Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Predicting Enzyme-Compound Interactions for Enzyme-Catalysed Reactions

Liam Brydon-Brown

lbry121@aucklanduni.ac.nz

University of Auckland

Gillian Dobbie

University of Auckland

Jörg Simon Wicker

University of Auckland

Katerina Taškova

University of Auckland

Research Article

Keywords: cheminformatics, biodegradation, hierarchical learning, enzyme prediction, product prediction

Posted Date: November 20th, 2025

DOI: https://doi.org/10.21203/rs.3.rs-8129973/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Predicting Enzyme-Compound Interactions for Enzyme-Catalysed Reactions

Liam Brydon-Brown[1,2*], Gillian Dobbie[1], Jörg Simon Wicker[1,2], Katerina Taškova[1]

[1]School of Computer Science, University of Auckland, Auckland, New Zealand.

[2]enviPath Limited, Auckland, New Zealand.

*Corresponding author(s). E-mail(s): lbry121@aucklanduni.ac.nz;

Abstract

Enzyme-catalysed reactions are common in many areas, including pharmaceutical metabolism and agricultural chemical biodegradation. Analysing and predicting how these reactions occur is increasingly important for identifying toxic by-products and achieving regulatory approval. Incorporating enzyme information into these predictions has been shown to improve prediction capabilities. However, existing methods require knowledge of the enzyme to perform prediction, and in many situations, especially biodegradation, the complexities of the reaction environment mean the exact enzymes are not known. In this paper, we alleviate this issue by proposing a framework to train and evaluate a hierarchical multi-label classifier to predict the enzyme commission numbers a chemical compound will react with. Our method achieves a hierarchical F1-score of up to 93.2%, outperforming existing methodologies. Additionally, we examine how including true and predicted enzyme information impacts product prediction performance compared to not using enzyme information. In our case study utilising biodegradation reaction data, we find that including enzyme commission numbers improve product prediction performance by approximately two percentage points.

Scientific Contribution We contribute a novel method for predicting enzyme-compound interactions using a hierarchical multi-label classifier framework. Our method is self tuning to find the best hyperparameters for a given dataset and achieves higher F1 scores than existing methods. We also contribute an investigation into including enzyme information into product prediction algorithms, showing that including this information can improve product prediction performance.

Keywords: cheminformatics, biodegradation, hierarchical learning, enzyme prediction, product prediction

1 Introduction

Predicting the products of chemical reactions is important for analysing how chemicals behave in the environment. For drug metabolism, performing these predictions allows chemists to potentially identify toxic metabolites before clinical testing [1]. Most recently, regulations have begun requiring biodegradation modelling to be submitted to obtain regulatory approval for any novel chemical [2]. Traditional lab experiments to analyse chemical behaviour are expensive and time-consuming. Subsequently, there has been a large research focus on developing machine learning methods to perform product prediction.

When dealing with chemicals in natural environments, it is extremely common for the reactions to be enzyme-catalysed, where an enzyme interacts with a compound to cause a change. The class of enzyme, defined by the Enzyme Commission (EC) number [3], plays a significant role in determining what reaction will occur. Existing product prediction methods largely ignore the enzyme class instead relying solely on the reactant chemical to perform prediction, commonly using sequence representations of compounds like the Simplified Molecular Input Line Entry System (SMILES) [4]. While some work has been done to include enzymes in product prediction inputs, it is limited to joining the EC number with the reactant SMILES and assumes the enzyme

class is known [5, 6]. For complex environments such as those in biodegradation, the enzyme classes are not always known, thus limiting the applicability of such methods. Recent works tackles this by taking a whole reaction containing the SMILES of both reactants and products and predicting the enzyme class [7, 8]. However, these methods create a circular dependency where predicting the product requires knowing the enzyme and predicting the enzyme requires knowing the product. This paper breaks the circular dependency by predicting the EC number(s) that a given compound is likely to interact with, rather than the EC number that caused a reaction. The EC number is represented by up to four numbers separated by dots, for example, 1.2.2.5. This is a hierarchical system where each position in the number is a subclass of the previous position. Subsequently, we propose the Enzyme Class Learning and Interaction Prediction SystEm (ECLIPSE), a framework for training and evaluating a hierarchical multi-label EC number classifier. ECLIPSE's primary function is to take in a compound and predict one or more EC numbers.

After developing and evaluating ECLIPSE, we test how including EC numbers influences product prediction performance. We do this with the Transformer architecture used by many existing works [9–14] and follow previous research in this area by concatenating the EC number to the reactant SMILES [7, 8]. In our product prediction analysis, we compare the performance of using the known EC numbers against no EC numbers and our predicted EC numbers. Comparing to our predicted EC numbers allows us to assess how the propagation of errors from ECLIPSE to the product prediction affects overall product prediction performance.

