EC Number Prediction: Integrating Machine Learning and Sequence Analysis for Enhanced Enzyme Function Annotation

Member: Tatsan Kantasit, Stephen Chen, Ridhi Soni, Isaac Pfaender, Ethan Legson, Vidur Pitumbur

***Abstract*—Enzyme Commission (EC) number prediction is important in functional genomics, allowing the annotation of enzyme functions from protein sequences. In this study, we developed a machine learning pipeline that combines many diverse feature types like physicochemical attributes, predicted structural properties, evolutionary profiles and much more from protein sequence to classify enzymes by their EC numbers. We implement Random Forest classifiers to classify the EC numbers. Our approach utilizes the EC40 dataset (proteins filtered at 40% sequence identity) and is compared against DIAMOND-based sequence alignment. The results show that the machine learning models achieve very good accuracy. A feature importance analysis also reveals that among our 6 feature categories, the model embedding feature group performs the best. Notably, the best model outperforms DIAMOND in precision on low-similarity sequences, highlighting the advantage of feature-driven methods.**

I. Introduction

A. Background and Motivation

ENZYMES are catalysts in nearly all biochemical reactions, and understanding their functionality is fundamental in biology, medicine and biotechnology. The enzyme commission (EC) numbering scheme provides a hierarchical classification for these functions, going from broad class to specific substrates, which is also widely used for enzyme annotation. However, the availability of experimental results of enzyme function is far behind the growth in protein sequence databases in the current decade. As a result, computational methods for predicting EC numbers have become an interesting and crucial area of research that hopes to bridge the sequence-function annotation gap [1].

Traditionally, sequence alignment-based approaches like BLAST or DIAMOND [2], [3] have been the backbone for finding enzyme function. These methods assume that sequences have similar functions, often holding a high sequence identity. However, their reliance on alignments limits their prediction as the enzyme function does not always correlate linearly with sequence similarity. In some cases, enzymes sharing over 90% sequence identity have different functions, with the possibility to even differ at the first EC digit. In contrast, others with less than 30% identity perform the same specific function. Such examples highlight the need for more advanced methods that can capture functional determinants beyond simple similarity checks, such as structural or homology-based features.

Machine learning (ML) offers this alternative for EC number prediction by leveraging features gathered from the sequence and learning the complex patterns distinguishing enzyme classes. Unlike the alignment method, the ML model can utilise many diverse pieces of information and patterns that humans cannot see or notice and combine them to classify proteins, thereby potentially recognising the latent relationship that alignment-based methods may miss.

Recent developments in deep learning further showcase the power of sequence-based models, where the language model and transformer achieved state-of-the-art accuracy in enzyme function prediction without needing explicit sequence alignments. These advances motivate our exploration of classical machine learning algorithms with rich feature engineering through research of possible features that can be derived from a mere sequence to predict EC numbers. We conducted our experiment following the machine learning principle with careful data splitting, features selection, cascade training, and metrics such as accuracy, precision, recall, and f1-score to ensure consistency with many papers that also attempt this challenge.

II. Related Work

A. EC Prediction Techniques

Accurate prediction of enzyme functions using the EC numbering system is essential for functional annotation. Traditional methods like sequence alignment like BLAST or DIAMOND rely on identifying closely related homologues. However, these approaches have limitations, particularly for newer proteins with distant evolutionary relationships [1], [2]. Therefore, some papers have utilised machine learning techniques, including Support Vector Machines (SVM) and Random Forest classifiers, have emerged to address these limitations by exploiting diverse sequence-derived features to classify enzyme functions [4], [5], while some other papers focus more on features selection and highlight the usage of protein secondary structure such as position-specific scoring matrix (PSSM) [3].

B. Machine Learning and Feature Engineering in Protein Function Prediction

ML methods like SVM and Random Forest have demonstrated strong performance in predicting enzyme function.

With [6] use of SVM leverage the physicochemical and evolutionary features proves effective in classifying remote homologous enzymes. While deep learning methods have gained attention, classical ML methods remain valued for interpretability and computational efficiency.

Practical feature engineering enhances the accuracy of prediction in bioinformatics. Commonly used features include several categories, i.e., sequential (e.g., amino acid composition, motifs), physicochemical (e.g., hydrophobicity, charge distribution), structural (e.g., secondary structure, solvent accessibility), and evolutionary (e.g., position-specific scoring matrices) attributes. Sequential features capture direct sequence information and patterns; physicochemical features reflect the biochemical properties; structural features provide spatial context; and evolutionary features encode conservation patterns and functional significance. Despite extensive use, the study of feature importance across these types is limited, which also acts as one of the motivations for this research to understand their relative contribution to EC prediction.

III. Materials and Methods

We use the EC40 dataset, a curated collection of protein sequences fully annotated with EC numbers designed to benchmark enzyme function prediction methods. Each protein in this dataset is labelled according to its catalytic activity, allowing accurate performance evaluation across diverse enzyme classes. Additionally, to investigate the effects of sequence similarity on predictive performance, they constructed two further datasets of EC40 and EC50 based on similarity thresholds of 40% and 50%, respectively, but for the simplicity and aims of this experiment, we will only look at EC40. These custom datasets were taken from DEEPre and ECPred papers, which cluster enzymes and non-enzymes separately. Nevertheless, in our experiment, we remove non-enzymes to ensure our objective of predicting the EC number rather than predicting if it is an enzyme or not.

To acquire the dataset, we downloaded the ec40.pkl file provided by the authors and converted it to a CSV format using Python's `pickle` and `csv` libraries. The resulting dataset contains three columns: accession (a unique identifier assigned to each protein in sequence databases), sequence (the amino acid sequence), and the associated EC numbers. For example: `Q7VRM4`, `MQAKILRIATRKSPLAIC...`, `['2.5.1.61']`.

