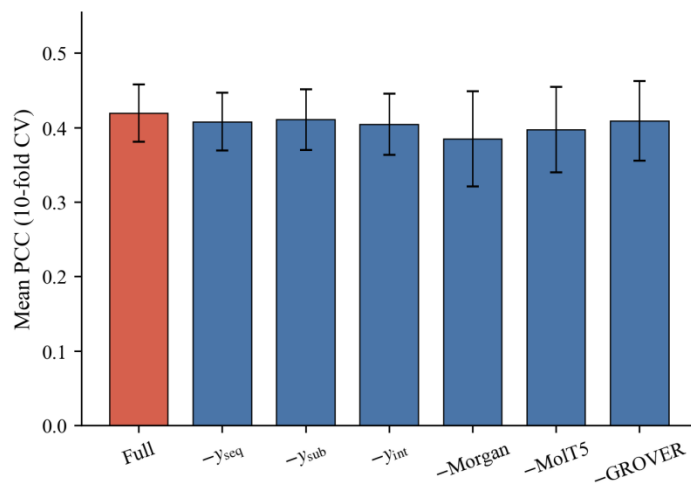
Benchmark

DLcatalysis was compared against four representative deep-learning predictors for enzyme kinetic-parameter prediction: DLkcat, UniKP, EITLEM-Kinetics, and CataPro. All baselines were re-trained on the curated dataset using the same CD-HIT 40 % identity-clustered ten-fold splits, and performance was reported as fold-wise mean PCC and RMSE.

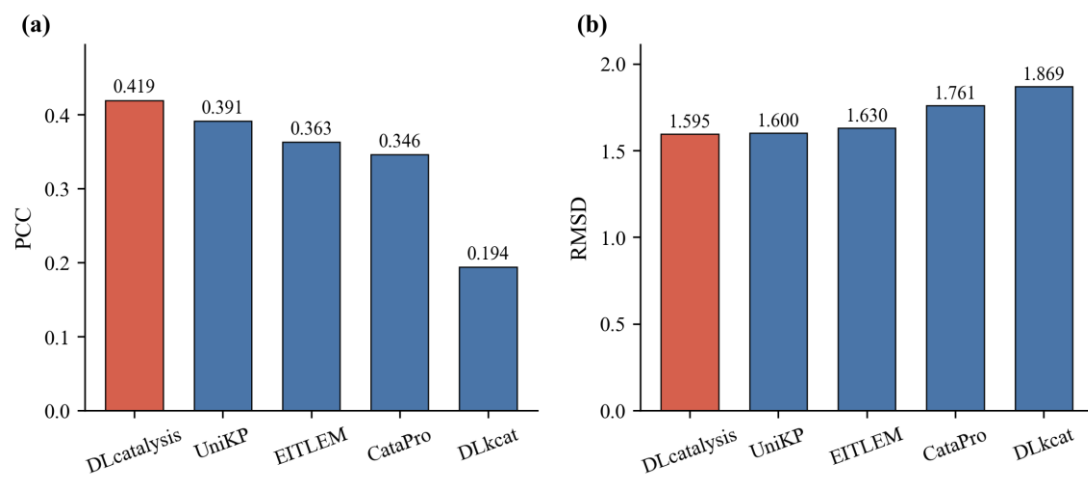
Please the figures in the supplement files.



Supplementary Fig. S1. Scatter plots showing the relationship between individual hyperparameters and mean Pearson correlation coefficient (PCC) during hyperparameter optimization. Each point represents one tested configuration. The evaluated hyperparameters include (a) learning rate, (b) weight decay, (c) dropout rate, (d) modality dropout rate, (e) hidden dimension, (f) batch size, (g) PCC loss weight, (h) ranking loss weight, (i) substrate-wise ranking loss weight, and (j) enzyme-wise ranking loss weight.



Supplementary Fig. S2. Component ablation of DLcatalysis. (a) Mean PCC from ten-fold cross-validation for the full DLcatalysis model and ablated variants. Error bars indicate variability across folds.



Supplementary Fig. S3. Comparison of DLcatalysis with existing k_{cat} K_m^{-1} predictors. (a) PCC and (b) RMSD of DLcatalysis, UniKP, EITLEM, CataPro and DLkcat evaluated under the same ten-fold cross-validation protocol.”