We summarise our contributions as follows:

- A hierarchical multi-label classification method for training and predicting enzyme-compound interaction likelihoods.
- An evaluation of how enzyme information impacts product prediction performance.

- A case study of enzyme and product prediction on small biodegradation datasets.

2 Related Work

In this section, we summarise previous works tackling similar problems. In Section 2.1, we look at reaction-level enzyme prediction methods, followed by existing work in hierarchical classification in Section 2.2 and how we can apply it to our problem. Lastly in Section 2.3, we summarise how enzyme information is incorporated to improve product prediction performance.

2.1 EC Number Prediction

Here we summarise previous EC number prediction research. We note that all these methods aim to predict EC numbers from a reaction. This differs to our goal of predicting EC numbers from a compound.

Qian et al. [7] looked at predicting the EC number of the enzyme that caused a given reaction. They use a BERT [15] style model that takes the reactants and products as inputs and performs classification to get the EC number. Their method achieves an F1-score of 91.3% on their task.

The Theia method published by Probst [16] provides a multilayer perceptron method for predicting the EC number of a reaction. They combine their predictions with DeepLIFT [17] to show which parts of the input molecule contributed to the predicted EC class, providing a high level of explainability for their predictions.

Another method by Zeng et al. [8] used a contrastive learning approach combined with reaction fingerprints to predict the EC number of a reaction. Contrastive learning aims to put data of the same class close together and different classes far apart in an embedding space. Prediction is then performed by finding the nearest known point to

the predicted point in the embedding space and assigning the known point's class to the prediction. Using this method, they achieve a weighted F1-score of 90%.

2.2 Hierarchical Classification

The hierarchical nature of EC numbers suggests that hierarchical classification methods are appropriate. Additionally, as multiple EC numbers can interact with a compound, a multi-label method would be most suitable. Such approaches have been applied in various domains, including the prediction of protein and gene function [18, 19].

There are three main ways hierarchical classifiers are built, Local Classifier per Level (LCL), Local Classifier per Parent Node (LCPN) and Local Classifier per Node (LCN) [20]. LCL trains one multi-class model per level to predict the class at that level, whereas LCPN trains a multi-class model per parent node to predict the child class. In contrast to LCL and LCPN, LCN trains a binary classifier on every node in the hierarchy. Subsequently, LCN is more naturally suited to a multi-label hierarchy as multiple classifiers at a level could return positive predictions, compared to the multi-class of LCL and LCPN, where the goal is generally to predict one class.

2.3 Product Prediction with Enzyme Information

Recent works have looked at incorporating enzyme information into reaction prediction for enzyme-catalysed reactions. These papers concatenate up to the third digit of the EC number to the end of the reactant, then use the commonly used transformer-based product prediction method to learn from this SMILES-EC input [5, 6]. They found that including the EC numbers improves reaction prediction performance.

3 Method

In this section, we first detail the datasets we use (Section 3.1) and how we preprocess them (Section 3.2). Following that, we give our method for performing EC number prediction, ECLIPSE, in Section 3.3 and how we incorporate the predicted enzymes into product prediction in Section 3.4. Lastly, Section 3.5 explains the metrics we use for evaluating ECLIPSE and the product prediction method.

3.1 Datasets

We use two main datasets in developing our method. These include ECMap [5, 6], and enviPath [21, 22] datasets.

