ECRECer: Enzyme Commission Number Recommendation and Benchmarking based on Multiagent Dual-core Learning

Zhenkun Shi, Qianqian Yuan, Zhitao Mao, Ruoyu Wang, Hoaran Li,
Xiaoping Liao✉, Hongwu Ma✉
Biodesign Center, Key Laboratory of Systems Microbial Technology
Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, 300308, Tianjin, China
National Technology Innovation Center of Synthetic Biology, 300308, Tianjin, China
{Zhenkun.Shi, yuan_qq, mao_zt, wangry, lihr, liao_xp, ma_hw}@tib.cas.cn

Abstract

Enzyme Commission (EC) numbers, which associate a protein sequence with the biochemical reactions it catalyzes, are essential for the accurate understanding of enzyme functions and cellular metabolism. Many ab-initio computational approaches were proposed to predict EC numbers for given input sequences directly. However, the prediction performance (accuracy, recall, precision), usability, and efficiency of existing methods still have much room to be improved. Here, we report ECRECer, a cloud platform for accurately predicting EC numbers based on novel deep learning techniques. To build ECRECer, we evaluate different protein representation methods and adopt a protein language model for protein sequence embedding. After embedding, we propose a multi-agent hierarchy deep learning-based framework to learn the proposed tasks in a multi-task manner. Specifically, we used an extreme multi-label classifier to perform the EC prediction and employed a greedy strategy to integrate and fine-tune the final model. Comparative analyses against four representative methods demonstrate that ECRECer delivers the highest performance, which improves accuracy and F1 score by 70% and 20% over the state-of-the-the-art, respectively. With ECRECer, we can annotate numerous enzymes in the Swiss-Prot database with incomplete EC numbers to their full fourth level. Take UniPort protein "A0A0U5GJ41" as an example (1.14.-.-), ECRECer annotated it with "1.14.11.38", which is supported by further protein structure analysis based on AlphaFold2. Finally, we established a webserver (https://ecrecer.biodesign.ac.cn) using an entirely could-based serveless architecture and provided an offline bundle to improve usability.

***K*eywords** EC prediction, protein language model, extreme multi-label classification, deep learning

1 Introduction

With the widespread adoption of high-throughput methods and high-quality infrastructure in biotechnology and bioindustry, the speed of new protein discovery has increased dramatically. However, this was not followed by a concomitant increase in the speed of protein annotation. For example, 5 241 146 sequences were added to TrEMBL in the UniProt database [1] in the single month of March 2021, while only 521 sequences were reviewed and added to Swiss-Prot in the same period (SI, Fig. 2). Such a slow speed of protein annotation considerably restricts related research and industrial applications.

Among the multiple and complex protein annotation tasks, one of the crucial steps is enzyme function annotation [2, 3]. Annotations of enzyme function provide critical starting points for generating and testing biological hypotheses [3]. Current functional annotations of enzymes describe the biochemistry or process by assigning an Enzyme Commission (EC) number. This is a four-part code associated with a recommended name for the corresponding enzyme-catalyzed reaction that describes the enzyme class, the chemical bond acted on, the reaction, and the substrates [4]. Thus, the primary task of enzyme annotation is to assign an EC number to a given protein sequence. However, as the uncertainty of the assignments for uncharacterized protein sequences is high and biochemical data are relatively sparse, both the speed and the quality of enzyme annotation are considerably restricted.

To achieve improved, rapid and intelligent functional annotation, computational methods were introduced to assign or predict EC numbers. The simplest and most commonly used method is multiple sequence alignment (MSA) [5], which can yield an appropriate annotation by using similar sequences. Based on this approach, researchers have developed most major EC databases and profile-based methods for the functional annotation of enzymes [6, 7, 8, 9]. However, these methods cannot perform annotations for novel proteins with no similar sequences, which is generally the case for newly discovered enzymes. To overcome this restriction, researchers introduced machine learning methods, such as SVM [10], KNN [11], and hidden Markov model [12] for the functional annotation of enzymes. Although these methods can predict EC numbers even if the given protein sequences have no similar references, the prediction speed and precision are not ideal. Since deep learning has delivered powerful results in many areas [13, 14, 15, 16], more researchers are trying to use deep learning methods to predict EC numbers and significantly improve the precision of functional annotation. However, deep learning methods are prone to overfitting due to an unbalanced distribution of training datasets. In EC number prediction, this leads to prediction results with high precision, medium recall, and low accuracy.

Overall, there has been a steady improvement in computational methods for enzyme annotation [9, 7, 17, 2], but several obstacles still exist that have slowed the progress of computational enzyme function annotation. One of the direct challenges is a lack of publicly available benchmark datasets to evaluate the existing and newly proposed models, which makes it troublesome for the end-user to choose the best method in their production scenario. Another notable challenge is the lack of an efficient and universal protein sequence embedding method. Thus, researchers have to spend large amounts of time on handcrafted feature engineering to encode the sequence, such as functional domain encoding [18] and position-specific scoring matrix encoding [19], as encoding quality dramatically impacts the performance of downstream applications [20]. The third challenge is the lack of an explicitly designed method to deal with this extreme multi-label classification problem (more than 5000 EC numbers in UniProt). Thus, obtaining reliable EC number prediction results is not straightforward, and the prediction performance is not ideal. The fourth noteworthy challenge is the usability of existing tools that need refinement so that the end-user can use them smoothly even with no coding experience.

In this paper, we take a unified approach to address these challenges. For the first challenge, we constructed three standard datasets for benchmarking and evaluation. The datasets contain more than 470,000 distinct labeled protein sequences from Swiss-Prot. To address the second challenge, we introduced the cutting-edge ideology from natural language embedding for protein sequence representation. Firstly, state-of-the-art deep learning methods were evaluated and adopted for universal protein sequence embedding [21, 22]. Then, we used a feedback mechanism to choose the most suitable method in response to the downstream tasks for optimization. To address the third challenge, we proposed a Dual-core Multiagent Learning Framework (DMLF) for EC number prediction. In DMLF, we formulate the EC number prediction as a three-step hierarchical extreme multi-label classification problem. The first step predicts whether a given protein sequence is an enzyme or not. The second step predicts how many functions the enzyme can perform, i.e., multifunctional enzyme prediction. The last step predicts the exact EC number for each enzyme function. We use traditional machine learning methods in the first two steps and a novel deep learning-based extreme multi-label classifier in the last step, then use a greedy strategy to integrate these steps to maximize the EC prediction performance. To address the last challenge, we streamlined the construction process and open-sourced our codes. Moreover, we published a webserver based on a serveless architecture, so that anyone can annotate EC numbers smoothly in high-throughput, whether they have coding experience or not.

2 Methodology

2.1 Problem Formulation

In order to annotate the enzyme function of a new protein sequence, the initial and basic task is to define whether a given protein is an enzyme. Since there are numerous multifunctional enzymes, the next task is to determine the quantity of EC number. After completing the above two tasks, it is necessary is to assign an EC number to each function. Based on these considerations, we proposed three basic tasks for the functional annotation of enzymes, as shown below.