Some EC annotations may contain missing digits, such as `['2.3.1.-']`, `['3.1.-.-']`, or `['4.-.-.-']`, representing partially classified enzymes. In this experiment, we include such entries but still exclude any instances that are not enzymes altogether.

A. Feature Extraction

We have gathered more than 20 unique features for this experiment, with the majority coming from `peptides.py`, which computes common descriptors for protein sequences given the sequence from our CSV file and a small portion from `Bio.SeqUtils.ProtParam` module.

Below are 6 major feature groups that we have extracted from the sequences. These feature groups are also used as part of an ablation study. This involves the grouping of features based on their type (QSAR descriptors, sequential, physicochemical, PSSM, HMM, Modern Embedding Techniques) and training a Random Forest classifier on all feature groups. Subsequently, one feature group is dropped at a time and the model is retrained each time to assess the relative contributions from and importance of each group to the final prediction.

1) Sequential Features: Sequential features capture fundamental sequence characteristics, revealing important patterns linked to enzyme functionality. Amino acid composition measures the proportion of each amino acid present, reflecting general biochemical trends. The dipeptide composition evaluates the occurrence of adjacent amino acid pairs, maintaining local sequence information that is important for enzymatic specificity. Additionally, the statistical distribution of residues provides insights into sequence conservation and functionally significant motifs.

2) Physicochemical Features: Physicochemical features capture key biochemical and thermodynamic properties of protein sequences, crucial for understanding protein behavior and stability. Molecular weight and isoelectric point (pI) provide insights into size-dependent characteristics and charge distribution that affect the protein solubility and interaction dynamics. The grand average of hydropathy (GRAVY) quantifies overall hydrophobicity, while aromaticity measures the frequency of aromatic residues, which can impact structural stability and binding interactions. Instability index and aliphatic index assess protein resilience, with the instability index predicting in vitro stability and the aliphatic index indicating thermostability. Additionally, the Boman index estimates protein interaction potential, offering insights into binding affinity and bioactivity. Other descriptors in this class include Cruciani properties, which reflect electronic and hydrophobic interactions, and Z-scales, which encode hydrophobicity and steric characteristics through multidimensional representations.

3) Evolutionary and Homology-Based Features: Evolutionary and homology-based features integrate sequence similarity and conservation data, leveraging known functional annotations to infer enzyme characteristics. Position-specific scoring Matrices (PSSMs), generated by aligning sequences against a reference database with PSI-BLAST, were employed to encapsulate conservation patterns crucial for catalytic function prediction. BLOSUM indices, which measure evolutionary substitution tendencies between amino acids, are also included in this category due to their representation of conserved substitution patterns across homologous proteins. Additionally, homology-based annotations were derived by aligning sequences with well-characterized homologous enzymes, thereby transferring known functional annotations to query proteins.

4) Hidden Markov Models (HMM): HMM captures sequence profiles by modeling the probability of residue

transitions and emissions in protein families. These features help identify conserved domains and motifs that are important to enzymatic function.

5) QSAR Descriptors: Quantitative Structure-Activity Relationship (QSAR) descriptors and sequence profiles were extracted to capture intricate biochemical, physicochemical, and structural characteristics crucial for enzyme function prediction. QSAR descriptors numerically represent properties such as amino acid substitution patterns, physicochemical interactions, and molecular features, providing insights into subtle biochemical relationships that influence enzyme activity.

6) Modern Embedding Techniques: This study employed ESM-2[15], a pre-trained protein language model, for the purpose of sequence embedding. Protein language models are being increasingly used for sequence based prediction tasks and are, in this case, trained using masked language modelling (MLM)[15]- a technique where amino acids in the sequence are masked for the model to predict using surrounding context. The resulting model can then be used to extract an aggregated per-residue embedding, which is suitable for this task.

B. DIAMOND Benchmark

The primary experimental baseline for the project is the DIAMOND sequence alignment method. This is a popular choice of baseline due to its speed and sensitivity, offering 20,000 times the speed of tools like BLASTX for short reads while maintaining similar sensitivity levels [3] (Buchfink et al., 2014). Given the time constraints on this project, this was the most suitable choice. For this study, we selected the UniRef90 databases as the reference due to their clustered and non-redundant structure, which improves alignment relevance. Protein sequences from the EC40 dataset were processed and aligned against the UniRef90 database using DIAMOND, and the results were saved in standard .m8 format. The DIAMOND executable was downloaded from the official source, and the UniRef90 database was prepared locally by converting the FASTA file using the ‘makedb‘ command. The complete setup and availability of resources ensure reproducibility and scalability for this annotation task. This benchmark uses a nearest neighbour prediction approach, where the function of query protein is based on its most similar match, which we call the nearest neighbour in the database that we reference, typically determined by sequence alignment scores.

C. Machine Learning Models

To classify the EC numbers based on the inputs discussed, we used a random forest classifier as well as a host of models with a cascade training strategy. All models were trained to handle a multi-output problem, predicting each of the four digits that make up an EC number simultaneously.

Random Forest classifiers are made up of multiple decision trees combined, with their aggregate output making random forests a highly accurate predictive model and reducing variance compared to a single decision tree. The decision trees are built using a random subset of the data, ensuring improved generalisation and reduced overfitting [7]. The final prediction is done by majority voting amongst the different trees making up the ensemble.

They are capable of handling a large number of features even when the number of observations is small. Furthermore, random forest classifiers provide measures of variable importance, which is useful for feature selection and model interpretability as well as making them ideal for use in enzyme function prediction. This project utilised Grid Search to tune the parameters of single random forest model, where an ensemble of 200 trees with no maximum depth parameter pair is selected to ensure robustness in performance.