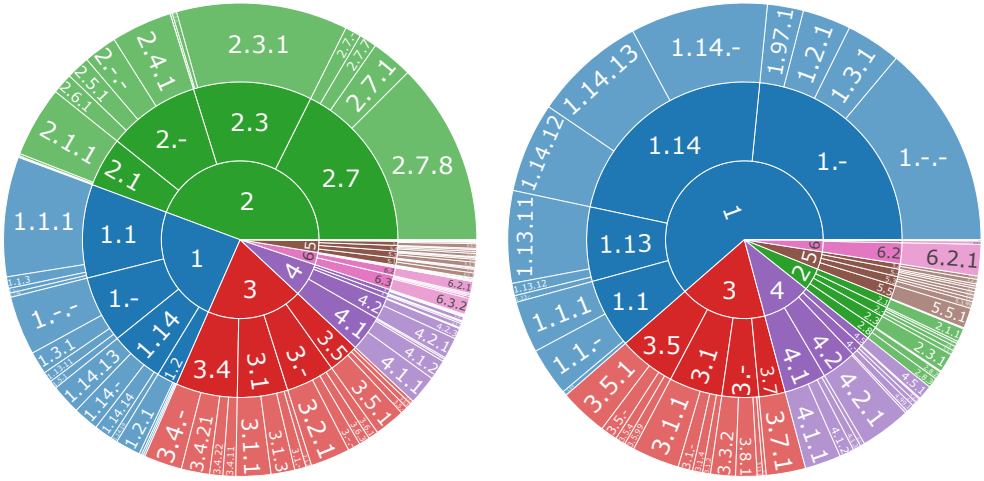
ECMap

ECMap contains $\sim 100,000$ enzyme catalysed reactions. We have created this dataset by combining the ECREACT dataset [6] and the EnzymeMap dataset [5]. Those datasets were originally extracted from sources such as BRENDA [23], KEGG [24], MetaNetX [25] and more. We removed any duplicate reactions by comparing reaction SMILES standardised with Rdkit [26]. This dataset gives the EC number alongside the reaction SMILES. Figure 1a shows that the distribution of the first EC number consists mainly of EC classes one, two and three, with significantly smaller amounts of four through seven. This is the primary dataset we use to train and evaluate our methods.

enviPath

EnviPath contains $\sim 4,400$ biodegradation reactions across three packages, Soil [27], Biocatalysis/Biodegradation Database (BBD) [28] and Sludge [29]. However, only the BBD package contains significant EC information with about 1,300 reactions containing ECs whereas Soil and Sludge contain approximately 10 reactions with EC

information. Figure 1b shows a very different distribution compared to ECMap, with EC classes one and three making up the vast majority of the dataset. As enviPath is substantially smaller than ECMap, we use it as a case study to see how smaller datasets change the performance of our EC number prediction method and the performance of incorporating the EC number into product prediction.



(a) ECMap EC number distribution (b) BBD's EC number distribution

Fig. 1: Distribution of EC numbers in ECMap (a) and enviPath's BBD (b). We show the top three most common classes at the second and third level and group the rest into the dash. The colour segments each of the seven level one EC numbers.

3.2 Preprocessing

To determine the number of classes for ECLIPSE, we use two parameters, *ec_level* and *min_samples*. EC level refers to how many digits of the EC number we consider at the start of our preprocessing. Minimum samples is the minimum instances an EC number must have to be used as a class. Using the whole EC number increases explainability by giving a precise enzyme class for the reaction. However, there is too little data per class on this level and thus insufficient data for a model to learn from. Conversely, only using one digit makes learning easier for a model, but it will give

a less precise overall class. To balance between the two, we use the method given in Algorithm 1. This method combines EC numbers into classes by iteratively grouping those with insufficient associated reactions by their parent class (i.e 3.2.1 and 3.2.2 could be grouped into 3.2.-). This occurs until each group has a size greater than the minimum samples parameter. Therefore, it is possible to have classes with different EC levels as some EC numbers have more data than others, however, no class will have more EC levels than the *ec_level* parameter. Any classes with less than *ec_level* have dashes appended until they reach the appropriate length. At the end of this process, any classes with insufficient data get removed. This only happens for classes with minimal data, such as EC number seven and its descendants.

Algorithm 1 Create the enzyme classes for ECLIPSE

Input: $min_samples > 0$
Input: $ec_level \in [1, 4]$
Input: *ec_groups* ▷ Enzymes grouped by their first *ec_level* digits
$ec_pos \leftarrow ec_level$
while any group has fewer than *min_samples* **do**
    for all $group \in ec_groups$ **do**
        if $group.size() < min_samples$ **and** $group.ec_digits() \geq ec_pos$ **then**
            $new_group \leftarrow$ first $ec_pos - 1$ digits of *group*
            Join *group* to *new_group* in *ec_groups*
            Delete *group* from *ec_groups*
        end if
    end for
    $ec_pos \leftarrow ec_pos - 1$
    if $ec_pos = 0$ **then**
        break
    end if
end while
for all $group \in ec_groups$ **do**
    if $group.size() < min_samples$ **then**
        Delete *group* from *ec_groups*
    end if
end for
$class_to_i \leftarrow$ each remaining group in *ec_groups* gets a unique index

3.3 EC Number Prediction

Our EC number predictor, ECLIPSE, aims to take a compound SMILES as the input and classify the EC number(s) with which the compound is most likely to interact. This model can be trained with different numbers of classes depending on the parameters given to Algorithm 1. Our method is also adaptive by utilising a tuning process to pick the best hyperparameters for the given dataset. We propose two versions of ECLIPSE, the Hierarchical ECLIPSE (H-ECLIPSE), described in Section 3.3.2 and the Flat ECLIPSE (F-ECLIPSE), described in Section 3.3.3.