2.1.1 Enzyme or Non-enzyme Annotation.

The enzyme or non-enzyme annotation task is formulated as a binary classification problem:

$$f : X \to \{0, 1\} \tag{1}$$

where $X = \{x_1, x_2, \cdots, x_n\}, n \geq 1$ represents a group of protein sequences, and $\{0, 1\}$ is the label indicting whether a given protein is an enzyme .

2.1.2 Multifunctional Enzyme Annotation.

Multifunctional enzyme annotation is formulated as a multi-classification problem:

$$f : X \to \{1, 2, \cdots, k\}, \tag{2}$$

where k represents the maximum number of EC number for a given protein.

2.1.3 Enzyme Commission Number Assignment.

The enzyme commission number assignment task is also formulated as a multi-classification problem as defined in Eq. 3.

$$f : X \to \{1.1.1.1, 1.1.1.2, \cdots\}, \tag{3}$$

2.2 Dataset Description

To address the first challenge, we constructed three standard datasets (SI, A. Dataset). Similar to previous work [21, 26], these datasets are extracted from the Swiss-Prot database. To simulate real application scenarios as closely as possible, we did not shuffle data randomly. Instead, after data preprocessing (SI, A. Preprocessing), we organized data in chronological order. Specifically, we used a snapshot from Feb 2018 as the training dataset. The training data contains 469,134 distinct sequences in a total 556,825 records, among which 53.56% are non-enzymes, while the remaining 47.44% are enzymes. The testing data was extracted from the June 2020 snapshot and sequences that appeared in the training set were excluded. The details are listed in SI, Table 7.

■ Dataset 1: Enzyme and Non-enzyme Dataset

The training set in total has 469,134 records, 222,567 of which are enzymes, and 246,567 are non-enzymes (SI, Table 2). The testing set contains 7101 records, 3304 of which are enzymes, and the other 3797 are non-enzymes. To make the data more inclusive, we did not filter any sequence in terms of length and homology, which is different from previous studies. An enzyme is labeled as 1 and non-enzyme is labelled as 0.

■ Dataset 2: Multifunctional Enzyme Dataset

The multifunctional enzyme dataset only contains enzyme data (225,871 records). The number of EC categories ranges from 1 to 8 (SI, Table 3).

■ **Dataset 3: EC number Dataset**

Similar to the multifunctional enzyme dataset, the EC number dataset consists of 225,871 enzyme records, 222,567 of which constitute the training dataset, and the remaining 3304 are the testing dataset, covering 5111 EC numbers. The test data include 257 newly added EC numbers compared with the training data (SI, Fig. 3), which means that these EC numbers did not appear in the training process, so predictive methods cannot handle this part of the EC numbers. Thus, we excluded the sequences with these 257 EC numbers in the evaluation process.

2.3 Proposed Framework

To address the second and third challenges: lack of a generic method with high EC prediction performance and an efficient universal protein sequence embedding method, we proposed the DMLF approach, composed of an embedding core and a learning core. These two cores operate relatively independently. The embedding core is responsible for embedding protein sequences into a machine-readable matrix. The learning core is responsible for solving specific downstream biological tasks (e.g., enzyme and non-enzyme prediction, multifunctional enzyme prediction, and EC number prediction). The overall scheme of DMLF is illustrated in Fig. 1.

■ **Core 1: Embedding**

The objective of this core is to calculate the embedding representations for protein sequences. For protein sequence encoding/embedding, recent studies have shown the superior performance of deep learning-based methods compared to traditional methods [23, 24]. Accordingly, we only compared one-hot encoding to show the difference between these two kinds of embedding in this study. Here, we adopted three different embedding methods to calculate the sequence embedding patterns that adequately represent protein sequences. The first one is commonly the used one-hot encoding [25]. The second is Unirep [21], an mLSTM "babbler" deep representation learner for proteins. We used the last layer for protein representation. The third is the evolutionary scale modeling embedding method (ESM) [22], a pretrained transformer language model for protein representation. We used the hidden states from the 1st, 32nd, 33rd layers as protein embeddings.

■ **Core 2: Learning**

The learning core is specialized to perform specific biological tasks using different agents. In this work, the learning core includes three agents. Agent-1 is a binary classifier that performs enzyme or non-enzyme prediction. This classifier was constructed using KNN [26]. Agent-2 is a multi-classifier that predicts the