Model Training Process: All model is implemented mainly via `scikit-learn` (version *1.6.1*). `thundersvm` (version *0.3.4*) is deployed to speed training using gpu. We employ a cascade training strategy that takes predictions from each step as additional features for next steps in order to capture dependencies between the EC digits, helping the model better capture the hierarchical relationships between the different levels of enzyme classification and support multi-label classification task. The process consists of the following steps:

1) **Feature Consolidation and Data Preparation**: The training, validation and testing datasets are imported as pickle files, and features and EC labels are separated with the labels undergoing further encoding into four integer columns. The feature set is filtered using a previously selected set of indices and then normalised using MinMaxScaler.
2) **Data Splitting**: The training and validation sets are combined, and a PredefinedSplit is created to keep track of samples belonging to each category in order to perform hyperparameter tuning further along the pipeline.
3) **Feature Selection**: In order to enhance model performance and reduce computational complexity, a three-step feature selection process was implemented.
   - A variance filtering step was applied by removing features with low variability (threshold set at 0.001 according to the Fig. 1 variance distribution), effectively eliminating non-informative features. This step results in a subset of features that have sufficient variation across samples.
   - Multi-output feature selection was performed using mutual information to evaluate the relevance of each feature with respect to each output (i.e., each column in the multi-label target) and aggregate their importance. Mutual information scores were computed in chunks for each output, and the top 1000 features were selected based on their cumulative scores (see Fig. 2). This step yields a reduced set of features that are most informative for the multi-label task.
   - Finally, Recursive Feature Elimination with Cross-

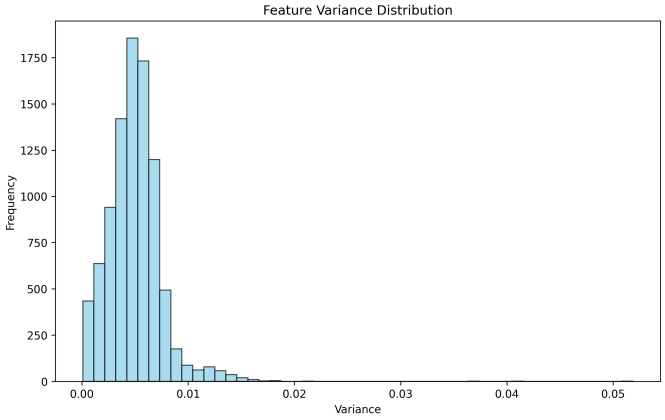


Fig. 1: Feature Variance Distribution

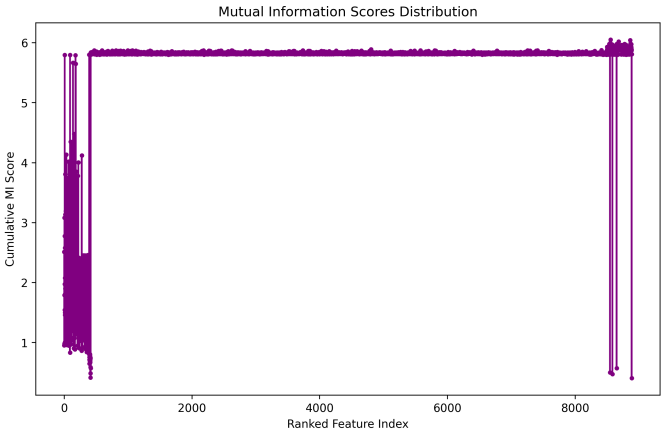


Fig. 2: Sorted Mutual Information Scores

Validation (RFECV) was employed to refine the subset of features further and find the optimal number of features. Then, using a RandomForestClassifier as the estimator produced a micro F1 score as the performance metric and a PredefinedSplit for training/validation separation. The elimination process proceeds in steps (removing 5 features per iteration) and with a minimum feature threshold of 100 until the optimal feature subset is identified, whose indices and ranking information were saved for further analysis. According to the iteration visualisation (Fig. 3), 100 became a most optimal choice. And worth noticing in the RFE tests, it appears that as the number of features increases, the average F1 score oscillates and declines. Because of the requirement to handle the positional encoding and inter-column correlations PSSM features/columns can't really be treated as features they way that other feature sets can. Raw PSSMs usually need some post-processing before being used as a feature set.

4) **Cascade Training**: For each EC digit, we do the following:
   - Each model is trained with 2 version of dataset, one with 100 selected features by feature selection part and another with whole feature set, aiming to verify the performance difference caused by feature dimension reduction.
   - The candidate model is trained on filtered features, ignoring all label-missing data at that EC position.
   - Once trained, the model predicts the given digit for the whole sample which is then appended to the feature set to train model next digit.
5) **Candidate Models**: The following candidate models were used in the cascade training process:
   - KNN
   - Logistic Regression
   - Random Forest
   - CatBoost Classifier (from *catboost version 1.2.6*)
   - SVM

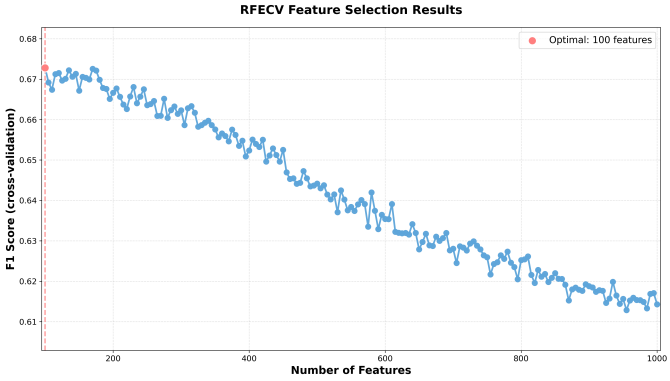


Fig. 3: RFECV Feature Selection

D. Baseline and Evaluation Metrics

We comprehensively evaluated our models using Exact Match Accuracy (EXAC), No Prediction (NP), and the position-wise metrics Accuracy (Acc), Precision (Prec), Recall (Rec), F1 Score (F1), Receiver Operating Characteristic (ROC) Curve and Area Under Curve (AUC)(for details on these metrics, see [9]). Our baseline comparison was conducted using the DIAMOND model.

