Bioinformatics Advances, 2026, **6(1)**, vbag004
https://doi.org/10.1093/bioadv/vbag004
Published: 8 January 2026
Original Article | Sequence analysis

OXFORD

EC-Bench: a benchmark for enzyme commission number prediction

Saeedeh Davoudi[1,*], Christopher S. Henry[2], Christopher S. Miller[3,*], Farnoush Banaei-Kashani[1,*]

[1]Department of Computer Science and Engineering, University of Colorado Denver, Denver, CO, 80204, United States
[2]Argonne National Laboratory, Lemont, IL, 60439, United States
[3]Department of Integrative Biology, University of Colorado Denver, Denver, CO, 80204, United States
*Corresponding authors. Saeedeh Davoudi, Department of Computer Science and Engineering, University of Colorado Denver, 1380 Lawrence Street, LW-831, Denver, CO 80204, USA. E-mail: saeedeh.davoudi@ucdenver.edu; Christopher S. Miller, Department of Integrative Biology, University of Colorado Denver, 1150 12th Street, SI 4098, Denver, CO 80204, USA. E-mail: chris.miller@ucdenver.edu; Farnoush Banaei-Kashani, Department of Computer Science and Engineering, University of Colorado Denver, 1380 Lawrence Street, Room 803, Denver, CO 80204, USA. E-mail: farnoush.banaei-kashani@ucdenver.edu.
Associate Editor: Yoshihiro Yamanishi

Abstract

Motivation: **Enzymes are proteins that catalyze specific biochemical reactions in cells. Enzyme Commission (EC) numbers are used to annotate enzymes in a four-level hierarchy that classifies enzymes based on the specific chemical reactions they catalyze. Accurate EC number prediction is essential for understanding enzyme functions. Despite the availability of numerous methods for predicting EC numbers from protein sequences, there is no unified framework for evaluating and studying such methods systematically. This gap limits the ability of the community to identify the most effective approaches for enzyme annotation.**

Results: **We introduce EC-Bench, a benchmark for EC number prediction, consisting of (i) an initial representative set of existing methods (including homology-based, deep learning, contrastive learning, and language model methods), (ii) existing and novel accuracy and efficiency performance metrics, and (iii) selected datasets to allow for comprehensive comparative study. EC-Bench is open-source and provides a framework for researchers to not only compare among existing methods objectively under uniform conditions, but also to introduce and effectively evaluate performance of new methods in a comparative framework. To demonstrate the utility of EC-Bench, we perform extensive experimentation to compare the existing EC number prediction methods and establish their advantages and disadvantages in a variety of prediction tasks, namely "exact EC number prediction," "EC number completion," and (partial or additional) "EC number recommendation." We find wide variation in the performance of different methods, but also subtle but potentially useful differences in the performance of different methods across tasks and for different parts of the EC hierarchy.**

Availability and implementation: **The benchmarking pipeline is available at https://github.com/dsaeedeh/EC-Bench.**

Introduction

Proteins are biopolymers primarily made up of 20 canonical amino acids. A significant proportion of proteins act as enzymes, which serve as catalysts to accelerate nearly all chemical reactions occurring within cells (Cooper 2000). The Enzyme Commission (EC), a four-level hierarchical classification system (e.g. 1.1.1.1), not only provides EC numbers as broadly used annotations for enzymes in genome analyses but also links these enzymes to the specific chemical reactions they catalyze within metabolic pathways (Kotera *et al.* 2004). The relationship between proteins and annotated EC numbers is complex: some proteins can have multiple EC numbers because they may perform different functions under different biological contexts, and diverse evolutionary history or protein structures can lead to similar EC number functions. Definitive enzyme annotation relies on experimental methods, which are time-consuming and struggle to keep pace with the rapidly increasing volume of newly discovered protein sequences. To address these challenges, researchers have explored various computational approaches and predictive models to infer the EC numbers for enzymes.

Traditional homology-based approaches (i.e. methods that rely on sequence similarity for EC prediction) (Altschul *et al.* 1990, Claudel-Renard *et al.* 2003, Yu *et al.* 2009, Buchfink *et al.* 2015) predict enzyme functions by aligning the enzyme sequence to those in annotated databases, presuming that similar sequences are likely to have similar evolutionary history and thus similar functions. Machine Learning (ML) models, a technology that empowers computers to learn from data and make predictions without being explicitly programmed for specific tasks,

Received: 28 July 2025. **Revised:** 1 December 2025. **Accepted:** 20 December 2025

offer new ways to predict EC numbers for enzymes by recognizing more complex patterns in the sequences rather than relying solely on sequence alignment (Shen and Chou 2007, Dalkiran *et al.* 2018, Ryu *et al.* 2019, Sanderson *et al.* 2023, Shi *et al.* 2023). Moreover, recognizing that protein sequences share structural similarities with text, some researchers have turned to language models (Vaswani 2017). These models, originally designed for Natural Language Processing (NLP), have been adapted for EC number prediction by interpreting protein sequences in a manner analogous to text (Strodthoff *et al.* 2020, Brandes *et al.* 2022, Buton *et al.* 2023, Kim *et al.* 2023). Finally, techniques like contrastive learning have been leveraged to improve EC prediction accuracy. By training models to distinguish between similar and dissimilar examples, contrastive learning models (Yu *et al.* 2023, Ayres *et al.* 2024) have pushed the boundaries of predictive performance in this domain.

Effective benchmarks have a long history of promoting and measuring advancements in protein data analysis. However, currently there is no comprehensive benchmark for EC prediction methods. The Critical Assessment of Protein Structure Prediction (CASP) (Kryshtafovych *et al.* 2019) is a benchmark that aims to evaluate the accuracy of protein structure prediction methods by providing an open-source platform for the research community to perform comparative empirical studies on structure prediction methods. Inspired by CASP, Critical Assessment of Function Annotation (CAFA) (Zhou *et al.* 2019) was designed to enable assessment of computational methods dedicated to predicting Gene Ontology (GO) terms. While historically helpful for assessing protein annotation prediction, CAFA is primarily a benchmark dataset and competition rather than a comprehensive "one-stop shop" benchmark platform (like our proposed benchmark), which in addition to a benchmark dataset offers implementation of a representative set of function annotation models in the field, and a collection of complementary metrics for fair evaluation. Such a benchmark platform allows researchers to develop and integrate new models into the benchmark system and seamlessly perform an objective comparative assessment versus state-of-the-art function annotation models. Moreover, such a benchmark platform enables studying all existing function annotation models in an objective way to settle their advantages and disadvantages for the benefit of the research community and the practitioners alike. Tasks Assessing Protein Embeddings (TAPE) (Rao *et al.* 2019) and Fitness Landscape Inference for Proteins (FLIP) (Dallago *et al.* 2021) introduce benchmarks specifically designed to evaluate the effectiveness of transformer-based protein language models across multiple biologically relevant tasks. ProteinGym (Notin *et al.* 2023) provides a collection of benchmarks specifically crafted for protein fitness prediction and design. However, no already benchmark like ProteinGym or FLIP exists for the task of predicting EC numbers based on protein sequence. CARE (Yang *et al.* 2024) is a recent benchmark for EC number prediction, covering both protein-to-EC number and reaction-to-EC number mapping. Protein sequences and their associated EC numbers used in CARE were sourced from SwissProt, the expertly curated subset of UniProtKB (UniProt Consortium 2025). While CARE evaluates the accuracy of a few models, it does not incorporate a broad range of models available in the field, limiting its comprehensiveness in benchmarking and learning from comparisons across diverse approaches.

Despite the availability of a wide array of computational methods for predicting EC numbers from protein sequences, there is no benchmarking tool that allows for systematic evaluation of these methods under uniform conditions. The current EC prediction landscape is fragmented, with different models being evaluated on different datasets, employing varied performance metrics, and utilizing different data preprocessing techniques.