3.3.1 SMILES Processing

For the F-ECLIPSE and H-ECLIPSE input, we convert molecule SMILES into molecular fingerprints that encode the structures of the molecule in a fixed-length vector. We opt to use the Morgan Fingerprint [30] implemented in rdkit [26] with the default values for radius (2) and fingerprint size (2048). We chose this as it is a commonly used fingerprint that provides a rich representation of the molecule's features. Additionally, research has shown that most molecule fingerprint methods are equivalent and the exact fingerprint is unlikely to change the method's performance [31]

3.3.2 Hierarchical Classifier

To develop our H-ECLIPSE, we use a hierarchical multi-label classification algorithm with LCN (as described in Section 2.2). We use the decision tree ensemble method XGBoost [32] for each node's classifier. As we use LCN, we need a binary training policy for each node. We use the *sibilings* policy for training the classifier on each node. This is where a subset of the whole dataset is used to train a given node. In Figure 2, the yellow outlined node is the node being trained. Only data from the classes associated with the given node's descendants, siblings, and siblings' descendants are used during training, these are all the outlined nodes. From this subset, the positive examples are

those belonging to the given node and its descendants, the green outlined nodes in Figure 2. The negative examples are those of the siblings and their descendants, the red outlined nodes in Figure 2. We chose this policy as it helps balance the subset of data used for training each node. Other policies use data from other branches in the hierarchy (the non-outlined in Figure 2), thus increasing the size of the negative class used to train each node. This disproportionately sized negative class would likely harm each classifier's performance.

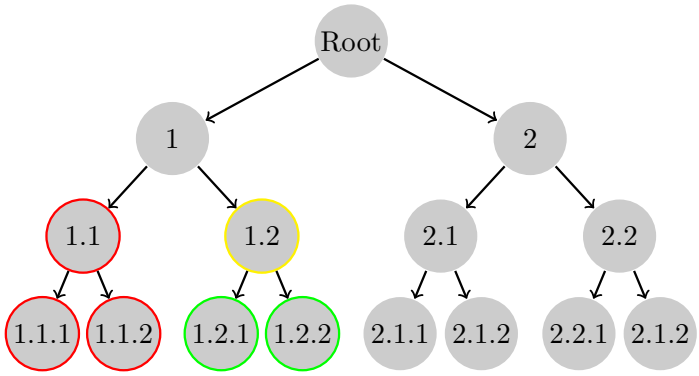


Fig. 2: The siblings binary-policy with the yellow outline node as the node being trained. Green nodes are positive examples, red nodes are negative examples and non-outlined nodes are not used for training.

When used for inference, all models at the first level are given the molecular fingerprint of the input SMILES. We then find the highest probability and subtract a *tolerance* value. All classes predicted with a probability greater than that are taken as the predicted classes at level one. For each predicted class at level one, the models associated with the children of the predicted level one classes are used with the same process. After performing this at each level in the hierarchy, the final prediction(s) are given. This method guarantees that at least one prediction will be made and allows us to tune the tolerance value (described in Section 3.3.4).

3.3.3 Flat Classifier

The F-ECLIPSE uses an Ensemble of Classifier Chains (ECC) with XGBoost as the individual classifier. Instead of treating the prediction as a hierarchy, the F-ECLIPSE 'flattens' the EC hierarchy into a standard multi-label classification problem. The F-ECLIPSE is trained with the same classes as the hierarchical classifier, extracted from Algorithm 1. This version of our method primarily serves as a simple baseline to compare to the H-ECLIPSE.

For inference, we perform one prediction with the F-ECLIPSE, giving probabilities for each class. We then use the tolerance to find the positive instances, classes with probabilities above $\max(probs) - \text{tolerance}$, the same process occurring at each H-ECLIPSE level.

3.3.4 Hyperparameter Tuning Process

The H-ECLIPSE and F-ECLIPSE utilise a tuning process to determine the best values for their respective hyperparameters. Both of them tune the number of estimators in XGBoost, the maximum depth of those estimators and the tolerance value. Additionally, F-ECLIPSE also tunes the number of chains in its ensemble. The value choices for these parameters are given in Table 1. We perform this tuning by creating a stratified validation split from the training data to estimate the performance of each configuration on unseen data. The tuning process utilises a simple grid search over all possible combinations of parameters. After each configuration is trained, we calculate performance metrics for each configuration using the validation data. The model whose parameters achieved the highest hierarchical F1-score (described in Section 3.5.1) is selected as the best model.