E. Train-Test Split

- Since EC40 test set is seperated with 40% sequence identity. We just get all test data via existing 'traintest' column.
- Protein sequences in training set with same CD-HIT 40 cluster were randomly assigned exclusively to either the training or the validation set until size of validation set reach 10% of total training set.

IV. Results

A. Performance

Tables I and II report the classification performance across EC number positions 1 to 4. The DIAMOND benchmark shows very low EXAC (Exact match) and high NP (No Prediction) values of 0.0392 and 0.9410, respectively, indicating

TABLE I: Performance on Position 1 and 2

Method	EXAC	NP	Position 1				Position 2			
			Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
Diamond Benchmark	0.0392	0.9410	0.0557	0.7212	0.0375	0.0709	0.0533	0.4862	0.0318	0.0585
Selected Features + Single Random Forest	0.3767	0.0000	0.7821	0.7390	0.8146	0.7636	0.7059	0.4374	0.6677	0.5076
Selected Features + Cascade RandomForest	0.3798	0.0000	0.7741	0.7281	0.8058	0.7529	0.7027	0.4637	0.6848	0.5324
All Features + Single Random Forest	0.2010	0.0000	0.6970	0.6420	0.7350	0.6607	0.5701	0.2566	0.6312	0.3192
All Features + Cascade Random Forest	0.1728	0.0000	0.6560	0.5690	0.7667	0.5989	0.5496	0.2304	0.5842	0.2838

TABLE II: Performance on Position 3 and 4

Method	Position 3				Position 4			
	Acc	Prec	Rec	F1	Acc	Prec	Rec	F1
Diamond Benchmark	0.0495	0.5863	0.0392	0.0721	0.0392	0.2752	0.0440	0.0685
Selected Features + Single Random Forest	0.6751	0.3173	0.6473	0.4028	0.3423	0.5919	0.6292	0.5924
Selected Features + Cascade Random Forest	0.6995	0.3820	0.5995	0.4388	0.4407	0.3014	0.2755	0.2698
All Features + Single Random Forest	0.5783	0.1636	0.5380	0.2182	0.1627	0.7460	0.7700	0.7543
All Features + Cascade Random Forest	0.5727	0.1583	0.5241	0.2064	0.3347	0.1550	0.2730	0.7543

that it frequently failed to produce predictions. It also yielded low F1 scores and accuracy across all positions (e.g., F1 of 0.0709 and accuracy of 0.0557 at Position 1; F1 of 0.0585 and accuracy of 0.0533 at Position 2). The only notable metric where DIAMOND performed well was precision, achieving a high value of 0.7212 at Position 1, which remains relatively high across all positions.

In contrast, all the random forest models show no NP errors. The Selected Features + Cascade Random Forest performs the best on EXAC of 0.3798 and accuracy of 0.7741 on position 1; this excellent performance is carried throughout other metrics such as precision-recall and F1-score, which is similar to the single version where the EXAC is 0.3767 and accuracy on position 1 is 0.7821. The all features + single random forest performs worse with EXAC of 0.2010 and accuracy of 0.6970 for the first position, which is a considerable drop from the selected features model. However, it performs better than all features + cascade random forest in all metrics except for recall, where it's 0.7350 to 0.7667.

At position 2, the two selected features version still dominate; however, with a slightly lower performance, for example, the single version accuracy, precision, recall, and f1-score saw a huge drop by 0.3016, 0.1469, and 0.2560, respectively. As for all features + single random forest and cascade random forest, it gives an accuracy of 0.5701 and 0.5496 and exhibits the same huge performance drop in precision, recall and f1-score similar to the selected features version. The narrative still follows with position 3; however, the result closely resembles those of position 2, with selected features + cascade random forest accuracy dropping slightly to 0.6995 but still dominating overall feature versions. Despite this, all feature versions showcase an unexpected trend where precision-recall and f1 decrease and the accuracy surges up to 0.5783 and 0.5727 for single and cascade versions, respectively.

Finally, at the most challenging downstream level, Position 4, the accuracy drops for all methods, with the highest accuracy of 0.4407 achieved by the selected features + cascade random forest. Surprisingly, the selected features + single random forest model drops to an accuracy of 0.3423, which is very close to the accuracy of the all features + cascade random forest model at 0.3347. Its precision and recall are also very low, at 0.1550 and 0.2730, respectively. Despite this, the story is different for the all features + single random forest model, which exhibits a very high precision of 0.7460 and recall of 0.7700 but only achieves an accuracy of 0.1627.