This lack of standardization makes it difficult to directly compare the efficacy of different approaches, as differences in dataset selection, evaluation metrics, and preprocessing steps can all significantly impact the reported results. Consequently, lack of a benchmark for fair evaluation of the EC prediction methods makes it challenging for researchers to determine which models are truly effective for enzyme annotation.

Here, we introduce EC-Bench, an open-source benchmarking tool that includes a representative set of 10 existing EC number prediction methods selected based on a comprehensive literature review (Table 1), while providing researchers with a platform to incorporate and evaluate new methods against existing techniques under uniform assumptions. EC-Bench includes a carefully curated standardized dataset and performance metrics that enable thorough evaluation of accuracy and efficiency for EC number prediction methods. We conduct extensive experimentation using EC-Bench to compare existing EC number prediction methods, highlighting their advantages and disadvantages across varied tasks.

Materials and methods

To address the gap in evaluating EC number prediction methods, we developed EC-Bench (Figs. 1 and 2), a benchmark framework designed for consistent and thorough assessment of EC number prediction. EC-Bench consists of three key components: Benchmark Dataset (Fig. 1), Model Pretraining and Training, and Evaluation Tasks (Fig. 2), each designed to ensure objective and consistent evaluation.

Benchmark dataset

Dataset construction

We constructed pretraining, training, and test datasets suitable for evaluation of all models (Fig. 1). Protein sequences and their corresponding EC numbers were primarily sourced from UniProtKB, a high-quality database. UniProtKB consists of two major components: Swiss-Prot, a manually curated dataset with high annotation reliability, and TrEMBL, which contains automatically annotated sequences. Given the importance of using well-curated and standardized datasets for benchmarking, we carefully selected different subsets of these datasets for pretraining, training, and testing:

- *Pretraining Data*: Extracted from *TrEMBL version 2018-02*, this dataset provides a large-scale, automatically annotated collection of protein sequences. It is particularly useful for models such as ProteinBERT (Brandes *et al.* 2022) that require extensive pretraining on diverse sequence representations and functions before fine tuning.
- *Training Data*: Derived from *Swiss-Prot version 2018-02*, this manually curated dataset ensures high-quality EC number annotations. This version was chosen to include the models

Table 1 Benchmark models in EC-bench.[a]

Model	Type	Pretraining data	Training data	URL
CatFam (Yu *et al.* 2009)	Homology	–	–	http://www.bhsai.org/downloads/catfam.tar.gz
PRIAM (Claudel-Renard *et al.* 2003)	Homology	–	–	https://web.archive.org/web/20160806022844/http://priam.prabi.fr/REL_MAR13/Distribution.zip
Diamond-BLASTp (Buchfink *et al.* 2015)	Homology	–	Swiss-Prot 2018–02	https://github.com/bbuchfink/diamond/wiki
ECPred (Dalkiran *et al.* 2018)	Machine Learning	–	Swiss-Prot 2017-03	https://github.com/cansyl/ECPred
ECRECer (Shi *et al.* 2023)	Machine Learning	–	Swiss-Prot 2018–02	https://github.com/kingstdio/ECRECer/tree/main/document
DeepEC (Ryu *et al.* 2019)	Deep Learning (Convolutional Neural Network)	–	TrEMBL/Swiss-Prot 2018–02	https://bitbucket.org/kaistsystemsbiology/deepec/src/master/
DeepECTransformer (Kim *et al.* 2023)	Large Language Models	–	TrEMBL/Swiss-Prot 2018-04	https://github.com/kaistsystemsbiology/DeepProZyme
ProteinBERT (Brandes *et al.* 2022)	Large Language Models	TrEMBL 2018–02	Swiss-Prot 2018–02	https://github.com/nadavbra/protein_bert/tree/master
EnzBert (Buton *et al.* 2023)	Large Language Models	BFD (Södinglab 2019)	Swiss-Prot 2018–02	https://gitlab.inria.fr/nbuton/tfpc
CLEAN (Yu *et al.* 2023)	Contrastive Learning	–	Swiss-Prot 2018–02	https://github.com/tttianhao/CLEAN

[a] Training code for ECPred, DeepEC, and DeepECTransformer were not available and their pretrained versions were used.

DeepEC (Ryu *et al.* 2019) and ECPred (Dalkiran *et al.* 2018), which do not provide capabilities for retraining, but were originally trained on this same version.

- *Test Data*: (i) *Swiss-Prot version 2023-01* was used as the primary test dataset, as it represents a recent version available at the time of evaluation. This dataset ensures that models are tested on the latest curated sequences, reflecting real-world conditions; (ii) *Price-149*, a challenging dataset of 149 experimentally curated bacterial protein sequences. Price-149 was previously used in the ProteInfer (Sanderson *et al.* 2023) and *CLEAN* (Yu *et al.* 2023) studies and is known to contain cases where homology-based methods struggle, making it useful for evaluating model robustness.

Data preprocessing

The training and test datasets were preprocessed to improve data quality and enhance model generalization. The preprocessing steps are as follows:

1) *Removing Non-Enzyme Proteins:* Non-enzyme sequences (those without annotated EC numbers) were removed from training and test datasets.
2) *Removing Duplicate Sequences:* Removing duplicates from within pretraining, training, and test datasets prevents bias in training and evaluation.
3) *Similar Sequence Filtering across test and training data:* This step ensures that models are evaluated on their ability to generalize beyond sequences they have already encountered. Sequences were filtered out of the training dataset at similarity thresholds of 100%, 90%, 70%, 50%, and 30%. This filtering ensured that identical (100%) or similar (other thresholds) sequences to the test data were not present in the training data, preventing data leakage and overfitting. We used MMseqs2 (Mirdita *et al.* 2019) to cluster all training and test data protein sequences at each threshold. We then check all test sample clusters and remove any training data sequences found in the same cluster.
4) *Keeping Enzymes with Three-Dimensional Structures:* With the recent advent of accurate three-dimensional protein structure prediction methods, we anticipate structure-informed EC prediction will become a growing trend. Only enzymes with available AlphaFold (Jumper *et al.* 2021) structural models were kept in training and test datasets, facilitating EC-Bench's future applicability to structure-aware function prediction.