Method	Parameter	Choices
F-ECLIPSE & H-ECLIPSE	Num Estimators	{20, 50, 100, 200, 500}
	Max Depth	{3, 6, 9, 12, 15}
	Tolerance	{0, 0.01, 0.025, 0.1, 0.4}
F-ECLIPSE	Num Chains	{5, 10, 20, 40}

Table 1: ECLIPSE tuning parameters and their options.

3.4 Product Prediction

We pretrain an encoder-decoder transformer model [9] on the United States Patent and Trademark Office dataset for our product prediction method [33]. The United States Patent and Trademark Office dataset is a commonly used generic reaction dataset that contains approximately one million reactions. Following existing research, we treat the product prediction problem as a sequence-to-sequence translation task where the product SMILES must be predicted given the reactant SMILES [10, 34]. We preprocess all the reaction SMILES by first canonicalising them with rdkit [26]. The canonicalising ensures all SMILES are formatted the same. We also keep stereochemistry present in all our datasets. We separate all the reactions into reactants and products by splitting them on the ">>" token. We then tokenise each side of the reaction using the regular expression below. If either side of the reaction is longer than 380 tokens, we discard the reaction.

$$[|]|[A-Z][a-z]?|[A-Z]|[a-z]|:[0-9]3|:[0-9]2|:[0-9]|[0-9]|(|)|.|=|\#|-|+|\backslash|/|:|\sim|@@|@|?|>>|>|*|\$|\%$$

The parameters of our pretrained model are taken from our previous work [12] and are given in Table 2. We then refine the pretrained model with the ECMap data, concatenating the actual EC class number produced by Algorithm 1 to the reactant.

Parameter	Value
Encoder Layers	4
Decoder Layers	4
Heads	8
Embedding Dimension	256
Feedforward Dimension	2048
Dropout	0.1
Max Sequence Length	380

Table 2: Our pretrained product prediction transformer parameters

We only use the EC numbers predicted by ECLIPSE during inference. For training, we use the true EC numbers produced by Algorithm 1 to ensure the product prediction model gets the most accurate information.

3.5 Evaluation Metrics

We use different evaluation metrics for ECLIPSE and for the product prediction evaluation. These are detailed in Sections 3.5.1 and 3.5.2.

3.5.1 ECLIPSE Metrics

Since we treat the EC number prediction as a hierarchical multi-label classification problem, we use precision, recall and F1-scores specifically designed for hierarchical multi-label tasks [35]. The difference between these and their regular counterparts is the ability to be partially correct. If a prediction gets EC levels one and two correct but not three, it still contributes two-thirds to the relevant metric. The contribution is calculated by converting each hierarchical class to a set containing the label at each level and then checking the intersection size between the true and predicted sets. Equations 1, 2, and 3 formalise these calculations where C and C' are the true and predicted sets, respectively.

$$\text{Precision} = \frac{|C' \cap C|}{|C'|} \tag{1}$$

$$\text{Recall} = \frac{|C' \cap C|}{|C|} \tag{2}$$

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

In our multi-label scenario, the true or predicted set could contain numerous classes at each level. It is possible to weight the errors made at each level of the hierarchy differently, however, we opt to keep them equal.

In addition to the hierarchical measures, we also calculate precision, recall and F1-scores at each EC level. These per-level metrics differ from the hierarchical as they require the first 'X' EC digits to match and do not give partial points for correct parent classes. For example, we take only the first two digits of the true and predicted EC class when evaluating at EC level two. These are then compared against each other, checking for equality.

3.5.2 Product Prediction Metrics

We use both Top-K and precision-recall (PR) curves to evaluate the product prediction model. Top-K gives an overall accuracy of the K most likely sequences that the model gives, and is commonly used in the existing research. However, we showed in our previous work that Top-K has significant flaws [12]. Most importantly, it ignores the drop in precision associated with higher K values caused by the increased number of predictions. Therefore, it is akin to recall, not accuracy. Secondly, Top-K also ignores the specific probabilities given to sequences. By simply ranking them, it is assumed that all predictions at a given K have the same probability, this is likely not the case.

Therefore, we also use PR curves to evaluate the predictions described in our previous work [12]. When comparing to PR curves, we say one curve is better than another if it is more towards the top right corner of the plot (precision and recall of one), and also compare the peak precision and recall values of each curve.

3.6 Experiment Design

For all experiments using ECMap, we stratify our train/test splits using a multi-label stratification based on the level three enzyme classes associated with each reactant compound [36]. We use a 10-fold stratified cross-validation for all experiments and report the mean and standard deviations of all our metrics. Following our methodology in Section 3.3.4, the tuning process occurs ten times, potentially giving different configurations across the different cross-validation folds. When we do product prediction, we use the same 10-folds as the ECLIPSEs are trained on to ensure no advantage is given to the ECLIPSEs by providing them with data they saw during training.