TABLE III: Ablation study of Random Forest with different feature sets (averaged over all positions)

Model	Feature Set	Avg Accuracy	Avg Precision	Avg Recall	Avg F1-score	Accuracy Δ (%)
Single Random Forest	All Features	0.5020	0.4520	0.6686	0.4881	0.00%
	Without HMM	0.5026	0.4665	0.6831	0.5030	+0.12%
	Without PSSM	0.5700	0.4868	0.6753	0.5253	+13.53%
	Without Physiochemical	0.5005	0.4626	0.6759	0.4984	-0.30%
	Without QSAR Descriptors	0.5029	0.4539	0.6610	0.4895	+0.18%
	Without Embedding	0.3124	0.2904	0.4229	0.2889	-37.77%
	Without Sequential	0.5020	0.4511	0.6625	0.4877	0.00%

B. Ablation Study

Table III presents the ablation study evaluating the impact of different feature groups on the performance of a Random Forest model across all EC number positions. For this table, we also calculate the percentage change in accuracy. Using all features, the model got an accuracy of 0.5020 and an average F1-score of 0.4881. By removing HMM, Physiochemical, QSAR Descriptors, or Sequential features, the group have minimal changes of 0.12%, -0.30%, 0.18% and 0.00%, respectively. The most significant gain in accuracy is when we exclude

PSSM features, with the highest accuracy of 0.5700, which is a 13.53% increase. On the other hand, if we exclude Embedding, the accuracy decreases to 0.3124, which is a 37.77% decrease. This suggests that the embedding features group is the most important in correct prediction, while PSSM is the feature that correlates with the wrong prediction the most. Also, average recall remains relatively high across all ablations, suggesting robustness in capturing true positives despite feature removal.

C. ROC

For notations in the graph, models with names beginning with **all_** were trained on the full feature set, whereas those starting with **selected_** were trained on the feature set produced by our proposed three-step feature selection method. Based on the ROC curves in Fig. 4, where subfigure (a) corresponds to **EC_0**, (b) to **EC_1**, (c) to **EC_2**, and (d) to **EC_3**, we observed that models such as RandomForest consistently achieve high and stable AUC values across all EC digits, indicating robust performance even as the classification task becomes more specific. In contrast, while *KNN*, *LogisticRegression*, and the cascade-trained SVM on selected features (*selected_cascade_models_SVM*) can achieve AUC values above 0.9 in certain cases, their performance exhibits greater variability between different EC positions. Notably, the SVM model trained on the full feature set (*all_cascade_models_SVM*) performed worse than expected; its AUC scores for EC_1 and EC_4 were 0.23 and 0.43 lower than those of the single RandomForest model, respectively. Although SVMs typically excel at handling high-dimensional feature spaces, we did not perform targeted hyperparameter optimisation for the full feature set and instead reused the parameters from the selected-features version, which likely restricted its performance. Similarly, *CatBoost* shows relatively lower AUC values at some positions, suggesting that its adaptability across different EC numbers is less stable compared to other models. Overall, the RandomForest model demonstrates the most balanced and stable performance, achieving the highest AUC metrics overall, thereby underscoring its feasibility and robustness for enzyme function prediction.

V. Discussion

A. Interpretation of Results

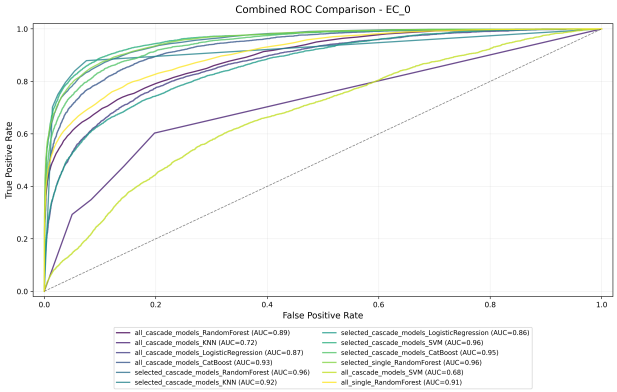
The random forest models demonstrate superior prediction compared to the DIAMOND benchmark, particularly highlighting the value of integrating machine learning with sequence-based features for EC number prediction. DIAMOND consistently failed to generate predictions with a high NP value of 0.9410 and showed below-par performance metrics aside from precision. This reinforces the limitation of sequence alignment-based methods in capturing the complexity of EC number prediction.

One notable observation from this experiment is that among all models, Random Forest consistently outperforms the others. In particular, reducing the feature set by employing selected features not only enhances the performance of certain models—as demonstrated by both single and cascade selected features model, which achieves an EXAC of 0.3767 and 0.3798, and also an accuracy of 0.7821 and 0.7741 respectively at position 1—but also substantially reduces training time and model complexity by eliminating noisy or redundant information, especially within the PSSM features group with over 100 sub-features, thus leading to more explicit decision boundaries. The reason behind the superior performance of selected features can be thought as that by selectively eliminating redundant or irrelevant features, the model can focus on the meaningful signals critical for accurately differentiating the enzymes. However, as the EC position becomes more specific (positions 2 to 4), all models, including the selected features version, show performance deterioration, reflecting the increasing biological specificity and complexity inherent in the EC numbering system.

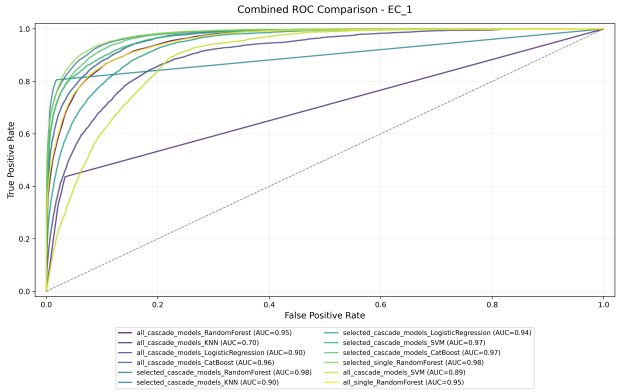
The massive decline in precision, recall, and F1 scores after Position 1 across all models can be seen as enzyme function annotation becoming more specific and subtle. Higher-level EC positions of 2, 3, and 4 differentiate enzymes based on fine-grained biochemicals, with one change in the sequence or specific residues could mean a whole new number on these lower levels. Furthermore, as we go down the EC number level, the number of classes increases, and the amount of training data per class decreases, leading to sparser and more imbalanced data. This makes accurate prediction more difficult. The decrease in model performance strongly suggests that the current feature set that we have, even after selection, struggles to adequately discriminate between closely related enzyme functions.