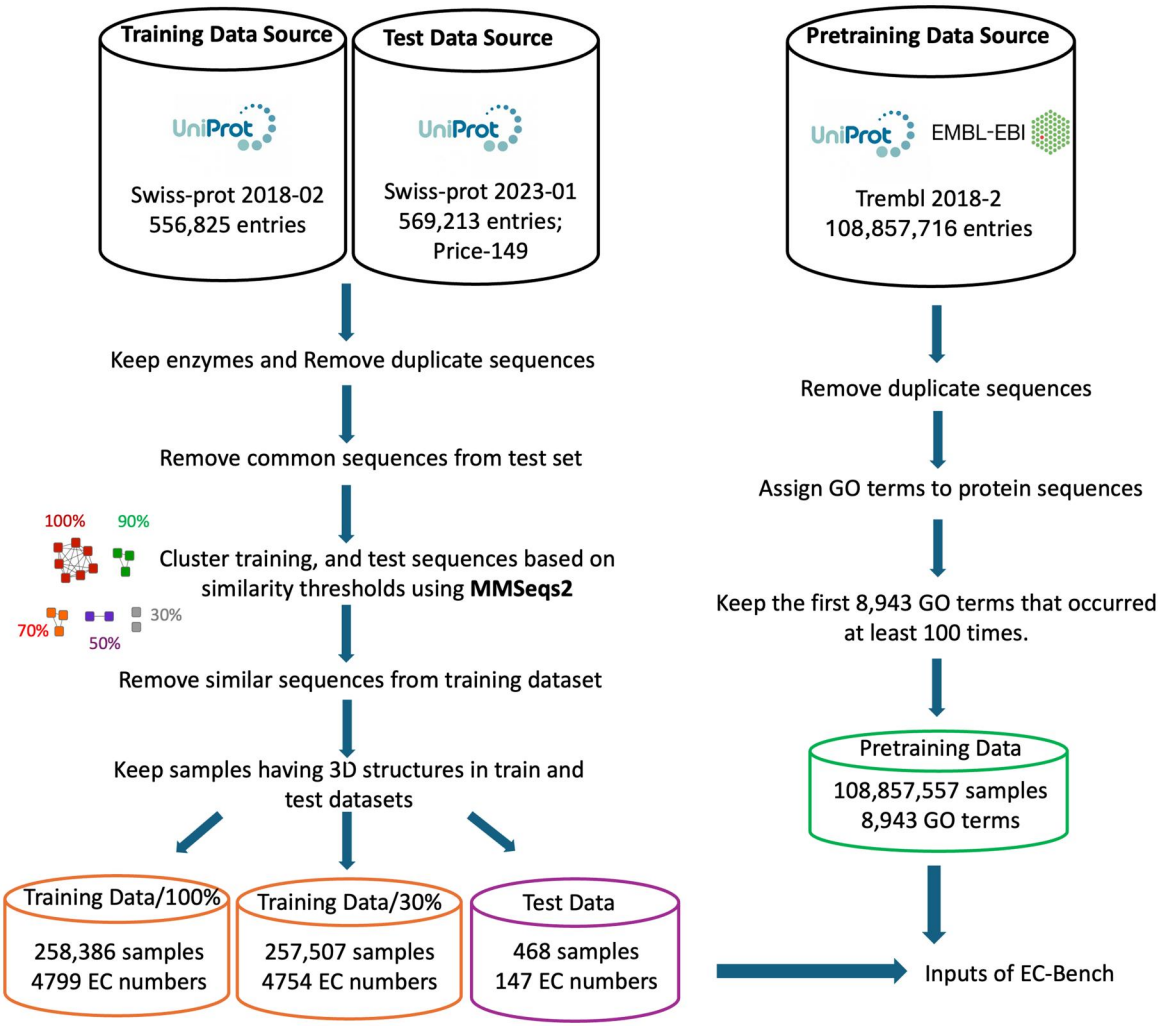

Figure 1 Overview of benchmark datasets and data preprocessing pipeline. The left side illustrates the preprocessing steps for generating training and test datasets, while the right side shows the preparation of the pretraining dataset. The final output includes two training sets (for 30% and 100% similarity thresholds across test and training data), one test set, and one pretraining set, all structured for use within the EC-Bench framework.

5) *Integration of Gene Ontology (GO) Terms:* Finally, we retrieved GO terms associated to protein sequences in the pretraining dataset from the EMBL-EBI database (Gene Ontology Consortium 2018). GO terms provide a broader functional context for models to learn from, capturing molecular functions, biological processes, and cellular components.

Table 2 shows the final dataset sizes across different similarity thresholds. Figure 16, available as supplementary data at *Bioinformatics Advances* online, also shows the distribution of taxonomic groups in the training and test datasets.

EC prediction model selection

EC-Bench includes 10 diverse models selected based on their availability, compatibility with the dataset, and prominence in the field. These models span a range of all major approaches introduced in the field, including homology-based approaches, deep learning, contrastive learning, and language models (Table 1). Additionally, we incorporated two ensemble learning methods in EC-Bench. By incorporating ensemble learning, EC-Bench evaluates whether combining methodologically diverse models can enhance EC number prediction accuracy. Two ensemble approaches are *Majority Voting*, a straightforward approach where the final prediction is determined by the most frequently predicted EC number, and *Stacking*, a more advanced method in which the predictions from a subset of models on a validation dataset are used as input features for a meta-model (see parameters in Table 1, available as supplementary data at *Bioinformatics Advances* online). This meta-model is then trained on input features to predict EC numbers on the test dataset.

EC-Bench also provides users with the flexibility to integrate and evaluate their own models within the benchmarking framework, and to compare them objectively to existing state-of-the-art approaches.

Evaluation tasks

The EC-Bench workflow (Fig. 2) begins with training (and pretraining for some models) and proceeds to evaluate models on

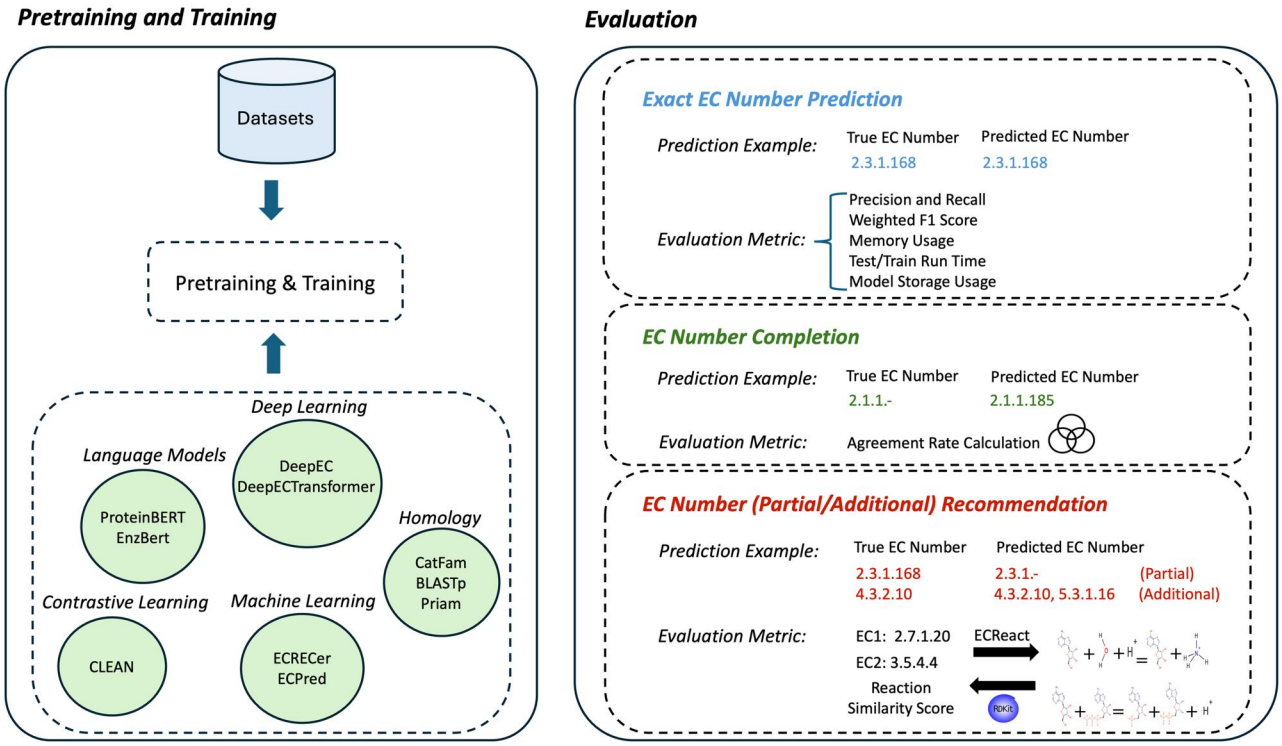

Figure 2 Pretraining, training, and evaluation tasks of EC-Bench. First, models are pretrained and trained using datasets curated by EC-Bench (left side). Then, the trained models are evaluated across three tasks (right side). For each evaluation task, an example prediction and the corresponding evaluation metrics are illustrated.