We compare our method to an adapted version of BECPred [7]. We re-trained BECPred following the original paper, but modified the fine-tuning input with only compound SMILES instead of reaction SMILES to match our task.

Our method is implemented in Python 3.12 and uses a custom version of HiClass 4.13 [37] combined with XGBoost [32] for the hierarchical classification and PyTorch 2.7 [38] for the product prediction. We use the Adam optimiser with a learning rate of 5×10^{-4} for training the product prediction model. We also use early stopping with a patience of five and restore the weights to the best validation set accuracy when the early stopping triggers.

All the enviPath data is available on enviPath, https://envipath.org/. Our code is available here and the ECMap dataset is available here.

4 Results & Discussion

In this section, we give the results of our method evaluation. First, Section 4.1 discusses the tuning results from our tuning methodology. Sections 4.2 and 4.3 evaluate ECLIPSE with our metrics and the inference runtime, respectively. Then, Section 4.4 combines the EC number prediction with product prediction and evaluates using the product prediction metrics. Lastly, Section 4.5 gives the results from the case study with the enviPath datasets.

4.1 Tuning Results

In Table 3 we present the best model's parameter configuration for each cross-validation fold. The configurations for both models largely choose the top number of estimators at 500, especially with the H-ECLIPSE in Table 3, where only one fold chooses 200 instead. The maximum depth is more varied, with three and four different values achieving the best performance, in the respective methods. Across both methods, the tolerance is always at zero, meaning only one class will be predicted for any given input. This is likely because only a small portion of the dataset has more than one class, and therefore, predicting multiple classes often results in a large reduction in precision without a corresponding increase in recall, thus, the F1-score drops. For the F-ECLIPSE, the number of chains also varied with 10, 20 and 40 values, all performing best on different folds. Overall, we see a bias towards larger, more complex models with high numbers of estimators and high maximum depths.

4.2 ECLIPSE Evaluation

For ECLIPSE model, we give the mean and standard deviation of the precision, recall and F1-score at each EC level and the overall hierarchical version. In Table 4, we see that our proposed methods, F-ECLIPSE and H-ECLIPSE, outperform the baseline BECPred and that H-ECLIPSE has the highest mean across all metrics.

F-ECLIPSE					H-ECLIPSE		
Est	**Depth**	**Chains**	**Tol**	**Fold**	**Est**	**Depth**	**Tol**
500	9	10	0.000	0	500	15	0.000
500	12	40	0.000	1	200	12	0.000
500	6	20	0.000	2	500	12	0.000
200	12	40	0.000	3	500	12	0.000
500	15	40	0.000	4	500	15	0.000
500	12	40	0.000	5	500	9	0.000
200	9	10	0.000	6	500	15	0.000
200	12	20	0.000	7	500	12	0.000
500	15	40	0.000	8	500	15	0.000
500	12	10	0.000	9	500	15	0.000

Table 3: ECMap Best Parameters. F-ECLIPSE paramters are on the left of the fold number and H-ECLIPSE are on the right. Est is the number of estimators, Depth is the maximum depth and Tol is the tolerance. Chains is the number of chains for F-ECLIPSE.

Comparing F-ECLIPSE and H-ECLIPSE to BECPred, the hierarchical metrics show increases in the means of all three metrics, ranging between 2.4 and 2.8 percentage points. H-ECLIPSE and F-ECLIPSE perform very similarly with precision, recall, and F1, all sitting around 93.0%. Looking at the specific EC level metrics, F-ECLIPSE and H-ECLIPSE once again perform similarly to each other but surpass BECPred across all three metrics and levels. We also see that the metrics decrease at higher EC levels for all methods. BECPred sees a drop of almost 11 percentage points between levels one and three, whereas F-ECLIPSE and H-ECLIPSE only drop by four percentage points.