Q2

Comment:

Splitting folds at random creates massive overlap between training and test sequences. For example, in the split EITLEM folds, 77% of the sequences in the kcat test set also appear in the

k_{cat} training set. More importantly, for 25% of the test set, entries share both the same sequence and identical k_{cat} values with entries in the training set; they differ only in the substrate, representing the same reaction with two substrates. Such data leakage will occur for every simply randomly split enzyme constant dataset. Careful splitting is thus crucial for a fair evaluation. These limitations need to be discussed.

Response:

We fully agree with the reviewer. The previous random split was insufficient and could lead to sequence-level leakage between training and test sets. To address this concern, we have completely revised the data-splitting strategy. In the revised manuscript, all enzyme sequences were clustered using CD-HIT at 40% identity, and sequence clusters rather than individual records were assigned to ten cross-validation folds. Therefore, no enzyme in the test fold shares more than 40% sequence identity with any enzyme in the corresponding training fold.

To make this correction explicit, we have added the following description in the Methods section (Line 593-607):

“Data preparation and deep learning model construction

The k_{cat}/K_m dataset compiled by CataPro³⁶, which collected experimentally measured k_{cat} and K_m values from BRENDA⁵² and SABIO-RK⁵³, was used as the starting point. Enzyme sequences were matched via UniProt identifiers and canonical substrate SMILES were retrieved from PubChem. Several additional curation steps were applied. Entries whose substrates were catalytic metal ions (e.g., Zn^{2+} , Mn^{2+}) were removed, as these cofactors do not undergo chemical transformation and lack meaningful $k_{cat} K_m^{-1}$ values. Sequences exceeding 1,000 residues were truncated, non-standard residues (U, Z, O, B) were replaced with X. Anomalous k_{cat}/K_m values caused by inconsistent K_m unit annotations in the source databases were identified and corrected. After duplicate removal, the dataset was supplemented with experimentally determined $k_{cat} K_m^{-1}$ values for four AGE enzymes reported in the literature but absent from BRENDA and SABIO-RK. The final dataset comprised 24,810 enzyme-substrate pairs spanning 11,845 unique sequences and 3,117 substrates. All sequences were clustered at 40% identity using CD-HIT v4.8.1, and clusters were assigned to ten cross-validation folds such that no test enzyme shared >40% identity with any training enzyme.”

Q3

Comment:

The large discrepancies in model performance between D_Lcatalysis, UniKP, and EITLEM appear implausible. The reported R² value for D_Lcatalysis is nearly three times higher than that of EITLEM. However, the EITLEM architecture is more advanced and has previously shown

substantially better performance than the DLKcat architecture on a closely related task when trained and evaluated on exactly the same data. Moreover, EITLEM leverages transfer learning from kcat and KM to kcat/KM prediction, which provides an additional advantage over models that do not employ such techniques. In the original publication, EITLEM achieved a test R^2 above 0.8, which is more than four times higher than the value reported here. The authors should provide a clear explanation for this substantial discrepancy, as random differences in data splits alone are unlikely to account for such variation. The same arguments apply to the unexpectedly low R^2 reported for UniKP. It is unrealistic and scientifically inappropriate to claim the development of a new state-of-the-art kcat/KM predictor without further justification and analysis, especially given the use of less sophisticated model architectures and training strategies compared to recent models. Finally, it is important that comparing models trained on different datasets using the same test set is problematic, as each model may exhibit different forms of data leakage. A possible solution would be to evaluate each model's performance across test subsets stratified by sequence similarity to their respective training sets.

Response:

We thank the reviewer for this detailed and constructive criticism. We agree that random splitting can cause substantial data leakage in enzyme kinetic prediction, especially when homologous or highly similar enzyme sequences appear in both training and test sets. Following the reviewer's suggestion, we therefore rebuilt the benchmark using a stricter and fairer evaluation strategy.

In the revised analysis, all enzyme sequences were clustered using CD-HIT at 40% identity, and sequence clusters rather than individual records were assigned to ten cross-validation folds. This ensured that no test enzyme shared more than 40% sequence identity with any training enzyme. We then retrained DLkcat, UniKP, EITLEM-Kinetics, CataPro, and DLCatalysis on the same curated dataset and evaluated all models using exactly the same CD-HIT 40% identity-clustered ten-fold splits. This avoids comparing models trained on different datasets or evaluated under different leakage conditions.

Because this clustered-split benchmark is much stricter than random-split evaluation, the absolute performance values are expected to be lower than those reported under less restrictive settings. This is also consistent with previous leakage-controlled evaluations. For example, in the CataPro study, when a strict sequence-clustered evaluation was applied, the reported PCC was approximately 0.413, which is comparable to the performance observed in our benchmark. Therefore, the PCC values reported here should not be directly compared with higher values obtained under different or less restrictive data-splitting strategies. Our aim was to compare all models fairly under the same curated dataset, same CD-HIT 40% sequence-clustered split, and same retraining protocol.