Table 2 Size of datasets per each similarity threshold.[a]

Similarity threshold	Pretraining data size	Training data size	Test data size
30%	108 857 557	257 507	468
50%	108 857 557	257 773	468
70%	108 857 557	257 986	468
90%	108 857 557	258 321	468
100%	108 857 557	258 386	468

[a] Similar sequences are only removed from the training data.

query proteins. EC-Bench employs three tasks to evaluate models (shown in different colors in Fig. 2). These tasks are designed to evaluate models in terms of accuracy, EC number completion, and the ability to recommend partial EC numbers or additional (i.e. multi-function) EC numbers.

Exact EC number prediction

This task measures how accurately models predict exact EC numbers. Since EC numbers follow a structured classification system, model performance is evaluated at increasing levels of specificity (levels 1 through 4).

Accuracy measurement

For accuracy measurement, we consider precision, recall, and weighted F1 score to compare different models in EC-Bench. Full mathematical definitions of these metrics are provided in Supplementary Materials, available as supplementary data at *Bioinformatics Advances* online.

Optimizing EC number prediction thresholds

Since enzymes can show multiple catalytic activities, EC number prediction is framed as a *multi-label classification task*. To improve classification accuracy, EC-Bench supports two thresholding strategies for converting predicted probabilities into EC number assignments. The *regular models* use fixed, probability-based rules to assign multiple EC numbers when confidence is high. The *learnt models* employ *class-specific thresholds* optimized to maximize the F1 score per EC number, improving precision and recall, especially for rare classes. A detailed formulation is provided in Supplementary Materials, available as supplementary data at *Bioinformatics Advances* online.

Performance efficiency

We also measure the computational efficiency of different models during both training and inference. We record the peak amount of runtime memory usage during model training, the total disk storage space required for the trained model file (model size), and the total duration required to train the model and to

generate predictions on the test dataset during inference (run times). Collectively, these metrics help evaluate the suitability and efficiency of the various models for deployment and/or scalability in a variety of environments and use cases.

EC number completion

EC number completion is an important task in enzyme function prediction because many enzymes in biological databases are only partially annotated. These are represented with dashes in place of the unknown levels (e.g. 1.1.-.-), indicating that while the enzyme's broad function is known (or inferred), finer details are still missing. Completing EC numbers allows researchers to infer more precise enzyme functions, which is essential for understanding metabolic pathways. In this evaluation task, we assess the ability of models to fill in missing levels of incomplete EC numbers ("-") in the test data at any level.

As we do not have the true completed EC numbers to directly validate the predictions, to evaluate models on this task we introduce a metric named *agreement rate*. The agreement rate measures the proportion of incomplete EC numbers for which most of the models converged on the same completed prediction. Specifically, it is calculated as the percentage of cases where more than half of the models returned the same EC number as their top prediction for a given incomplete entry. This metric serves as an indicator of inter-model consistency and highlights how often models independently arrive at a shared decision when faced with uncertain or under-specified input. A high agreement rate suggests that the model is making consistent and potentially confident predictions. Conversely, a low agreement rate may indicate high variability in model behavior, reflecting either uncertainty in the input data or diverse model strategies for EC number completion.

Partial/additional EC number recommendation

In many real-world biological scenarios, there are enzymes whose exact functions are unknown. Even partial EC number recommendations may be valuable for inferring participation in metabolic pathways or to suggest experimentally testable hypotheses. In contrast to *Exact EC Number Prediction* (see Exact EC number prediction section), this task evaluates the ability of models to generalize to new EC numbers (*partial*, as full EC number may be absent from the training data) and to suggest *additional* EC numbers for multi-function enzymes.

Because the true recommended EC numbers are not explicitly available for validation, and experimental validation is not feasible at scale, we need a task-specific novel evaluation metric to assess how biologically relevant the model's recommendations are. As EC numbers represent chemical reactions, we chose to evaluate predictions by comparing the similarity between the reactions of the existing true and predicted EC numbers using RDKit (RDKit Development Team 2024). Reaction similarity scores between two EC Numbers range from 0 to 1. Enzymatic reactions and their corresponding EC numbers were previously extracted from major databases and merged into a unified dataset called ECReact (Probst *et al.* 2022). ECReact was constructed by integrating reaction entries from multiple databases. Although species coverage reflects the curation biases of these individual resources, combining multiple repositories maximizes the achievable taxonomic breadth. However, we found the list of EC numbers in ECReact is incomplete. In cases where an EC number is incomplete or missing, we ignore the fourth level of the EC number and randomly sample EC numbers that share the first three levels. We compute the reaction similarity as the average similarity between the sampled EC numbers and the true EC number. If no EC numbers exist that share the first three levels, we progressively reduce the specificity by ignoring additional levels until we can calculate an average reaction similarity. For cases where there are multiple predicted EC numbers or multiple true EC numbers, we take the maximum similarity score among the EC numbers. To compare models together across all samples, we report weighted similarity score for each model:

$$\text{Coverage} = \frac{\text{Number of proteins with predictions}}{\text{Total number of proteins in the dataset}} \times 100 \quad (1)$$

$$\text{Average similarity score} = \frac{\sum \text{Similarity scores for each protein}}{\text{Number of proteins with predictions}} \quad (2)$$

$$\text{Weighted similarity score} = \text{Average similarity score} \times \frac{\text{Coverage}}{100} \quad (3)$$

Average similarity score measures the functional similarity between predicted and true EC numbers. Weighted similarity score combines accuracy (similarity score) with coverage, ensuring models that predict for more proteins are not penalized unfairly.

Results and discussion

In the following subsections, we present a detailed analysis of each evaluation task described above, integrating these findings and discussing their implications for enzyme annotation and functional prediction. We evaluate models under two sequence similarity thresholds: 100% similarity (where test sequences may have closely related training counterparts) and 30% similarity (where test sequences are more dissimilar from the training data). This comparison highlights the trade-offs between different methods, offering insights into their strengths and weaknesses in different sequence similarity scenarios and for different parts of the EC hierarchy.

Results of exact EC number prediction

Accuracy measurement

To evaluate exact EC number prediction, we analyze weighted F1 scores, precision, and recall across the four EC hierarchy levels.

Weighted F1 score

Overall, there was a surprisingly high level of variability in model performance. On the test dataset, weighted F1 scores at EC level 1 ranged from 0.164 to 0.906 at the 100% similarity threshold, and from 0.164 to 0.895 at the 30% threshold. At EC level 4, scores ranged from 0.074 to 0.639 (100%) and from 0.074 to 0.524 (30%), highlighting the increasing difficulty of fine-grained classification as sequence similarity in the training data decreases. Considering the capability of adaptive thresholding, *EnzBert-learnt* achieved the highest weighted F1 scores at levels 1, 2, and 3 for both similarity thresholds on the test dataset (Fig. 3a and b). At 100%