We group the predictions by their true EC level one to further analyse the methods' performance. Table 5 shows the hierarchical F1 for each EC. We give the full table with precision and recall in Table 1 of the supplementary material. F-ECLIPSE and H-ECLIPSE can outperform BECPred with increases in mean F1 of up to 4.3 percentage points on ECs one through four. ECs five, six and seven have higher standard deviations across all methods, and thus, there is large overlap between the standard deviations of the results. This is likely due to the small amount of data available for those classes. However, F-ECLIPSE and H-ECLIPSE still achieve higher mean F1 in

Level	Metric	BECPred	F-ECLIPSE	H-ECLIPSE
H	F1	90.49% ± 0.45	93.02% ± 0.16	**93.20% ± 0.20**
	Precision	90.49% ± 0.45	93.14% ± 0.30	**93.20% ± 0.20**
	Recall	90.49% ± 0.45	92.89% ± 0.50	**93.20% ± 0.20**
L1	F1	95.66% ± 0.22	96.27% ± 0.12	**96.50% ± 0.16**
	Precision	96.31% ± 0.22	96.92% ± 0.12	**97.15% ± 0.17**
	Recall	95.02% ± 0.23	95.62% ± 0.13	**95.85% ± 0.16**
L2	F1	89.77% ± 0.42	92.45% ± 0.21	**92.62% ± 0.26**
	Precision	91.26% ± 0.44	93.99% ± 0.16	**94.16% ± 0.22**
	Recall	88.32% ± 0.42	90.96% ± 0.27	**91.12% ± 0.30**
L3	F1	84.87% ± 0.57	89.07% ± 0.28	**89.12% ± 0.31**
	Precision	87.10% ± 0.62	91.40% ± 0.22	**91.46% ± 0.26**
	Recall	82.76% ± 0.56	86.84% ± 0.37	**86.89% ± 0.37**

Table 4: Performance of BECPred, F-ECLIPSE and H-ECLIPSE on the ECMap dataset. Results are split into hierarchical metrics and per EC level metrics. The highest mean for each metric is bolded.

these ECs with mean F1 increases of 3.9, 0.8 and 5.3 percentage points. Between F-ECLIPSE and H-ECLIPSE, the models perform very similarly, with often less than one percentage point difference between them. We see the largest gap between F-ECLIPSE and H-ECLIPSE on EC five, where H-ECLIPSE has a mean F1-score 1.7 percentage points higher. The performance on different ECs generally follows the proportion of each class present, with EC two performing the best and EC seven the worst (Figure 1a). EC six is an exception where, despite its small proportion in the dataset, all three methods achieve F1-scores over 90%. Overall, these results suggest that ECLIPSE methods can better learn the EC numbers related to a compound compared to BECPred.

4.3 Runtime

We benchmarked the inference runtime performance of all three methods across batch sizes 1, 10, 100, and 1000. In Figure 3 we see that H-ECLIPSE has a runtime over 16 times quicker than BECPred and F-ECLIPSE for batch sizes 1, 10 and 100. At a batch size of 1000, this drops to four times quicker. Due to its hierarchical structure, H-ECLIPSE only needs to use the positively predicted branch(es) classifiers rather

EC	BECPred	F-ECLIPSE	H-ECLIPSE
1	86.67% ± 0.91	90.25% ± 0.57	**91.02% ± 0.50**
2	96.22% ± 0.35	97.22% ± 0.20	**97.27% ± 0.22**
3	84.59% ± 0.67	**88.61% ± 0.45**	88.47% ± 0.44
4	76.40% ± 1.48	**80.51% ± 1.60**	80.01% ± 1.36
5	62.56% ± 4.02	64.70% ± 2.18	**66.42% ± 3.14**
6	90.91% ± 1.19	**91.70% ± 1.72**	91.50% ± 1.36
7	45.37% ± 8.85	50.13% ± 11.82	**50.70% ± 12.31**

Table 5: This table gives the hierarchical F1 grouped by EC level one for BECPred, F-ECLIPSE and H-ECLIPSE, on the ECMap dataset. The highest mean for each EC level is bolded.

than all its classifiers like F-ECLIPSE. Additionally, the ECC comprising F-ECLIPSE contains significantly more classifiers than H-ECLIPSE. Both these factors lead to H-ECLIPSE's highly efficient inference runtime. All three methods handle increasing batch sizes very well with minimal increase in inference runtimes. This suggests that these methods are highly parallelised. BECPred and F-ECLIPSE both see almost zero increase in runtime as the batch size increases.

4.4 Product Prediction Evaluation

Here, we compare the incorporation of known enzymes, predicted enzymes, and no enzymes into the standard product prediction transformer. We opt to use the H-ECLIPSE model for the predicted EC as it achieved the highest overall F1-score (Table 4). Table 6 shows that including the true enzyme class (True EC) increases the overall mean Top-1 by 1.2 percentage points compared to no EC number. There are similar increases across EC classes one through five. EC six sees a much smaller increase of less than 0.1 percentage points, and EC seven sees a 1.2 reduction in Top-1, however, the increase swings back in favour of True EC for Top-2 and Top-3.