Under this unified benchmark, DLCatalysis achieved a 10-fold CV PCC of 0.419, compared with UniKP (0.391), EITLEM-Kinetics (0.363), CataPro (0.346), and DLkcat (0.194). In addition,

DLcatalysis directly predicts k_{cat}/K_m in a single end-to-end framework, avoiding the need for separate intermediate k_{cat} and K_m modelling or additional correction/transfer steps used in staged frameworks such as EITLEM-Kinetics and CataPro. This simplified design reduces computational complexity while maintaining strong performance in our benchmark.

Q4

Comment:

The authors provide no justification for their belief that 162 experiments would have been necessary to obtain 12 positive variants without deep learning.

Response:

We thank the reviewer for pointing this out. In the revised manuscript, we have removed the previous insufficiently justified estimate and redesigned the model-application strategy. Instead of making a hypothetical comparison, we now describe the actual SEED workflow used in the revised study.

In this new strategy, PROSS and DeMaSk first generated a non-redundant library of 716 candidate single-point variants. Rather than experimentally testing this full candidate space, we first selected 20 variants for initial experimental validation and used the measured k_{cat}/K_m values to update DLcatalysis. The retrained model was then used to rank all possible AGE^{Bf} single-substitution variants *in silico*, from which another 20 top-ranked variants were selected for experimental testing. Positive variants were then further combined through iterative mutagenesis. Overall, approximately 50 experimental tests were sufficient to identify variants with improved catalytic efficiency, including AGE^{Bf}M9, which showed the highest activity among the tested combinations.

Thus, the revised workflow demonstrates the practical value of DLcatalysis in improving experimental efficiency: from 716 candidate substitutions, only a small experimentally prioritized subset was tested, yet higher-activity mutants were successfully identified. We have revised the manuscript to present this as an experimentally supported model-guided screening strategy, rather than as an unsupported estimate of how many experiments would otherwise have been required.

To make this clear, we have added the following text in line 313-352:

“Deep learning-guided engineering of the key NeuAc biosynthetic module B. Although AGE^{Bf} exhibited higher catalytic efficiency than AGE^{Ana}, its turnover under the tested conditions remained insufficient to fully relieve the rate-limiting step in NeuAc biosynthesis, and its use was further limited by modest soluble expression. We therefore developed a Sequential Experimentally Enhanced Design (SEED) workflow (Fig.4a) to explore the AGE^{Bf} single-

substitution landscape by combining computational screening, experimental testing and DLcatalysis-guided selection.

In the SEED workflow, candidate substitutions were first selected using two complementary computational tools, PROSS⁴¹ and DeMaSk⁴², which evaluate mutations from different perspectives. PROSS was used to identify substitutions predicted to improve protein stability, using an AlphaFold3 model of AGE^{Bf} as input because no experimentally determined AGE^{Bf} structure is currently available in the PDB. In parallel, DeMaSk was used to identify substitutions predicted to be evolutionarily favourable. PROSS identified 40 substitutions (Supplementary Table. S5), whereas DeMaSk identified 688 substitutions with positive evolutionary scores (Supplementary Fig. S4). Twelve substitutions were shared by both methods, resulting in a non-redundant library of 716 candidate single-point variants.

To provide experimental data for model updating, we selected the ten highest-ranked variants from each method, giving 20 candidates in total, and measured their $k_{cat} K_m^{-1}$. All mutant candidates were expressed in *E. coli* BL21(DE3) using pET-28a vector to verify their catalytic activities. Among these mutant candidates, 5 mutants (K170R, Y220F, T227S, S296M, V402L) exhibited higher activity than AGE^{Bf} (Supplementary Fig. S5). The resulting $k_{cat} K_m^{-1}$ values of 20 variants and WT were added to the training dataset, and DLcatalysis was retrained using the same training settings as the baseline model. The PCC and RMSD reached 0.428 and 0.159 after hyperparameter optimisation (Supplementary Fig. S6a-h and S7a-b), respectively. We next generated all possible single-point substitutions across AGE^{Bf} in silico and used the retrained DLcatalysis model to predict their $k_{cat} K_m^{-1}$ values. Then top 20 ranking variants were selected to test the catalysis activity. 9 mutants (D2R, D2C, S3W, S3C, S3H, N5R, N6W, N6R, K411R) exhibited higher activity than AGE^{Bf} (Supplementary Fig. S5). Then 6 mutants which $k_{cat} K_m^{-1}$ are 20% higher than AGE^{Bf} were selected to do the iterative mutagenesis with AGE^{BfK170R} used as the initial template. As the fig.4b shown, the activity of AGE^{BfM9} (N6R, K170R, S296M, V402L) was highest among all the variants which is 28476.34 $k_{cat} K_m^{-1}$. To assess the impact of AGE^{BfM9} on NeuAc production, the plasmid was developed and introduced into EMSA12, generating EMSA12-M. As shown in Fig. 4c, AGE^{BfM9} improved the conversion efficiency of GlcNAc to NeuAc to various extents, which increased NeuAc production yield to 8.54 g L⁻¹, reduced GlcNAc titer to 2.37 g L⁻¹ indicating that the improvement of the catalytic efficiency of AGE is crucial for enhancing the metabolic flux from GlcNAc to NeuAc. While ManNAc titer was increased to 1.43 g L⁻¹, suggesting that an insufficient supply of PEP could be another potential bottleneck in NeuAc biosynthesis. ”