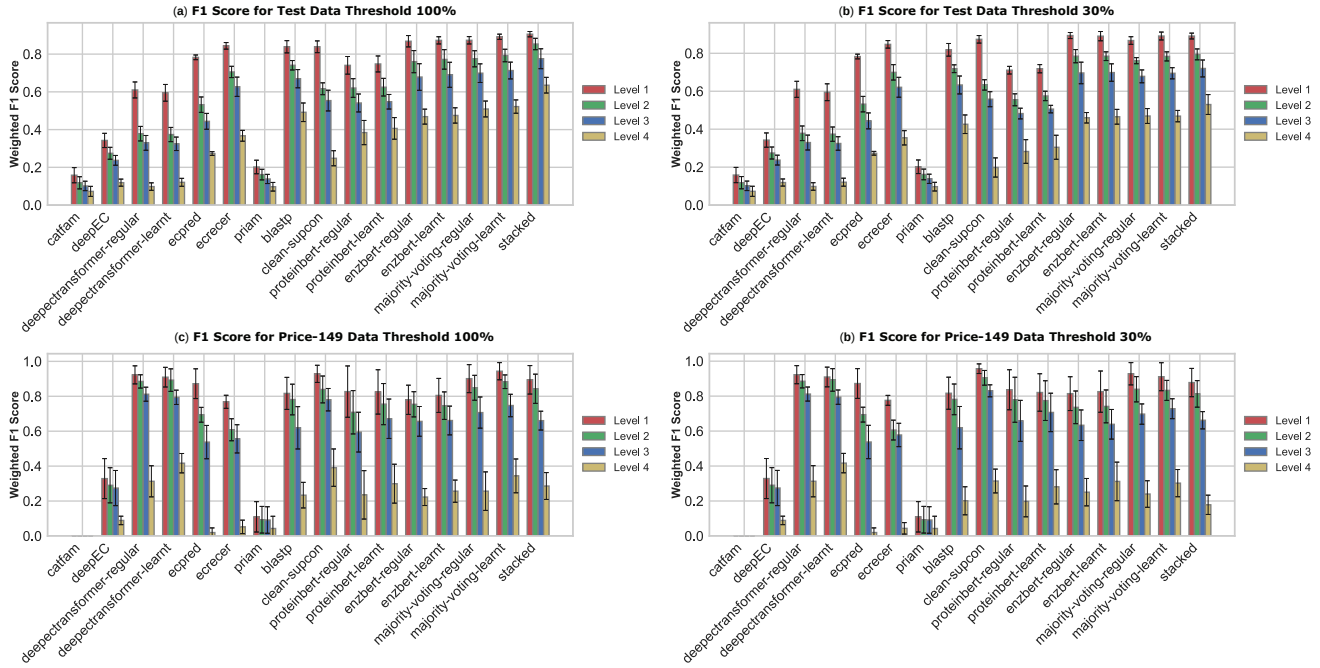

Figure 3 Weighted F1 scores with variance for exact EC number prediction across all models and test sets. Panel (a) shows results for the main test set at the 100% sequence similarity threshold, and Panel (b) at the 30% threshold. Panels (c) and (d) present the corresponding results for the Price-149 test set. Error bars represent the standard deviation of F1 scores calculated across five equal-sized, non-overlapping subsets that together cover the full test dataset, indicating performance variability.

similarity, its F1 scores for level 1, 2, and 3 are 0.873, 0.777, and 0.692, while at 30% similarity, they remain stable at 0.895, 0.794, and 0.702, respectively. At level 4 on the 100% similarity test set, the homology-based *BLASTp* slightly outperforms deep learning models, achieving a weighted F1 score of 0.502 compared to *EnzBert-learnt*'s F1 score of 0.475 (Fig. 3a). However, at 30% similarity, *EnzBert-learnt* surpasses BLASTp, achieving an F1 score of 0.463 compared to *BLASTp*'s score of (0.431) (Fig. 3b). Across both similarity thresholds, other models performed nearly as well as the top-performing models.

When threshold learning was excluded, the performance pattern closely mirrored that of the learnt models. *EnzBert-regular* remained the top performer across both similarity thresholds (Fig. 3a and b) at EC levels 1–3. At level 4, its weighted F1 score was slightly lower than that of *BLASTp* at the 100% similarity threshold, but it surpassed *BLASTp* under the more challenging 30% similarity condition.

On the more challenging Price-149 dataset, *DeepECTransformer-learnt* achieves the highest F1 score at level 4 (0.427), followed by *CLEAN* (0.406) (Fig. 3c). However, because we could not separately train *DeepECTransformer*, this model may have had data leakage. Except for *CLEAN* and *DeepECTransformer*, all models exhibit a drop in their weighted F1 scores when moving from the test dataset to the out-of-distribution Price-149 dataset (Fig. 9, available as supplementary data at *Bioinformatics Advances* online).

When threshold learning was removed, the overall pattern remained similar. *CLEAN* achieved the best results at broader EC levels (1–2) for both similarity thresholds, reaching up to 0.954 at level 1. *DeepECTransformer-regular* performed best at the more specific levels 3 and 4 (0.812 and 0.324 at 100% similarity). *EnzBert-regular* remained competitive but slightly lower, with F1 scores decreasing from 0.787 at level 1 to 0.241 at level 4. Other models such as *BLASTp*, *ECPred*, *DeepEC*, *Priam*, and *CatFam* performed notably worse, especially at lower similarity thresholds.

Collectively, these results suggest that while commonly used homology-based approaches like *BLASTp* perform well when sequence similarity to proteins in the training set is high, language models such as *EnzBert-learnt* generalize better across diverse enzyme families and are more robust when test sequences are distantly related to training sequences. Many models, like the contrastive-learning-based *CLEAN* generalize better to out-of-distribution data than homology-based models.

We also evaluated model performance across major taxonomic domains by reporting the domain composition of the training and test sets and calculating weighted F1-scores separately for Eukaryotic and Bacterial samples. Although absolute weighted F1-scores vary somewhat between domains, the overall performance trends of the models remain consistent (Supplementary Materials, Figs. 16 and 17, available as supplementary data at *Bioinformatics Advances* online).

Results for Macro and Micro averaging schemes are reported in the Supplementary Materials, available as supplementary data at *Bioinformatics Advances* online.

Weighted precision and recall trade-offs

Interesting model-type-specific patterns emerged when comparing precision and recall across the different test datasets. *BLASTp* consistently achieves the highest precision across all levels at both test set similarity thresholds (Fig. 1a and b, available as supplementary data at *Bioinformatics Advances* online). However, at level 4, its precision drops when sequence similarity to the training data decreases, from 0.612 (100% similarity) to

0.527 (30% similarity) (Fig. 1a vs. 1b, available as supplementary data at *Bioinformatics Advances* online). *CLEAN* also declines in precision at level 4, dropping from 0.488 (100%) to 0.308 (30%) (Fig. 1a vs. b, available as supplementary data at *Bioinformatics Advances* online). The LLM-based model *EnzBert (both regular and learnt)* demonstrates better precision stability across the test datasets, with only a minor drop at level 4 from 0.511 (100%) to 0.49 (30%) (Fig. 1a vs. b, available as supplementary data at *Bioinformatics Advances* online). It also maintains the highest recall across all levels, with minimal change from 100% to 30% similarity (Fig. 2a and b, available as supplementary data at *Bioinformatics Advances* online). This suggests that this LLM-based model captures functional relationships beyond strict sequence similarity. On Price-149, *CLEAN* maintains its superiority at level 4 for both precision (Fig. 1c and d, available as supplementary data at *Bioinformatics Advances* online) and recall (Fig. 2c and d, available as supplementary data at *Bioinformatics Advances* online) at 100% and 30% similarity thresholds.

Impact of ensemble learning

When sufficient training data is available, the *stacked* model outperforms all other models. With the 100% similarity threshold, the weighted F1 is 0.906 at level 1 and 0.639 at level 4 (Fig. 3a). At 30% similarity, the *stacked* model's performance declines slightly, with an F1 score of 0.892 at level 1 and 0.524 at level 4 (Fig. 3b). However, on the Price-149 dataset, the *stacked* model fails to outperform other models (Fig. 3d), likely due to its reliance on training distributions that differ from Price-149. This highlights the need for ensemble models to integrate more diverse data during training to generalize effectively to challenging datasets.

The simpler *majority voting* model does not always outperform other models, as its performance depends on the collective predictions of individual models. When most models predict an incorrect EC number, the voting mechanism reinforces this error, leading to suboptimal performance. In such cases, individual models that make correct predictions can achieve higher accuracy than the majority voting approach.