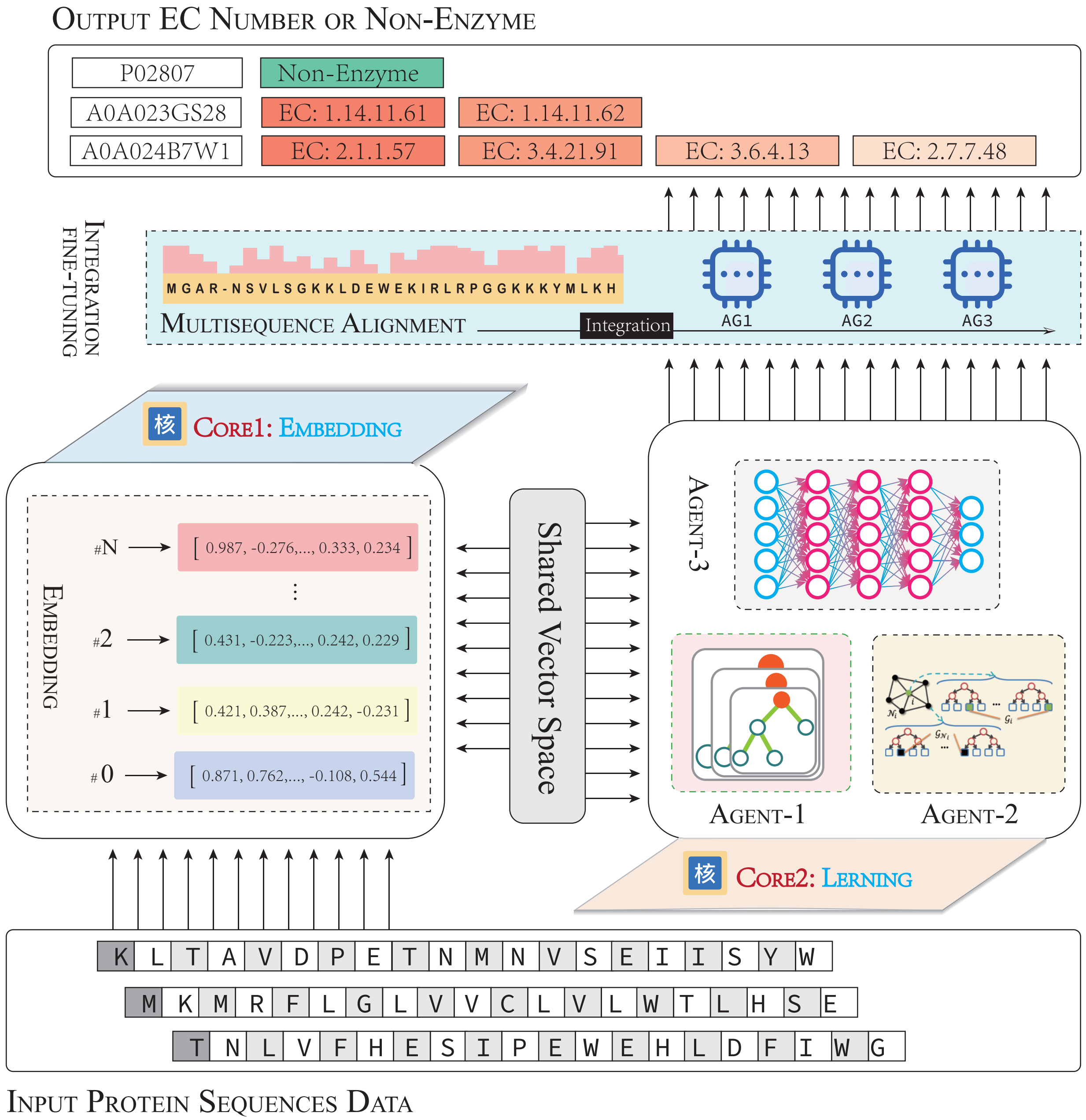


Figure 1: DMLF is an explicitly designed dual-core driven framework for EC number prediction. It consists of 2 independent operation units - an embedding core and a learning core. The embedding core is tasked with converting protein sequences into features. The learning core is designed to address the specific biological tasks defined in the problem formulation section. We use different agents to solve different tasks. Agent 1 was designed to solve the enzyme or non-enzyme classification task, agent 2 was designed to solve the multifunctional enzyme prediction task, and agent 3 was designed to solve the EC number assignment task.

number of putative functions for a given enzyme. It was implemented using an integrated sequence aligner, a gradient boost decision tree, and XGBoost. Agent-3 is also a multi-classifier that performs the EC number prediction task. As EC number prediction is an extreme multilabel classification (5852 classes in this benchmark), the performance of traditional multilabel classification methods such XGBoost, decision tree, and SVM is abysmal (less than 5% in terms of accuracy). Therefore, we trained a scalable linear extreme classifier (SLICE)[27] to obtain a more reliable classification performance in this study. The details of agent implementation and parameter settings can be found in SI, C. Models.

■ **Integration, fine-tuning and output** As illustrated in Fig. 1, the final EC number prediction output is an integrated process. As shown in Eq. 4, we formulated this integrated process as an optimization problem:

$$\underset{F1}{MAX}\{f(ag_1, ag_2, ag_3, sa)\} \tag{4}$$

where ag_1, ag_2, and ag_3 are the respective prediction results from Agent-1, Agent-2, and Agent-3, while sa is the predicted result from multiple sequence alignment. The integration and fine-tuning process aims to maximize the optimizing objective. In this work, the objective is the performance of EC number prediction in terms of the F1 score. We used a greedy strategy to perform this optimization.

2.4 Compared Baselines

To evaluate our proposed method comprehensively, we compared our proposed method with four existing state-of-the-art techniques with 'GOOD' usability (SI, RELATED WORK). Four state-of-the-art techniques are: CatFam, PRIAM (version 2), ECPred, and DeepEC.

2.5 Evaluation Metrics

To comprehensively evaluate the proposed method and existing baselines, we use 5 metrics to evaluate binary classification problems and 4 metrics to evaluate multiple classification problems. For the binary classification task, the evaluation criteria include ACC(accuracy), PPV (positive predictive value, precision), NPV(negative predictive value), RC (recall), and F1 value:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN + UP + UN} \tag{5}$$

$$PPV = \frac{TP}{TP + FP} \tag{6}$$

$$NPV = \frac{TN}{TN + FN} \tag{7}$$

$$Recall = \frac{TP}{TP + FN + UP} \tag{8}$$

$$F1 = \frac{2 \times PPV \times Recall}{PPV + Recall} \tag{9}$$

where TP is the true positive value, FP is the false positive value, TN is the true negative value, FN is false negative value, UP is unclassified positive samples, and UN is unclassified negative samples.

For multiple classification problems, the evaluation criteria included mACC (macro-average accuracy), mPR(macro-average precision), mRecall(macro-average recall), and mF1(macro-average F1 value):

$$mACC = \frac{\sum_{i=1}^{n} ACC_i}{n}, n = 1, 2, 3, \cdots, N \tag{10}$$

$$mPR = \frac{\sum_{i=1}^{n} PPV_i}{n}, n = 1, 2, 3, \cdots, N \tag{11}$$

$$mRecall = \frac{\sum_{i=1}^{n} Recall_i}{n}, n = 1, 2, 3, \cdots, N \tag{12}$$

$$mF1 = \frac{2 \times mPR \times mRecall}{mPR + mRecall} \tag{13}$$

where N represents the total number of classes, while ACC_i, PPV_i, and $Recall_i$ represent the accuracy, precision, and recall of the i-th class in a one-VS-all mode[28], respectively.

3 Results

3.1 Embedding Core Performance Evaluation

We evaluated five different protein embedding methods, one-hot embedding, Unirep embedding, and ESM embedding with three different layers (1,32,33) in our three proposed tasks. We used six machine learning baselines, including K-nearest neighbor (KNN), logistic regression (LR), XGBoost, decision tree (DT), random forest (RF), and gradient boosting decision tree (GBDT) to conduct this evaluation. For embedding, ESM-32 exhibited the best overall performance among all six baselines regarding all evaluation metrics for embedding (SI, Tables 12 and 13). As shown in Fig. 3, in task 1, ESM-32 achieved 21.67 and 6.03% improvements over