Comment:

If the authors chose not to use DLKcat at this stage due to its limited capabilities, it is unclear why it was still employed elsewhere in the pipeline. Additionally, we don't understand what is meant by "eliminating interference". We also still do not understand why the authors did not apply their own kcat/KM prediction models at this stage of the workflow. The criteria for deciding when to integrate these prediction models into enzyme screening and when to omit them should be clearly explained and justified.

Response:

We thank the reviewer for pointing out this ambiguity. We have revised the manuscript to clarify the role of different prediction tools at different stages of the study. In the early stage of pathway construction, DLKcat/GotEnzymes was used only as an initial exploratory attempt to prioritize several candidate enzymes for pathway assembly. This was part of the actual research process when we were still identifying which pathway step limited NeuAc production, rather than the final model-guided enzyme-engineering strategy.

After experimental pathway optimization revealed that GlcNAc-to-ManNAc conversion remained a major bottleneck, our objective changed from broad enzyme screening to catalytic-efficiency-guided enzyme mining and mutational design. At this stage, we developed and applied DLCatalysis, a direct kcat/Km prediction model, to assist enzyme selection and reduce experimental workload. We retained the early exploratory DLKcat/GotEnzymes-based result because it reflects the actual research process. We did not hide this result or retrospectively repeat the initial enzyme-combination screening with DLCatalysis, because doing so would make the workflow appear artificially reconstructed rather than experimentally driven.

In the revised workflow, DLCatalysis was applied specifically when catalytic-efficiency ranking was required, including AGE homolog selection, AGE^{Bf} mutational design, and NeuBNm tuning. This distinction has now been clarified in the revised manuscript.

Response to reviewers' comments point by point:

Response to Reviewer #2

We sincerely thank the reviewer for the positive reassessment of our revised manuscript and for recognizing the improvements in the DLCatalysis description, model benchmarking, mechanistic analysis, and experimental validation. We are grateful for the constructive suggestion regarding the future extensibility of the proposed computational-metabolic engineering framework.

Following the reviewer's suggestion, we have added a brief discussion to clarify that the current study validates the framework on the NeuAc pathway and that future work will focus on testing its generalizability across different enzymes and biosynthetic products. Specifically, we now discuss the potential application of the CLEAN-DLCatalysis-SEED strategy to other rate-limiting enzymes, including cytochrome P450 variants for plant natural product biosynthesis, reductases in carotenoid or astaxanthin biosynthesis, and carboxylation enzymes relevant to CO₂ fixation. This addition highlights that the present work establishes a proof-of-concept framework, while broader pathway-level validation will be an important direction for future studies.

To address this point, we added the following text to the Discussion section:

“DLCatalysis offers several key advantages for enzyme discovery and engineering. The incorporation of MSA Transformer features into DLCatalysis was inspired by a key principle underlying AlphaFold, which evolutionary information encoded in multiple sequence alignments can provide powerful structural and functional constraints for protein and enzyme modelling. By integrating residue-level protein language model embeddings, evolutionary information, multimodal substrate representations, and bidirectional enzyme–substrate cross-attention, the framework explicitly captures sequence, substrate, and interaction-dependent determinants of catalysis. Its sample wise dynamic gating strategy further enables flexible integration of enzyme baseline, substrate-dependent, and interaction terms. Together, these features improve predictive performance over existing models and provide a practical computational framework for identifying and optimizing enzymes with enhanced catalytic efficiency.”