Model comparison using EC hierarchical plots

To provide a more detailed comparison between models, EC-Bench generates interactive HTML sunburst plots (see Supplementary Materials, available as supplementary data at *Bioinformatics Advances* online) that visualize the F1 scores for each EC number at each hierarchical level. For each pair of models, two sunburst plots allow side by side comparison, along with a third plot highlighting the differences in their F1 scores across all EC numbers. These visualizations offer an intuitive understanding of where in the EC hierarchy models agree and where they differ in prediction quality. For example, when comparing *BLASTp* and *CLEAN* at 100% similarity threshold, we observed that BLASTp outperforms CLEAN on EC numbers associated with fewer training samples *(e.g.* 4.2.3.127*)*. However, BLASTp tends to perform worse when predicting more frequent and diverse EC numbers such as 2.7.7.7. In contrast, when comparing *BLASTp* and *EnzBert*, the behavior of the two models is more similar overall, perhaps reflecting the complementary nature of sequence similarity-based and language-model-based approaches. Additionally, a comparison between *CLEAN* and *EnzBert-learnt* shows that *EnzBert-learnt* consistently performs better on EC numbers that have either a large number of training samples (e.g. 2.5.1.-) or very few samples (e.g. 4.2.3.127). This highlights the ability of large language models to generalize both in data-rich and data-scarce scenarios, unlike *CLEAN*, which tends to struggle on these same examples where data is highly imbalanced.

Performance efficiency

We compared the computational efficiency of all methods in training and inference. Inference run times varied over several orders of magnitude. *ECRECer* and *ProteinBERT* are among the fastest methods at inference (2 and 17 seconds on test data; Table 4). ECRECer and BLASTp require minimal memory, making them especially beneficial for large, high-similarity datasets. However, *BLASTp*'s dependence on close homology causes a considerable decline in performance at the 30% threshold (Fig. 3b and d), and ECRECer relies on Evolutionary Scale Modeling (ESM) embeddings (Meta AI 2023) which means it needs extra preprocessing before making predictions. In contrast, *EnzBert* and *ProteinBERT* demand more computational resources (Table 3) but may justify this overhead by remaining more resilient across varying similarity levels (Fig. 3a–d). Notably, *ProteinBERT* stands out for its relatively fast inference among deep learning/LLM models, thanks to a smaller parameter count (Table 4). The contrastive-learning model *CLEAN* offers memory efficiency yet requires extensive training (5160 epochs at 100% similarity and 1875 at 30%). *CLEAN* also requires ESM embeddings, which can add additional time to the overall process. This time for computing embeddings was not included in the run times reported in Tables 3 and 4. Despite this, *CLEAN*'s inference phase is relatively quick once the model is fully trained.

Table 3 Training run time, storage and memory usage for models.[a]

Model	Memory usage (GiB)	Model size (MiB)	Run time per epoch[b]	Total training run time
ECRECer	6.1	5000	–	1327s[c]
CLEAN	8.2	**4**	30s	2d [5160 epochs (100%), 1875 epochs (30%)]
ProteinBERT	40	394	150s	3.5d (81 epochs)
EnzBert	17	1640	1d	15d (15 epochs)
Diamond-BLASTp	**1.9**	681	–	**17s**

[a] Includes models trained from scratch. Bold values are used to highlight the best results.
[b] On a RTX6000-40GB GPU.
[c] d, days; s, seconds.

Results of EC number completion

In this section, we evaluate the ability of models to complete EC numbers for proteins in the test data that do not have full 4-level EC Number annotations (312 out of 468 test samples), asking them to fill in missing parts of enzyme classifications. As by definition we do not know the correct 4-level classification for this task, for evaluation purposes we compare both the number of completions made, and the agreement rate among models. *CLEAN* completes notably more EC numbers than other models at level 4 and in total (Fig. 10, available as supplementary data at *Bioinformatics Advances* online). The agreement count (shown as mean ± standard deviation) tells us how often each model matches the majority EC label. This helps us see which models usually agree with the others and which ones differ more often (Fig. 4). CLEAN has the highest average number of agreements and shows little variation across the data splits (five batches).

Table 4 Test run time for all models.[a]

Model	Run time (test dataset)
CatFam	67s[b]
DeepEC	32s
DeepECTransformer	28s
ECPred	5h
ECRECer	**2s**
PRIAM	50s
Diamond-BLASTp	30s
CLEAN	42s
ProteinBERT	17s
EnzBert	63s

[a] All models are tested using a single thread. Bold value is used to highlight the best result.
[b] h, hours; s, seconds.

This means it performs well on fully annotated proteins (see Accuracy measurement section) and stays consistent with the other models even when annotations are incomplete, making it a strong choice for filling in missing EC numbers.

To validate our method of assessing EC number completion, we also performed majority agreement evaluation on samples having completed EC numbers (e.g. known EC numbers; Fig. 15, available as supplementary data at *Bioinformatics Advances* online). We observed a strong positive correlation between agreement count and F1 score at both 100% similarity (Pearson $r = 0.78$, $P < .001$) and 30% similarity ($r = 0.81$, $P < .001$). This indicates that models more aligned with the collective prediction trend on completed EC numbers tend to have higher predictive accuracy. These results justify using majority agreement as a confidence measure for evaluating EC number completion task when the true annotation is unknown.

Results of partial EC number recommendation

We next evaluated the ability of the models to recommend new partial EC numbers (levels 1-3) for EC numbers that exist in the test data but were never seen in the training data (Fig. 11, available as supplementary data at *Bioinformatics Advances* online). Compared to other models, the *EnzBert-learnt* model predicts more closely related EC numbers across all three levels at the 30% similarity threshold. This suggests that *EnzBert-learnt* excels in handling more diverse and less similar sequences, likely due to its ability to generalize well in more complex data scenarios. However, at the 100% similarity threshold, the top three models across all levels are *majority voting-regular*, *CLEAN*, and *ProteinBERT-learnt*. The shift in model performance between thresholds highlights the varying strengths of different models depending on the nature of the input data.

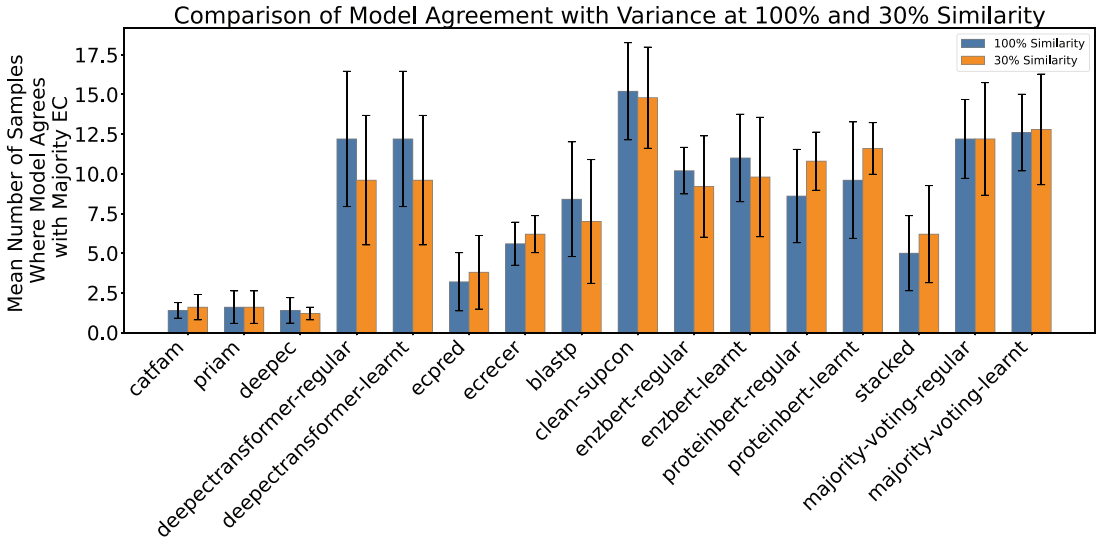

Figure 4 Model participation in majority EC number agreements with variance. For each model, the mean number of EC numbers in which it participated in a majority agreement is shown, with error bars representing the standard deviation across five equal-sized, non-overlapping batches that together cover the full test dataset. An EC number was considered to have majority agreement if more than half of the models predicted it for the same protein. This analysis reflects how consistently each model aligns with the consensus across the dataset, highlighting differences in model conformity and prediction stability.