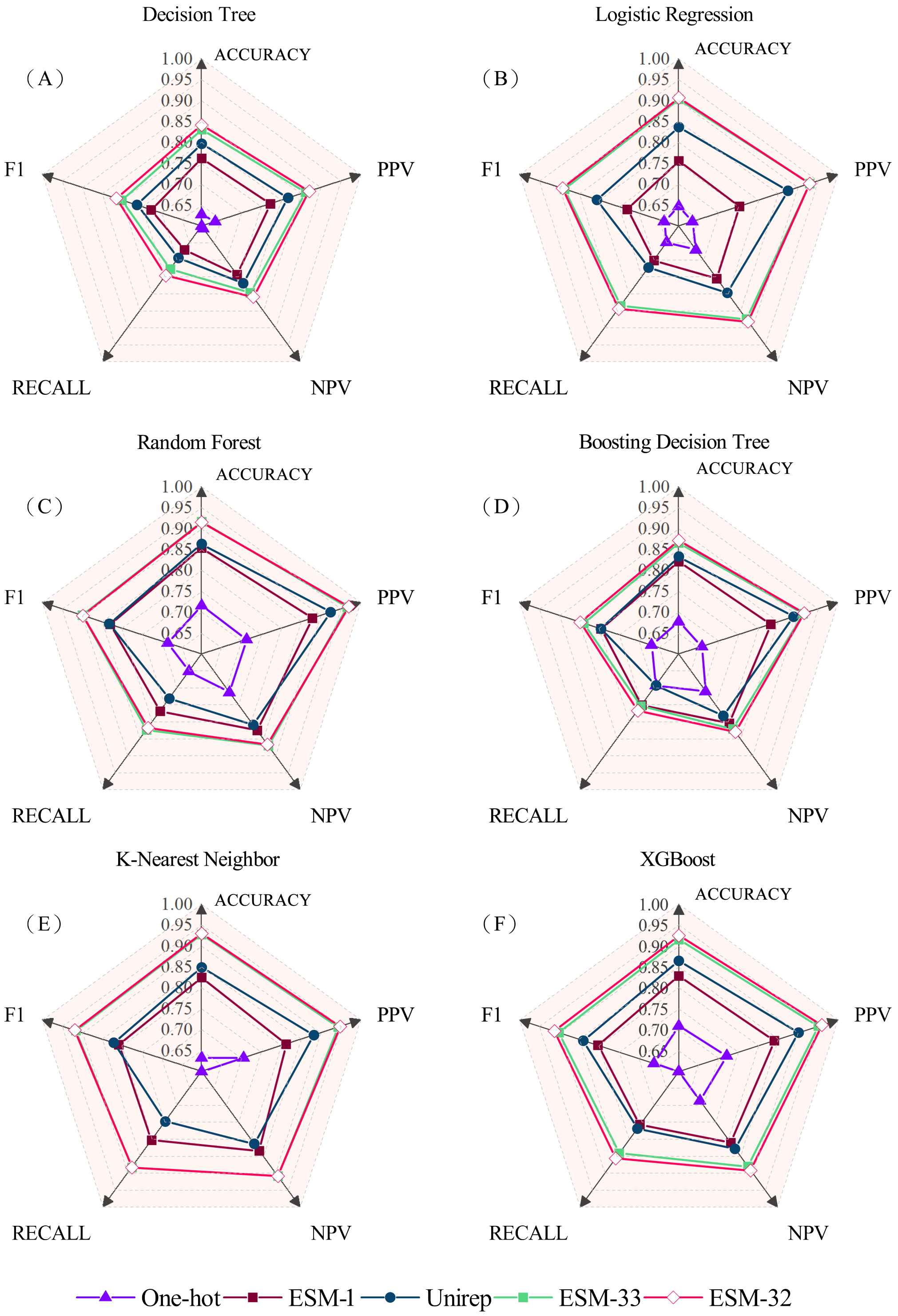


Figure 2: Performance comparison of different embedding methods for enzyme or non-enzyme annotation.

one-hot and Unirep in terms of accuracy, as well as 27.20 and 7.32% in terms of F1, respectively (SI, Table 11). This experiment suggests that better embedding can lead to better learning performance, and deep latent representation can comprehensively represent the protein sequence. The embedding performance of ESM-32 was better than that of ESM-33, suggesting that a deeper embedding layer is not always better. DMLF can automatically choose the best embedding methods based on the downstream tasks, and ESM-32 exhibited the best performance in this work.

3.2 Task 1: Enzyme or Non-enzyme Prediction

In this work, the workflow of enzyme number assignment is: *task 1*, determine whether the given protein sequence is an enzyme → task 2, if the given protein is an enzyme, then predict how many enzyme functions it can perform; → *task3*, assign an EC number for each enzyme function. According to this workflow, the first benchmarking task is enzyme or non-enzyme prediction. In this task, we trained an integrated binary classification model, which is driven by KNN and sequence alignment. KNN was implemented using scikit-learn, and the alignment was implemented using diamond v2.0.11.

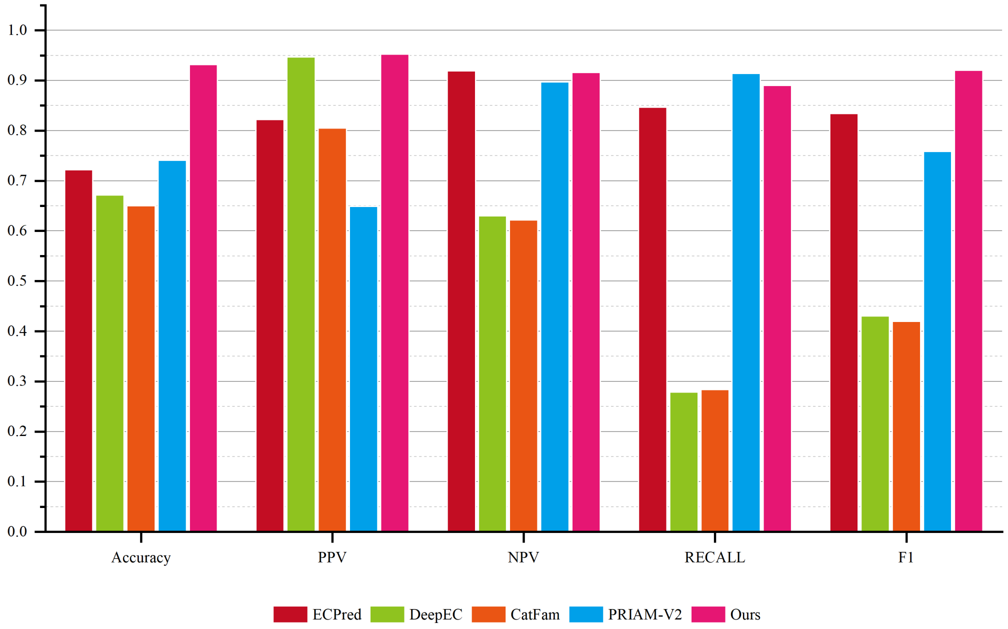


Figure 3: Task 1 Comparison of enzyme or non-enzyme annotation

As shown in Fig. 3, our method can achieve scores of 93.12, 95.25, and 88.99% in terms of accuracy, precision, and recall, respectively (SI, Table 8). Compared with other state-of-the-art tools and techniques the overall accuracy was greatly improved. For example, DeepEC yielded 74.10%, compared with 93.12% using our algorithm. Many previous methods were designed to obtain high precision while neglecting accuracy, NPV, and recall. For example, DeepEC can reach 94.68% precision while recall is only 20.83%. Methods that only offer high precision are very likely to miss many new functions. The F1 score might be a better evaluation metric for the EC assignment of real-world proteins.