Interestingly, the performance difference between single and cascade random forests offers a fascinating insight. The cascade version tends to perform worse at the earlier levels but improves at the deeper levels. This is clearly seen in the all features version, where the single random forest achieves an accuracy of only 0.1627, while the cascade version reaches 0.3347. Despite this improvement in accuracy, the cascade model performs very poorly in terms of precision and recall. This suggests that while cascading models may capture general predictive trends more effectively at deeper levels, they often do so at the cost of precision—likely due to the propagation of earlier classification errors or the amplification of existing biases.

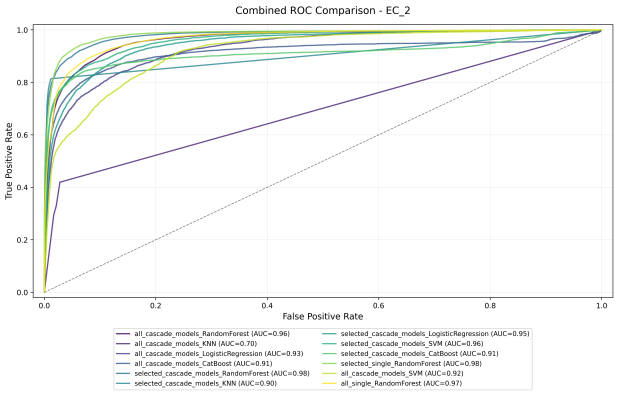
In biological contexts, even moderate success in predicting the initial positions of 1 and 2 dramatically benefits researchers by narrowing down enzyme classes and focusing experimental validations on smaller subsets of potential functions. However, the significant performance drop at positions 3 and 4 highlights an important area of study in this field and the need for further refinement of feature representation. Overall, the results presented support our hypothesis of machine learning approaches, especially with random forest classifiers coupled with strategic feature selection as a crucial tool in enzyme function annotation and even extend to other bioinformatics topics like protein prediction. The ablation study highlights the varying contributions of each feature group to the overall performance of the Random Forest classifier. Surprisingly, excluding PSSM features—typically considered informative for sequence-based tasks—resulted in the highest performance, indicating potential redundancy or overfitting when PSSM



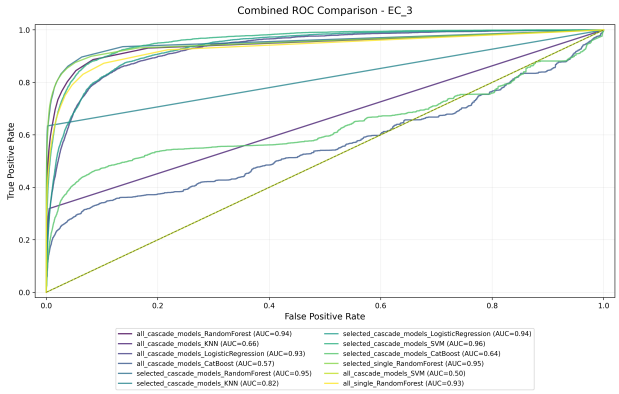
(a) EC_0 ROC



(b) EC_1 ROC



(c) EC_2 ROC



(d) EC_3 ROC

Fig. 4: ROC Curves for Four Positions

is combined with other feature types in this setup. This suggests that the PSSM features might not offer additional discriminative power beyond what is already captured by the remaining descriptors. Also, this may be due to their lack of true positional correspondence across sequence, making them less suitable for models that focus on statistics like SVM, though potentially better suited for neural networks. Furthermore, the unprocessed PSSMs give multiple uncorrelated noises to our feature set, with a given column in the PSSM unable to be correlated to a class that is predicted. The removal of sequential, HMM, physicochemical and QSAR descriptors leads to only minor changes, indicating that these features may play a supporting role rather than being critical drivers of performance. As for the main contributor, it is the modern embedding techniques because they effectively capture complex biological patterns and relationships through dense vector representations. These embeddings integrate contextual information and latent semantic properties of protein sequences, providing enhanced discriminative capabilities for EC number prediction tasks. Unlike traditional descriptors which often rely on simpler, handcrafted features, these embeddings can encode nuanced semantic information. Overall, the results imply that not all commonly used biological descriptors contribute equally to EC number prediction and that feature redundancy or interference should be carefully managed when designing input representations for enzyme function classifiers.

Finally, the ROC curve analysis shows that for the first EC digit, the curve is closer to the middle diagonal line and appears relatively smooth, indicating moderate classification confidence. However, as we move to the second, third, and fourth EC digits, the ROC curves become steeper and move further from the middle line, demonstrating an increase in classification difficulty as specificity increases. Overall, our results indicate that while our models significantly improve upon the DIAMOND benchmark, challenges remain in accurately classifying highly specific enzyme functions at deeper levels of the EC hierarchy. Future work could focus on refining feature extraction techniques and integrating deep learning methods to further enhance classification performance.

B. Comparison with Existing Methods

Our approach to predicts the EC number extracts multiple representations of characteristics, including sequential, physicochemical, Position-Specific Scoring Matrix (PSSM), Hidden Markov Model (HMM), and QSAR descriptor features. These extracted features are then used to train machine learning models such as Support Vector Machines (SVM) and Random Forest, employing a standard data-splitting approach for training and validation on EC40 dataset. A comparison with ECPred, a state-of-the-art enzymatic function prediction tool, highlights the differences and advantages of our method. ECPred employs an ensemble learning approach

TABLE IV: F1-scores Comparison for various EC classification levels.