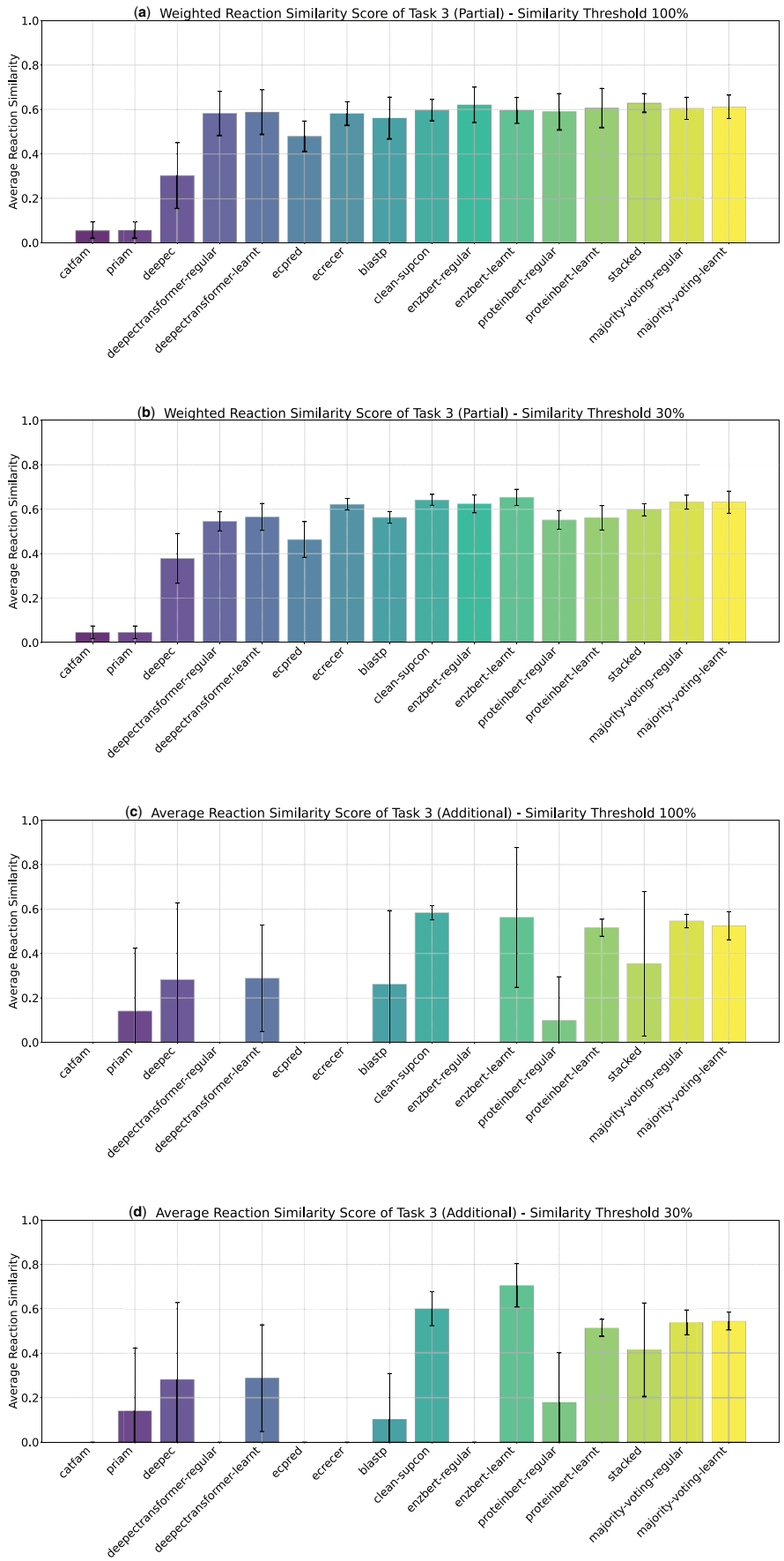

Figure 5 Reaction similarity scores for models on EC number recommendation tasks. Panels (a) and (b) show weighted reaction similarity scores with variance for partial EC number prediction at 100% and 30% sequence similarity thresholds. Panels (c) and (d) show average reaction similarity scores with variance for additional EC number prediction at 100% and 30% sequence similarity thresholds. Variance indicates the consistency of model predictions across data subsets.

Partial EC number predictions might still have value if they are "close" to correct (i.e. enzymes are predicted to perform a chemical reaction similar to the true reaction). Therefore, we also evaluated how similar predicted partial EC numbers were to the true reaction for this task. At the 100% similarity threshold, almost all models perform nearly equally well (Fig. 5a), predicting reactions similar to the truth for many of the unseen EC numbers. However, at the 30% similarity threshold, there is more variability, with some models demonstrating better generalization capabilities when challenged with lower sequence similarity (Fig. 5b).

Results of additional EC number recommendation

Some enzymes can have multiple functions, and models should be able to recommend additional functions when appropriate. We first counted cases where models correctly predicted the annotated function, but also provided additional EC Number predictions (Fig. 12a and b, available as supplementary data at *Bioinformatics Advances* online). *Majority voting* models have the tendency to predict additional EC numbers due to ties in voting; however, excluding these, *EnzBert-learnt* (at the 30% threshold) and *ProteinBERT-learnt* (at 100%) predict the most additional EC numbers. If correct, language models thus might be useful for suggesting multiple or additional enzymatic functions. It is also possible that these new EC numbers could represent corrections to the annotated function.

Under the assumption that additional new functions should be metabolically related to annotated ones (Fig. 13, available as supplementary data at *Bioinformatics Advances* online), we computed a reaction similarity score between existing and additional new annotations (Fig. 5c and d). Many models suggest additional functions with average reaction similarity scores similar to the distribution of known multi-function enzymes. Whether our assumption holds and these are indeed accurate recommendations of new additional functions will need experimental validation.

Conclusion

In this work, we introduced EC-Bench, a benchmark for evaluating EC number prediction methods. EC-Bench provides an end-to-end solution, covering data collection, preprocessing, model training, and evaluation, while enabling objective comparisons of diverse predictive models. Through extensive experiments on 10 representative models, including homology-based, deep learning, contrastive learning, and language models, we demonstrated the platform's ability to compare models under standardized conditions. Homology-based methods have been the annotation method of choice for decades. We found *BLASTp*'s precision is high when dealing with well-characterized enzyme families at high similarity, but a limitation of any benchmark is that almost all existing database annotations were assigned via homology. Despite this, language models and threshold-optimized methods excel in more diverse or novel datasets. With these caveats, minimizing false positives favors homology-based approaches, but when discovering new enzymes or functional variants, the higher recall of language models is advantageous. For some models, their agreement rates, average reaction similarity scores, and weighted similarity scores are higher under the 30% similarity threshold compared to the 100% threshold. This is somewhat counterintuitive, as more similar sequences available in training should in theory equate to better performance in testing. However, some models may also generalize better with a training set with slightly less redundancy. This behavior warrants further investigation in future work. Most of the models used here assume the training and test proteins are enzymes. The field as a whole needs to move toward tasks that also distinguish enzymes from non-enzymes, and EC bench should allow this functionality to be easily tested. Lastly, we focus here on predicting EC numbers from primary protein sequence. It would also be interesting to integrate EC number prediction from other data types like protein structures (Song *et al.* 2024) or molecular structures of substrates and/or products (Yamanishi *et al.* 2009, Reade and Chow 2023) to EC-Bench, which may complement sequence-based approaches and add value to ensemble approaches.