3.3 Task 2: Multifunctional Enzyme Prediction

The second benchmarking task we addressed is multifunctional enzyme prediction. The backward prediction engine is agent 2 (Fig. 1). In this task, we trained an integrated multiple-classification model driven by sequence alignment and XGBoost. The results demonstrated that our method was superior to existing baselines (SI, Table 9). Our method achieved 91.71% accuracy with 58.37% mPR and mRecall% recall. The low mRecall and mPR are mainly due to the data sparseness of 3-8 functional enzymes (8.55‰ of all enzymes), which results in the classifier being more prefer to predict an enzyme as a single function enzyme. PRIAM and DeepEC are not explicitly designed to predict multifunctional enzymes, so the performance is deficient when dealing with multifunctional enzyme prediction (PRIAM: accuracy 13.11%, mPR 22.92%, mRecall 4.46%, mF1 7.47%; DeepEC: accuracy 22.16%, mPR 12.38%, mRecall 3.30%, mF1 5.22%). ECPred and CatFam were proposed to solve singlefunctional enzyme prediction, so they are not applicable in this task. Hence, the performance of existing methods were notably insufficient when dealing with multifunctional enzyme prediction. The low performance is mainly due to a lack of multifunctional enzyme data (SI, Table 3). Although our proposed method is 6.3 times better than random prediction in terms of accuracy, the performance is still insufficient, so it should be further improved in future work.

3.4 Task 3: Enzyme Commission Number Prediction

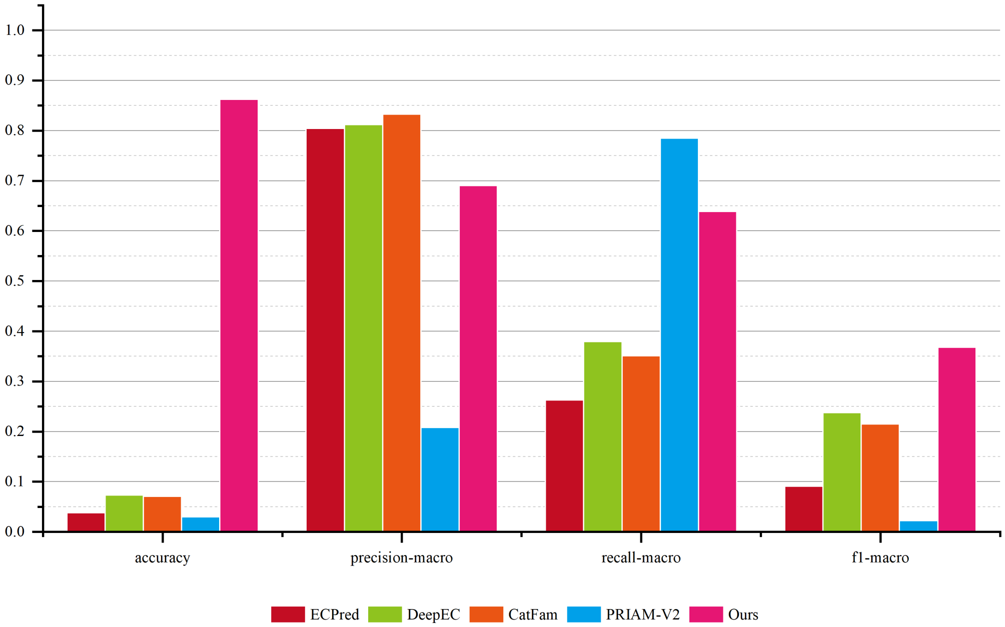


Figure 4: Task 3 Comparison of EC number prediction results.

This task corresponds to agent-3 in DMLF. In order to develop a balanced EC number prediction algorithm with high accuracy combined with reasonable precision and recall, we trained an extreme multi-label classification model.

Our method achieved 86.91% accuracy with 69% precision and 63.88% recall (Fig. 4), which means that if 100 protein sequences were uploaded for annotation, we can obtain approximately 87 correct annotations. PRIAM is mainly designed to include more sequences, so the recall is high (78.48%), while the accuracy (3.0%) and precision (20.80%) are very low. DeepEC, ECPred, and CATFAM pursue high precision, so the accuracy is very low (less than 7.5%), which means that if we upload 100 protein sequences for annotation, we can only obtain 7.5 correct annotations while the remaining 92.5 are wrong. Obviously, our method shows a clear advantage in terms of EC number assignment.

4 Web Server Implementation

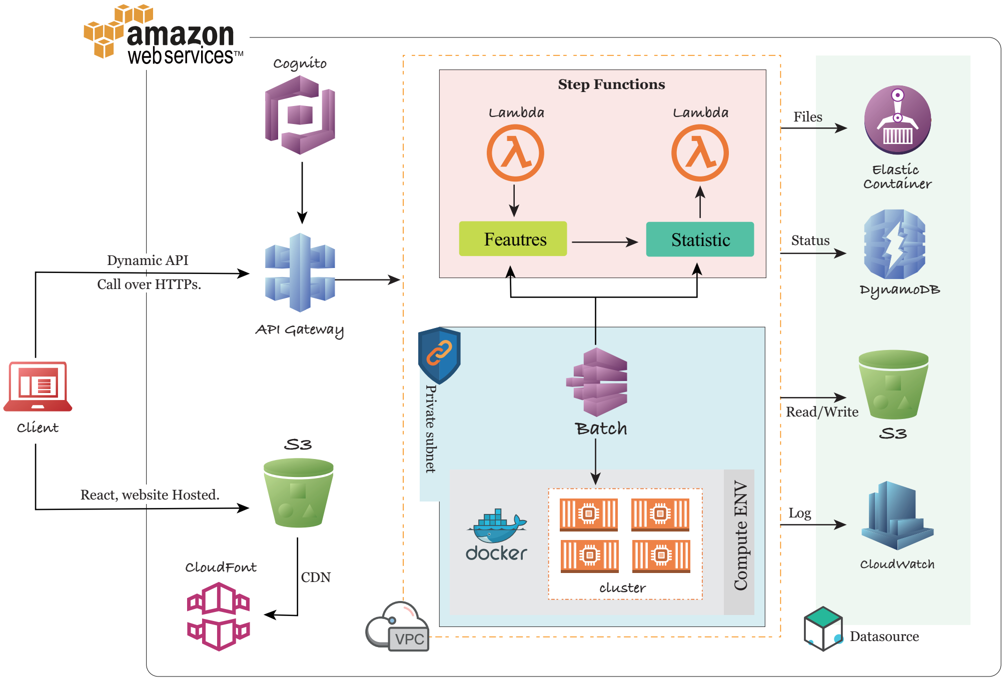


Figure 5: The architecture of the web platform.