Level	Classification Task	ECPred's F1-score	Our best model's F1-score
0	Enzyme vs. non-enzyme classification	0.83	N/A
1	Main EC class prediction	0.48	0.76
2	EC subclass classification	0.26	0.51
3	EC sub-subclass classification	0.22	0.40
4	Substrate-level EC classification	0.14	0.59

that integrates three classifiers: SPMap (subsequence-based classification), BLAST-kNN (homology-based classification), and Pepstats-SVM (physicochemical property-based classification). It achieves high classification performance across different EC levels, as shown in Table IV. These results indicate that while ECPred maintains strong performance in distinguishing enzymes from non-enzymes and in coarse-level function predictions, its accuracy decreases for more specific EC levels. In contrast, our approach integrates a wider set of feature types—including structural and physicochemical properties—which improves generalization, particularly for novel sequences without strong homologous matches. Compared to ECPred, our method provides a more feature-rich representation of protein sequences, allowing it to generalize better to unseen proteins.

C. Limitations

Despite the improvements demonstrated by our models over the DIAMOND benchmark, several limitations must be acknowledged. One of the primary challenges is the inherent bias in the dataset. Since the EC40 dataset is constructed with a sequence similarity threshold of 40%, it may still contain biases towards well-characterized enzyme families, leading to an overrepresentation of certain functional classes. This bias could result in inflated performance metrics that may not generalise well to completely novel enzymes. Also, given that the size of our dataset is very small due to time constraints on our end, we could use a bigger dataset with a more balanced group of position one, but the best approach would be constructing our own dataset where we have more control. Another limitation is the performance decline observed at deeper levels of EC classification. While our model performs well in distinguishing broad enzyme classes (e.g., EC1, EC2, and EC3), the accuracy significantly drops for subclass (EC2.X, EC3.X) and sub-subclass (EC2.X.X, EC3.X.X) levels. This drop is likely due not only to the increasing granularity of enzymatic functions, where feature differentiation becomes more challenging, but also to the fact that as the number of classes grows, each class receives less training information, making accurate prediction increasingly difficult. Finally, the use of traditional machine learning models such as SVM and Random Forest, while effective, may not fully exploit the deep hierarchical patterns present in enzymatic functions. Our expectation was that cascade models could capture the underlying correlations between EC numbers and that SVM would be able to handle the complicated dataset. However, it appears that a single multi-output Random Forest is performing a better job. A significant limitation in our predictor's performance is that we did not explore hyperparameter tuning for models other than Random Forest. Although Random Forests are somewhat robust to suboptimal parameter settings, hyperparameter tuning is essential for models such as SVM, which are highly sensitive to parameter selection. Due to time and computational resource constraints, we did not perform targeted feature elimination and hyperparameter tuning for models other than Random Forest Model, since they inherently do not support direct multiclass prediction and require tens of times longer training durations. More advanced deep learning architectures could potentially offer better generalisation and feature extraction capabilities.

D. Implications for Future Research

Future research could focus on addressing the dataset biases by incorporating additional enzyme sequences from diverse sources. One possible direction is to integrate protein function databases that include low-confidence annotations and use semi-supervised learning techniques to refine predictions for underrepresented enzyme classes. To improve performance at deeper EC levels, future models could explore hierarchical learning approaches that explicitly capture the dependency between enzyme classification levels. Techniques such as multi-task learning or hierarchical deep learning architectures could be used to improve subclass and sub-subclass predictions. Moreover, leveraging deep learning models such as convolutional neural networks (CNNs) or transformers could enhance prediction accuracy by automatically extracting relevant patterns from protein sequences without relying on predefined feature sets. Hybrid models that combine machine learning and deep learning approaches could be tested to optimize predictive performance.

VI. Conclusion

A. Summary of Findings

This study explored the effectiveness of machine learning models, particularly Support Vector Machines (SVM) and Random Forest, in predicting enzymatic functions based on a diverse set of protein descriptors. By integrating sequential, structural, physicochemical, PSSM, HMM, and QSAR features, our approach demonstrated a significant improvement over the DIAMOND benchmark. A key takeaway from our results is that feature diversity plays an important role in improving predictive accuracy, with specific feature sets contributing disproportionately to performance. From the six features group, the modern embedding features proved to be the most informative, consistently boosting accuracy across all levels. In contrast, PSSM features had a negative impact, showing a strong negative correlation with accuracy and contributing to a weaker performance compared to using all the features. Sequential features, on the other hand, had the least impact, showing minimal influence on the overall results. Our experiments also highlighted the challenges of enzymatic classification at deeper EC levels, where accuracy declines as specificity increases. The comparative analysis with existing methods underscored the advantage of our approach in generalization, especially for proteins lacking strong homologous matches. Additionally, the ROC analysis revealed a progressive change in classification difficulty across EC levels, reflecting the hierarchical complexity of enzymatic functions.

B. Future Work

1) Leveraging CATH Classification from the TED Database: We explored the potential use of the TED (The Encyclopedia of Domains) dataset due to its extensive structural annotations but faced challenges in direct integration. TED provides a comprehensive mapping of protein domains, classifying them based on structural and evolutionary similarities. Within TED, the CATH classification system is particularly valuable for function prediction, as it organizes protein domains hierarchically into Class, Architecture, Topology, and Homologous superfamilies, reflecting both structural and functional relationships.