DLcatalysis predicts $k_{cat}K_m^{-1}$ within a single architecture by combining residue-token cross-attention with a three-branch residual head whose substrate and interaction contributions are modulated by a sample-wise gate, and by training jointly with enzyme and substrate margin-ranking objectives so that the model is exposed to ordinal structure that pointwise regression alone does not provide. On the unified evaluation set, DLcatalysis improved Pearson correlation over UniKP and DLkcat respectively, while operating in one prediction pass rather than the multi-stage training of k_{cat} and K_m models that has been used in representative prior approaches such as CataPro and EITLEM. The absolute correlation, however, has not reached a level that

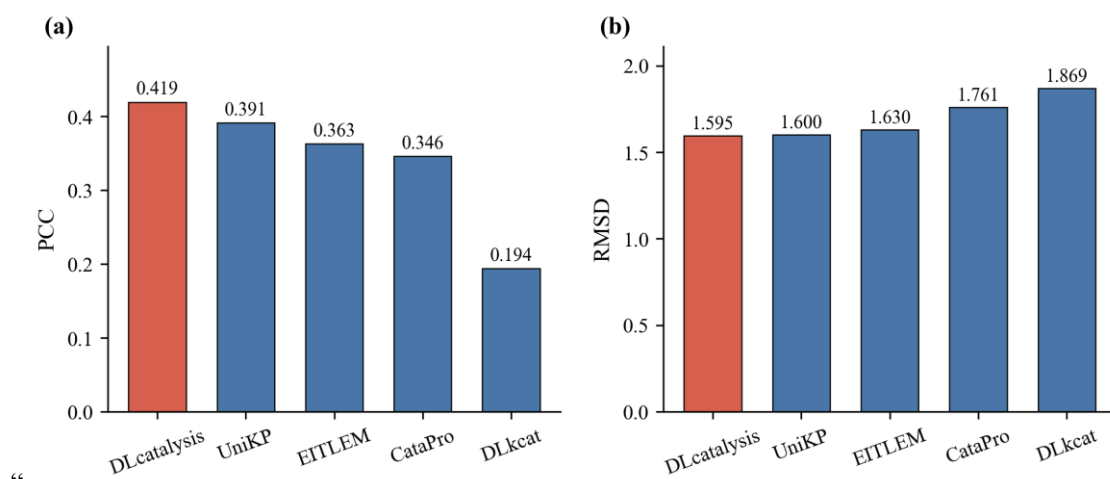
satisfies us: at PCC the model orders catalytic efficiencies usefully but does not match the precision required to fully replace experimental screening, and the gap is most pronounced for enzyme families whose activity depends on long-range conformational effects that one-dimensional sequence representations only weakly encode.

Closing this gap is unlikely to come from larger sequence-only models. Catalytic efficiency is shaped by active-site geometry, substrate access trajectories, and the conformational ensemble that the enzyme samples around the transition state properties that sequence alone does not encode. We are pursuing an extension that folds first-principles signals from the catalytic process (transition-state geometries, quantum-mechanically informed reaction profiles, and short MD-derived flexibility statistics around the active site) into the same residue-token cross-attention framework, so that prediction is conditioned on both sequence context and the three-dimensional reactive environment. With such structurally and dynamically grounded inputs, the framework is well placed for enzymes whose catalytic optimisation has long been bottlenecked by the cost of experimental screening, including cytochrome P450 variants for selective oxidative functionalisation, RuBisCO variants for improved CO₂ fixation, and homologous mining of catalytically superior variants for other key biosynthetic enzymes.”

Please see Lines 489-526 in the revised manuscript.

Specific minor point:

We also thank the reviewer for pointing out the need to clarify the benchmark context in Supplementary Fig. S2c–d. We have revised the figure caption to explicitly state the size and nature of the test set used for comparing DLCatalysis, UniKP, and EITLEM.” Please see in Line 63-71 of Supporting Information.



Supplementary Fig. S3. *Comparison of DLcatalysis with existing k_{cat} K_m^{-1} predictors. (a) PCC and (b) RMSD of DLCatalysis, UniKP, EITLEM, CataPro, and DLkcat evaluated using the same curated k_{cat} K_m^{-1} dataset and the same CD-HIT 40% sequence-identity-clustered ten-fold cross-validation protocol. The dataset contained 24,810 enzyme–substrate pairs, 11,845 unique enzyme sequences, and 3,117 unique substrates. In each fold, sequence clusters rather than individual records were assigned to the test set, ensuring that no test enzyme shared more than 40% sequence identity with any training enzyme.”*