To make the workflow accessible for biologists around the world, we built a web application using an entirely cloud-based serverless architecture (Fig. 5), offering high reliability, robustness and scalability. End-users can simply upload sequences to our platform, and then click the submit button to trigger the prediction workflow. In general, the whole workflow can be completed in a few seconds. We use Amazon DynamoDB to store job information, and users can track the previous submission records and corresponding status information. Once the analysis is finished, the user can view or download the corresponding results.

For EC assignment workflow, we use Amazon ECR to store Docker images, which packages a set of bioinformatics software, such as diamond and in-house python scripts. We built a scalable, elastic, and easily maintainable batch engine using AWS Batch. This solution took care of dynamically scaling our computer resources in response to the number of runnable jobs in our job queue. Finally, we used AWS step functions to coordinate the components of our applications easily, process messages passed from AWS API Gateway, and invoke the workflows asynchronously. AWS API Gateway was used as the API server to handle the HTTP requests and route traffic to the correct backends. The static website was hosted by AWS S3 and sped up using AWS CloudFront.

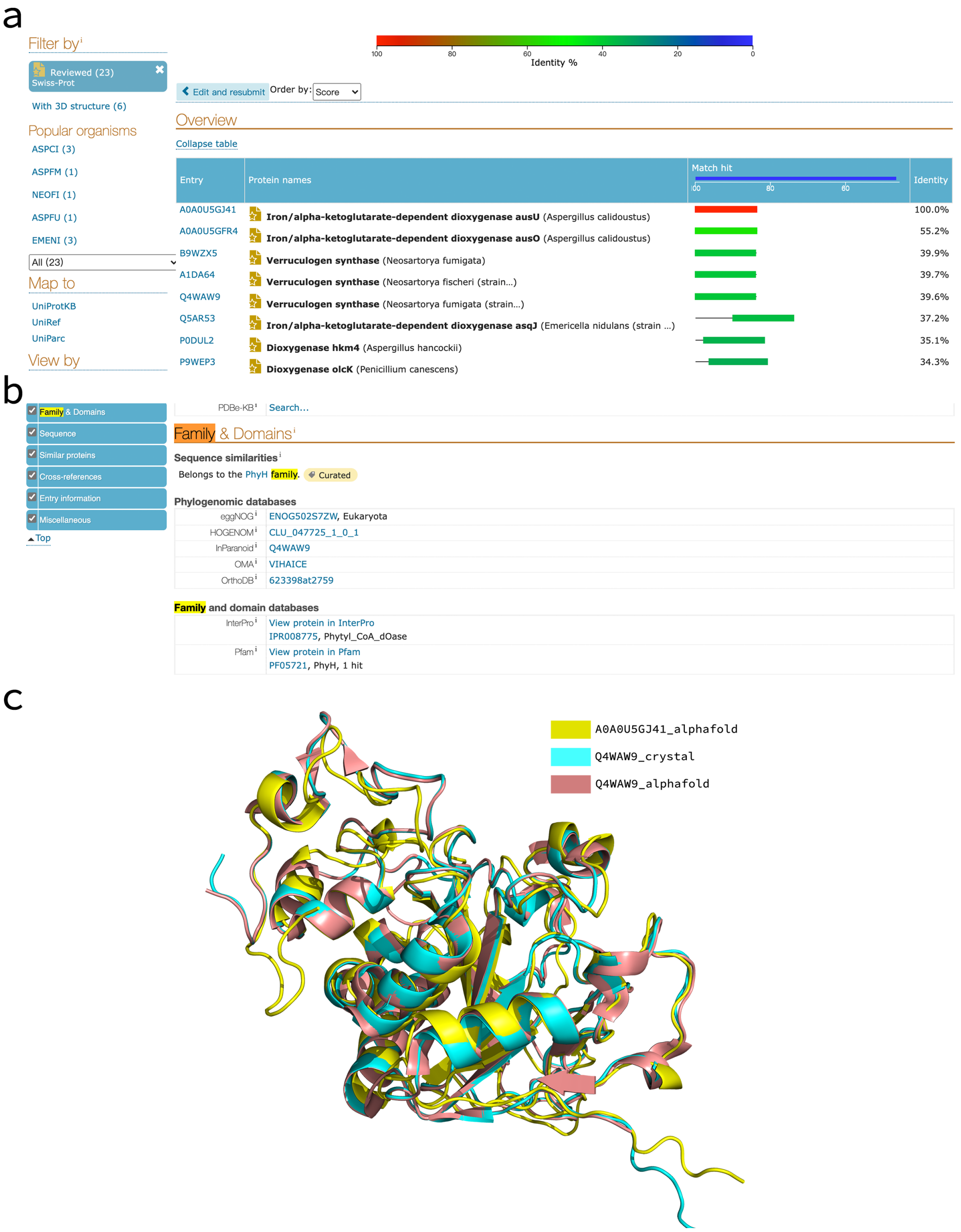
5 Case Study and Discussion

When dealing with the EC assignment problem in a daily production scenario, ECRECer offers two optional modes for end-users: a prediction mode and a recommendation mode. In prediction mode, we provide the results with the highest probability, while in the recommendation mode, we deliver up to 20 possible EC number annotations ranked by their respective likelihood. Here, we present an up-to-date EC number prediction case to simulate the real-time challenge by conducting EC number assignments in the prediction mode. We collected testing protein sequences from June 2020 to November 2021, encompassing 1968 records. These data were not employed in the development process of the existing methods or our proposed method, which is in line with the daily production scenario. In this evaluation, we compared our method with the state-of-the-art method DeepEC. The comparison results demonstrated the exceptional performance of our proposed method in terms of accuracy and F1 score. Specifically, our method successfully predicted 1739 records, while DeepEC correctly predicted 1025 only records (88.36 vs. 52.08%, 60.40 vs. 10.96% in terms of accuracy and F1, respectively, 992 identical prediction tasks). For the sake of comparison with methods not handling the 7th EC class (translocases), we reconstructed a reduced sub-dataset comprising 997 enzyme sequences labeled with EC codes from classes 1 to 6. In this scenario, our methods still deliver a better performance. ECRECer and DeepEC predicted 807 and 100 correct enzyme records, respectively (96 identical prediction tasks). This proved the outstanding performance of ECRECer in enzyme annotation.

In the databases, many enzymes with EC numbers exist in an uncompleted three-level, two-level, or even one-level state. However, these proteins with incomplete EC numbers might not directly be utilized for retrieving enzymatic reactions. For example, in the above case, we mispredicted 163 monofunctional enzymes, 67 of which had incomplete EC numbers, while ECRECer completed 38 records to the final fourth level. For instance, an enzyme iron/alpha-ketoglutarate-dependent dioxygenase AusU (UniProt ID: A0A0U5GJ41) has a two-level EC number in the database (1.14.-.-). When we used ECRECer for EC annotation, it assigned this protein with the fourth-level EC 1.14.11.38. This protein was recently integrated into UniProtKB/Swiss-Prot (September 29, 2021). After blasting it against the UniProtKB database, we found that the top 5 reviewed proteins with the highest identities include three verruculogen synthases (Fig. **??**a). We took protein Q4WAW9 as an example and found that both genes belong to exactly the same protein families with the same domains (Fig. **??**b). To further validate the results, we compared the structure of A0A0U5GJ41 (alphfold2 predicted) and Q4WAW9 (alphfold2 predicted and crystal structure). The results showed that these two proteins have a highly similar structure (SI, Figs. 3-6) with small RMSD (1.104). Therefore, the protein could be potentially annotated as EC 1.14.11.38 as well.