However, due to the computational complexity and size of the TED dataset, direct integration posed challenges. Filtering relevant structural domains and efficiently embedding CATH-based features into machine learning pipelines remains an open problem. Future research should explore scalable approaches, such as selecting representative domains from CATH that are most informative for enzyme function, integrating precomputed embeddings, or using dimensionality reduction techniques to retain essential structural information while minimizing computational overhead. Additionally, coupling CATH classification with deep learning models could help refine structure-function predictions by learning from hierarchical domain similarities.

2) Enhancing Structural Feature Representation: Advancements in graph-based neural networks (GNNs) and geometric deep learning allow protein structures to be represented as spatial graphs. Using these techniques, we can better capture intricate structural relationships, potentially improving function prediction at deeper EC levels. Additionally, integrating structural constraints into machine learning models, such as attention mechanisms highlighting functional residues, could enhance classification accuracy.

3) Improving our Research and Model: Our current approach relies primarily on sequence-based features, which limits its effectiveness for proteins with low sequence similarity to known enzymes. Future improvements include integrating predicted domain architectures, which can provide more functionally relevant context beyond the primary sequence. Feature selection could also be enhanced by applying deep learning techniques, hence reducing dependence on manual feature engineering. Optimizing model hyperparameters could further improve generalization. Expanding the dataset with low-confidence annotations and semi-supervised learning techniques could help improve predictions for underrepresented enzyme classes. Finally, exploring hierarchical learning approaches, such as multi-task learning, could enhance classification at deeper EC levels. Hybrid models combining machine learning with deep learning techniques may also offer improved performance. Validating these refinements with independent test datasets from external sources will be crucial for assessing real-world applicability and benchmarking against existing state-of-the-art enzyme function prediction methods. Addressing these areas will refine our methodology, ensuring that protein function prediction models better leverage structural insights for more accurate and generalizable predictions.

C. Code Availability

The complete code and all results are fully reproducible. Please refer to the repository for details: https://github.com/ScooterStuff/ec-prediction-ml

References

[1] Pearson WR. An introduction to sequence similarity ("homology") searching. Curr Protoc Bioinformatics. 2013 Jun;Chapter 3:3.1.1-3.1.8. doi: 10.1002/0471250953.bi0301s42. PMID: 23749753; PMCID: PMC3820096.

[2] Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol. 1990 Oct 5;215(3):403-10. doi: 10.1016/S0022-2836(05)80360-2. PMID: 2231712.

[3] Buchfink, B., Reuter, K. Drost, HG. Sensitive protein alignments at tree-of-life scale using DIAMOND. Nat Methods 18, 366–368 (2021). https://doi.org/10.1038/s41592-021-01101-x

[4] A. Srivastava, A. Mahmood and R. Srivastava, "A Comparative Analysis of SVM Random Forest Methods for Protein Function Prediction," 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC), Mysore, India, 2017, pp. 1008-1010, doi: 10.1109/CTCEEC.2017.8455066.

[5] Baohui Lin, Xiaoling Luo, Yumeng Liu, Xiaopeng Jin, A comprehensive review and comparison of existing computational methods for protein function prediction, Briefings in Bioinformatics, Volume 25, Issue 4, July 2024, bbae289, https://doi.org/10.1093/bib/bbae289

[6] Tang ZQ, Lin HH, Zhang HL, Han LY, Chen X, Chen YZ. Prediction of functional class of proteins and peptides irrespective of sequence homology by support vector machines. Bioinform Biol Insights. 2009 Nov 24;1:19-47. doi: 10.4137/bbi.s315. PMID: 20066123; PMCID: PMC2789692.

[7] Ziegler, Andreas König, Inke. (2014). Mining data with random forests: Current options for real-world applications. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 4. 10.1002/widm.1114.

[8] Geethu S., Vimina E.R., Protein Secondary Structure Prediction Using Cascaded Feature Learning Model, Applied Soft Computing, Volume 140, 2023, 110242, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2023.110242.

[9] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks", Information Processing Management, vol. 45, no. 4, pp. 427–437, Jul. 2009, doi: https://doi.org/10.1016/j.ipm.2009.03.002.

[10] Dan Ofer, Michal Linial, ProFET: Feature engineering captures high-level protein functions, Bioinformatics, Volume 31, Issue 21, November 2015, Pages 3429–3436, https://doi.org/10.1093/bioinformatics/btv345

[11] Nicolas Buton, François Coste, Yann Le Cunff, Predicting enzymatic function of protein sequences with attention, Bioinformatics, Volume 39, Issue 10, October 2023, btad620, https://doi.org/10.1093/bioinformatics/btad620

[12] Nils Strodthoff, Patrick Wagner, Markus Wenzel, Wojciech Samek, UDSMProt: universal deep sequence models for protein classification, Bioinformatics, Volume 36, Issue 8, April 2020, Pages 2401–2409, https://doi.org/10.1093/bioinformatics/btaa003

[13] Yoshihiko Matsuta, Masahiro Ito, Yukako Tohsato, ECOH: An Enzyme Commission number predictor using mutual information and a support vector machine, Bioinformatics, Volume 29, Issue 3, February 2013, Pages 365–372, https://doi.org/10.1093/bioinformatics/bts700

[14] A. Dalkiran, A. S. Rifaioglu, M. J. Martin, R. Cetin-Atalay, V. Atalay, and T. Doğan, "ECPred: a tool for the prediction of the enzymatic functions of protein sequences based on the EC nomenclature," BMC Bioinformatics, vol. 19, Sep. 2018, doi: https://doi.org/10.1186/s12859-018-2368-y.

[15] L. C. Vieira, M. L. Handojo, and C. O. Wilke, "Scaling Down for Efficiency: Medium-Sized Transformer Models for Protein Sequence Transfer Learning," bioRxiv (Cold Spring Harbor Laboratory), Nov. 2024, doi: https://doi.org/10.1101/2024.11.22.624936.