Acknowledgements

This work used the computing resources at the Center for Computational Mathematics, University of Colorado Denver, including the Alderaan cluster, supported by the National Science Foundation award OAC-2019089.

Supplementary material

Supplementary material is available at *Bioinformatics Advances* online.

Conflicts of interest

None declared.

Funding

This work was supported by the US Department of Energy Office of Science, Office of Biological and Environmental Research (BER), grant no. DE-SC0021350.

Data availability

The data underlying this article and full benchmark implementation with documentation are available in GitHub, at https://github.com/dsaeedeh/EC-Bench.

References

Altschul S, Gish W, Miller W *et al.* Basic local alignment search tool. *J Mol Biol* 1990;**215**:403–10.

Ayres G, Munsamy G, Heinzinger M *et al.* Annotates the microbial dark matter with HiFi-NN. *iScience* 2025;**28**:112480.

Brandes N, Ofer D, Peleg Y *et al.* ProteinBERT: a universal deep-learning model of protein sequence and function. *Bioinformatics* 2022;**38**:2102–10. https://doi.org/10.1093/bioinformatics/btac020

Buchfink B, Xie C, Huson D. Fast and sensitive protein alignment using DIAMOND. *Nat Methods* 2015;**12**:59–60.

Buton N, Coste F, Le Cunff Y. Predicting enzymatic function of protein sequences with attention. *Bioinformatics* 2023;**39**: btad620. https://doi.org/10.1093/bioinformatics/btad620

Claudel-Renard C, Chevalet C, Faraut T *et al.* Enzyme-specific profiles for genome annotation: PRIAM. *Nucleic Acids Res* 2003;**31**:6633–9. https://doi.org/10.1093/nar/gkg847

Cooper G. *The Cell: A Molecular Approach*. 2nd edn. *The Central Role of Enzymes as Biological Catalysts*. Sunderland (MA): Sinauer Associates, 2000.

Dalkiran A, Rifaioglu A, Martin M *et al.* ECPred: a tool for the prediction of the enzymatic functions of protein sequences based on the EC nomenclature. *BMC Bioinformatics* 2018;**19**:334. https://doi.org/10.1186/s12859-018-2368-y

Dallago C, Mou J, Johnston K *et al.* FLIP: Benchmark tasks in fitness landscape inference for proteins. bioRxiv 2021.11.09.467890. https://doi.org/10.1101/2021.11.09.467890

Gene Ontology Consortium. 2018. GOA UniProt. https://ftp.ebi.ac.uk/pub/databases/GO/goa/UNIPROT/, Access Date: 2024-06-14.

Jumper J, Evans R, Pritzel A *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* 2021;**596**:583–9. https://doi.org/10.1038/s41586-021-03819-2

Kim G, Kim J, Lee J *et al.* Functional annotation of enzyme-encoding genes using deep learning with transformer layers. *Nat Commun* 2023;**14**:7370. https://doi.org/10.1038/s41467-023-43216-z

Kotera M, Okuno Y, Hattori M *et al.* Computational assignment of the EC numbers for genomic-scale analysis of enzymatic reactions. *J Am Chem Soc* 2004;**126**:16487–98.

Kryshtafovych A, Schwede T, Topf M *et al.* Critical assessment of methods of protein structure prediction (CASP)-round XIII. *Proteins* 2019;**87**:1011–20. https://doi.org/10.1002/prot.25823

ESM Meta AI. 2023. Evolutionary Scale Modeling: Protein Models Repository. https://github.com/facebookresearch/esm, Access Date: 2024-02-19.

Mirdita M, Steinegger M, Söding J. MMseqs2 desktop and local web server app for fast, interactive sequence searches. *Bioinformatics* 2019;**35**:2856–8. https://doi.org/10.1093/bioinformatics/bty1057

Notin P, Kollasch A, Ritter D *et al.* ProteinGym: large-scale benchmarks for protein design and fitness prediction. *bioRxiv [Preprint]*. 2023 Dec 8:2023.12.07.570727. https://doi.org/10.1101/2023.12.07.570727

Probst D, Manica M, Nana Teukam Y *et al.* Biocatalysed synthesis planning using data-driven learning. *Nat Commun* 2022;**13**: 964. https://doi.org/10.1038/s41467-022-28536-w

Rao R, Bhattacharya N, Thomas N *et al.* Evaluating protein transfer learning with TAPE. *Adv Neural Inf Process Syst* 2019;**32**.

RDKit Development Team. *RDKit: Open-Source Cheminformatics Software*. RDKit Development Team, Zurich, Switzerland, 2024. https://www.rdkit.org/

Reade W, Chow A. *Explore Multi-Label Classification with an Enzyme Substrate Dataset*. https://kaggle.com/competitions/playground-series-s3e18. 2023. Kaggle.

Ryu J, Kim H, Lee S. Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers. *Proc Natl Acad Sci USA* 2019;**116**:13996–4001. https://doi.org/10.1073/pnas.1821905116

Sanderson T, Bileschi M, Belanger D *et al.* ProteInfer, deep neural networks for protein functional inference. *Elife* 2023;**12**: e80942. https://doi.org/10.7554/eLife.80942

Shen H, Chou K. EzyPred: a top-down approach for predicting enzyme functional classes and subclasses. *Biochem Biophys Res Commun* 2007;**364**:53–9. https://doi.org/10.1016/j.bbrc.2007.09.098

Shi Z, Deng R, Yuan Q *et al.* Enzyme commission number prediction and benchmarking with hierarchical dual-core multitask learning framework. *Research (Wash DC)* 2023;**6**:0153. https://doi.org/10.34133/research.0153

Södinglab SA. 2019. BFD: Big Fantastic Database. https://bfd.mmseqs.com/.

Song Y, Yuan Q, Chen S *et al.* Accurately predicting enzyme functions through geometric graph learning on ESMFold-predicted structures. *Nat Commun* 2024;**15**:8180. https://doi.org/10.1038/s41467-024-52533-w

Strodthoff N, Wagner P, Wenzel M *et al.* UDSMProt: universal deep sequence models for protein classification. *Bioinformatics* 2020; **36**:2401–9. https://doi.org/10.1093/bioinformatics/btaa003

UniProt Consortium. 2025. UniProtKB. https://www.uniprot.org/uniprotkb.

Vaswani A, Shazeer N, Parmar N *et al.* Attention is all you need. *Adv Neural Inf Process Syst* 2017;**30**.

Yamanishi Y, Hattori M, Kotera M *et al.* E-zyme: predicting potential EC numbers from the chemical transformation pattern of substrate-product pairs. *Bioinformatics* 2009;**25**:i179–186. https://doi.org/10.1093/bioinformatics/btp223

Yang J, Mora A, Liu S *et al.* Care: a benchmark suite for the classification and retrieval of enzymes. *Adv Neural Inf Process Syst* 2024;**37**:3094–121.

Yu C, Zavaljevski N, Desai V *et al.* Genome-wide enzyme annotation with precision control: catalytic families (catfam) databases. *Proteins* 2009;**74**:449–60.

Yu T, Cui H, Li J *et al.* Enzyme function prediction using contrastive learning. *Science* 2023;**379**:1358–63. https://doi.org/10.1126/science.adf2465

Zhou N, Jiang Y, Bergquist T *et al.* The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol* 2019;**20**:244. https://doi.org/10.1186/s13059-019-1835-8

Bioinformatics Advances, 2026, 6, 1–12
https://doi.org/10.1093/bioadv/vbag004
Original Article