In addition to EC number assignment, another advantage of ECRECer is the recommendation of EC numbers, which makes our tool unique. Our method can give more possibilities via the recommended mode, which can facilitate the experimental scientists to explore the research. This is particularly useful in downstream applications, such as genome-scale metabolic network construction, where we have alternate options to do operations like gap filling if the predicted EC results are incorrect. An example is laccase (UniProt ID: A0A7T1FRB0, EC: 1.10.3.2), as its functional annotation is relatively sparse in the reviewed Swiss-Prot data. In this case, DeepEC, sequence alignment, and our prediction mode give out the prediction of a non-enzyme, but when we used the recommendation mode, we obtained a recommendation list [non-enzyme, 1.10.3.2, 1.-.-.-, 6.1.1.3, ...], and the correct annotation as offered as the second recommendation.

To demonstrate the inclusiveness and predictive ability of our proposed method, we conducted EC number prediction on an unreviewed protein family. *Corynebacterium glutamicum*, the famous industrial workhorse for amino acid production with a current output of over 6 million tons per year (Lee et al., 2016), is increasingly being adopted as a promising chassis for the biosynthesis of other compounds. However, unlike *E. coli* (1652 protein sequences with EC numbers out of 4322 proteins, 38.2%), the protein sequences of *Corynebacterium glutamicum* were not well annotated. Out of 3305 protein sequences, only 537 were reviewed and included in the Swiss-Prot database (357 proteins have assigned EC numbers). We used the other 2768 protein sequences to compare our tool with DeepEC. Our approach was able to assign 1056 proteins with EC numbers, while DeepEC only assigned 157 EC numbers (123 same EC numbers between DeepEC and ECRECer). Although there is no gold standard to decide which prediction is correct, we believe our algorithm should provide a more reasonable prediction as the proportion of protein sequences with EC numbers is similar to that of *E. coli* (42% vs. 15.5% in the case of DeepEC). The newly predicted EC numbers for the protein sequences are crucial for further analysis, such as retrieving metabolic reactions for genome-scale modeling.



Entry	Protein names	Identity
A0A0U5GJ41	**Iron/alpha-ketoglutarate-dependent dioxygenase ausU** (Aspergillus calidoustus)	100.0%
A0A0U5GFR4	**Iron/alpha-ketoglutarate-dependent dioxygenase ausO** (Aspergillus calidoustus)	55.2%
B9WZX5	**Verruculogen synthase** (Neosartorya fumigata)	39.9%
A1DA64	**Verruculogen synthase** (Neosartorya fischeri (strain...)	39.7%
Q4WAW9	**Verruculogen synthase** (Neosartorya fumigata (strain...)	39.6%
Q5AR53	**Iron/alpha-ketoglutarate-dependent dioxygenase asqJ** (Emericella nidulans (strain ...)	37.2%
P0DUL2	**Dioxygenase hkm4** (Aspergillus hancockii)	35.1%
P9WEP3	**Dioxygenase olcK** (Penicillium canescens)	34.3%

PDBe-KB Search...

Family & Domains

Sequence similarities

Belongs to the PhyH family. Curated

Phylogenomic databases

eggNOG	ENOG502S7ZW, Eukaryota
HOGENOM	CLU_047725_1_0_1
InParanoid	Q4WAW9
OMA	VIHAICE
OrthoDB	623398at2759

Family and domain databases

InterPro	View protein in InterPro IPR008775, Phytyl_CoA_dOase
Pfam	View protein in Pfam PF05721, PhyH, 1 hit

Figure 6: a) Blast search of the protein sequence against the UniProtKB database; b) Annotation of protein families and domains; c)Comparison of structural similarity.

6 Conclusion

In this work, we proposed a novel dual-core multiagent learning framework to complete three benchmarking tasks: 1) enzyme or non-enzyme annotation; 2) quantity of EC numbers ; and 3) EC number prediction. The method developed in this work has two calculation cores, an embedding core and a learning core. The embedding core is responsible for selecting the best available embedding method among one-hot, Unirep, and ESM to calculate sequence embeddings. The learning core is responsible for completing the specific benchmarking tasks using the best calculated protein sequence embedding as input.

We were guided by two principles in the design of our methods. The first principle is high usability (both can be accessed via the world-wide-web and provide standalone suit for high throughput prediction) with relatively balanced prediction performance (which can achieve the best accuracy with reasonable precision and recall). The second principle is providing comprehensive evaluation metrics with accessible reproduction datasets and source codes. To implement the first principle, we proposed DMLF. To implement the second principle, we provided a web server, standalone packages and opened all the source codes, including data preprocessing, dataset buildup, model training, and model testing/evaluation.

Experiments on real-world datasets and comprehensive comparisons with existing state-of-the-art methods demonstrated that our tool is highly competitive, has the best performance with high usability, and meets the proposed objectives. Although our tool exhibited the best performance, it still has space for improvement. For example, the performance of multifunctional enzyme annotation is relatively low, while the accuracy and recall of EC number annotation are less than 90%. Our feature work will focus on improving the prediction precision.

Key Points

- A multiagent dual-core learning framework is proposed to predict Enzyme Commission (EC) Numbers by using protein sequence data.
- A protein language model and an extreme multi-label classifier are adopted to reduce the heavy head-crafted feature engineering and elevate the prediction performance.
- The proposed framework remarkably outperforms the existing state-of-the-the-art method in terms of accuracy and F1 score by 70% and 20%, respectively.
- An online service and an offline bundle are provided for end-users to annotate EC numbers in high-throughput easily and efficiently.

7 Supplementary Data

Supplementary data are available onle at https://github.com/kingstdio/ECRECer/blob/main/document/supplementary.pdf

8 Data availability

The data underlying this article are available in the article and in its online Supplementary Material. The code of ECRECer, the training data, and the prediction results are available at https://ecrecer.biodesign.ac.cn/.

9 Author contributions statement

Z.S. and X.L. designed and implemented the model, conducted the experiments, analyzed the results and wrote the manuscript. H.M., Z.M. and Q.Y. reviewed the manuscript. R.W. and H.L. designed the website.

10 Funding

This work was supported by the National Key Research and Development Program of China [2018YFA0900300, 2020YFA0908300], the International Partnership Program of Chinese Academy of Sciences [153D31KYSB20170121], the Youth Innovation Promotion Association CAS, and Tianjin Synthetic Biotechnology Innovation Capacity Improvement Project [TSBICIP-PTJS-001, TSBICIP-CXRC-018].

References

[1] The UniProt Consortium. Uniprot: the universal protein knowledgebase in 2021. *Nucleic Acids Research*, 49(D1):D480–D489, 2021.

[2] Jae Yong Ryu, Hyun Uk Kim, and Sang Yup Lee. Deep learning enables high-quality and high-throughput prediction of enzyme commission numbers. *Proceedings of the National Academy of Sciences*, 116(28):13996–14001, 2019.

[3] Nicholas Furnham, John S Garavelli, Rolf Apweiler, and Janet M Thornton. Missing in action: enzyme functional annotations in biological databases. *Nature chemical biology*, 5(8):521–525, 2009.

[4] Wikimedia. Enzyme commission number. [EB/OL], 2021. https://en.wikipedia.org/wiki/Enzyme_Commission_number Accessed November 29, 2021.

[5] Jui-Hung Hung and Zhiping Weng. Sequence alignment and homology search with blast and clustalw. *Cold Spring Harbor Protocols*, 2016(11):pdb–prot093088, 2016.

[6] Chenggang Yu, Nela Zavaljevski, Valmik Desai, and Jaques Reifman. Genome-wide enzyme annotation with precision control: Catalytic families (catfam) databases. *Proteins: Structure, Function, and Bioinformatics*, 74(2):449–460, 2009.

[7] Clotilde Claudel-Renard, Claude Chevalet, Thomas Faraut, and Daniel Kahn. Enzyme-specific profiles for genome annotation: Priam. *Nucleic acids research*, 31(22):6633–6639, 2003.

[8] Nirvana Nursimulu, Leon L Xu, James D Wasmuth, Ivan Krukov, and John Parkinson. Improved enzyme annotation with ec-specific cutoffs using detect v2. *Bioinformatics*, 34(19):3393–3395, 2018.

[9] Chengxin Zhang, Peter L Freddolino, and Yang Zhang. Cofactor: improved protein function prediction by combining structure, sequence and protein–protein interaction information. *Nucleic acids research*, 45(W1):W291–W299, 2017.

[10] Ying Hong Li, Jing Yu Xu, Lin Tao, Xiao Feng Li, Shuang Li, Xian Zeng, Shang Ying Chen, Peng Zhang, Chu Qin, Cheng Zhang, et al. Svm-prot 2016: a web-server for machine learning prediction of protein functional families from sequence irrespective of similarity. *PloS one*, 11(8):e0155290, 2016.

[11] Alperen Dalkiran, Ahmet Sureyya Rifaioglu, Maria Jesus Martin, Rengul Cetin-Atalay, Volkan Atalay, and Tunca Doğan. Ecpred: a tool for the prediction of the enzymatic functions of protein sequences based on the ec nomenclature. *BMC bioinformatics*, 19(1):1–13, 2018.

[12] Adrian K Arakaki, Ying Huang, and Jeffrey Skolnick. Eficaz 2: enzyme function inference by a combined approach enhanced by machine learning. *BMC bioinformatics*, 10(1):1–15, 2009.

[13] Taofeek D Akinosho, Lukumon O Oyedele, Muhammad Bilal, Anuoluwapo O Ajayi, Manuel Davila Delgado, Olugbenga O Akinade, and Ashraf A Ahmed. Deep learning in the construction industry: A review of present status and future innovations. *Journal of Building Engineering*, page 101827, 2020.

[14] Haoyang Li, Shuye Tian, Yu Li, Qiming Fang, Renbo Tan, Yijie Pan, Chao Huang, Ying Xu, and Xin Gao. Modern deep learning in bioinformatics. *Journal of molecular cell biology*, 12(11):823–827, 2020.

[15] Yuxi Li, Yue Zuo, Houbing Song, and Zhihan Lv. Deep learning in security of internet of things. *IEEE Internet of Things Journal*, 2021.

[16] Zhenkun Shi, Sen Wang, Lin Yue, Lixin Pang, Xianglin Zuo, Wanli Zuo, and Xue Li. Deep dynamic imputation of clinical time series for mortality prediction. *Information Sciences*, 579:607–622, 2021.

[17] Hong-Bin Shen and Kuo-Chen Chou. Ezypred: a top–down approach for predicting enzyme functional classes and subclasses. *Biochemical and biophysical research communications*, 364(1):53–59, 2007.

[18] Yu Li, Sheng Wang, Ramzan Umarov, Bingqing Xie, Ming Fan, Lihua Li, and Xin Gao. Deepre: sequence-based enzyme ec number prediction by deep learning. *Bioinformatics*, 34(5):760–769, 2018.

[19] Ji Yong An, Yong Zhou, Yu Jun Zhao, and Zi Ji Yan. An Efficient Feature Extraction Technique Based on Local Coding PSSM and Multifeatures Fusion for Predicting Protein-Protein Interactions. *Evolutionary Bioinformatics*, 15, 2019.

[20] Kevin K. Yang, Zachary Wu, Claire N. Bedbrook, and Frances H. Arnold. Learned protein embeddings for machine learning. *Bioinformatics*, 34(15):2642–2648, 2018.

[21] Ethan C Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M Church. Unified rational protein engineering with sequence-based deep representation learning. *Nature methods*, 16(12):1315–1322, 2019.

[22] Roshan Rao, Joshua Meier, Tom Sercu, Sergey Ovchinnikov, and Alexander Rives. Transformer protein language models are unsupervised structure learners. In *International Conference on Learning Representations*, 2020.

[23] Marco Anteghini, Vitor Martins dos Santos, and Edoardo Saccenti. In-pero: Exploiting deep learning embeddings of protein sequences to predict the localisation of peroxisomal proteins. *International Journal of Molecular Sciences*, 22(12):6409, 2021.

[24] Hannah-Marie Martiny, Jose Juan Almagro Armenteros, Alexander Rosenberg Johansen, Jesper Salomon, and Henrik Nielsen. Deep protein representations enable recombinant protein expression prediction. *bioRxiv*, 2021.

[25] Hesham ElAbd, Yana Bromberg, Adrienne Hoarfrost, Tobias Lenz, Andre Franke, and Mareike Wendorff. Amino acid encoding for deep learning applications. *BMC bioinformatics*, 21(1):1–14, 2020.

[26] Shichao Zhang, Xuelong Li, Ming Zong, Xiaofeng Zhu, and Debo Cheng. Learning k for knn classification. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(3):1–19, 2017.

[27] Himanshu Jain, Venkatesh Balasubramanian, Bhanu Chunduri, and Manik Varma. Slice: Scalable linear extreme classifiers trained on 100 million labels for related searches. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 528–536, 2019.

[28] Ryan Rifkin and Aldebaro Klautau. In defense of one-vs-all classification. *The Journal of Machine Learning Research*, 5:101–141, 2004.