Pred EC performs similarly to No EC in overall performance and EC's one through five. EC seven sees a drastic decrease in performance with the mean Top-1 dropping by nine percentage points. The performance of Pred EC on each EC class in product

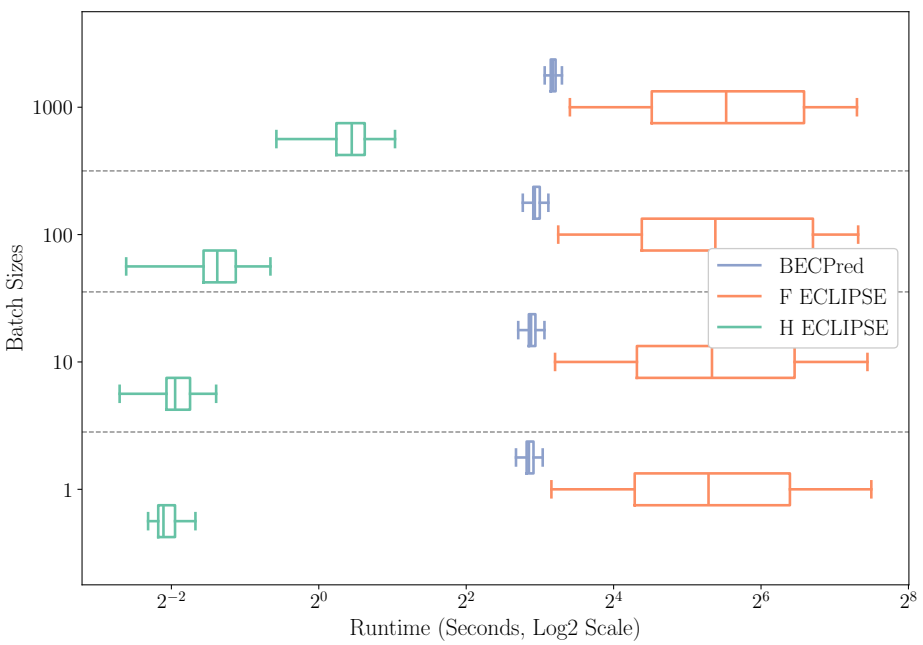


Fig. 3: Inference runtimes of BECPred, F-ECLIPSE and H-ECLIPSE on different batch sizes. Note the Log2 scale of the X-axis.

prediction is strongly linked to the H-ECLIPSE's ability to predict that EC class. EC two sees the best result for Pred EC with an increase of 0.8 percentage points over No EC. This corresponds to where H-ECLIPSE performs the best with an F1-score of 97.27%. Following this pattern, EC seven's drastic drop in performance of nine percentage points between Pred EC and No EC can be explained by H-ECLIPSE's F1-score of only 50.70% on EC seven.

The PR curves in Figure 4 show the overall product prediction performance and product prediction on EC two. All three methods perform very similarly in the overall plot (Figure 4a). The True EC and Pred EC achieve a slightly higher peak precision than No EC. True EC also achieves a higher peak recall than No EC. Looking at EC class two (Figure 4b), we see True EC and Pred EC outperform No EC with higher peak precision and curves consistently more towards the top right. True EC and Pred

EC	Top-K	True EC	Pred EC	No EC
Overall	Top-1	**59.66% ± 1.81**	58.09% ± 1.69	58.46% ± 2.32
	Top-2	**68.68% ± 1.69**	66.75% ± 1.57	67.85% ± 2.04
	Top-3	**72.05% ± 1.69**	69.93% ± 1.54	71.18% ± 1.96
1	Top-1	**49.32% ± 1.71**	47.48% ± 1.75	47.98% ± 2.15
	Top-2	**59.35% ± 2.14**	56.99% ± 2.00	58.12% ± 2.13
	Top-3	**63.82% ± 2.58**	61.18% ± 2.45	62.48% ± 2.15
2	Top-1	**68.32% ± 1.90**	67.85% ± 1.88	67.08% ± 2.74
	Top-2	**76.01% ± 1.37**	75.45% ± 1.34	75.51% ± 2.08
	Top-3	**78.53% ± 1.20**	77.88% ± 1.18	78.04% ± 1.90
3	Top-1	**57.75% ± 2.17**	54.67% ± 1.81	56.85% ± 1.99
	Top-2	**68.37% ± 2.11**	64.59% ± 1.83	67.82% ± 1.94
	Top-3	**72.07% ± 1.93**	68.09% ± 1.49	71.41% ± 1.74
4	Top-1	**51.20% ± 3.14**	48.80% ± 2.77	49.65% ± 3.33
	Top-2	**59.43% ± 3.35**	56.56% ± 2.99	57.47% ± 3.66
	Top-3	**63.33% ± 3.55**	60.11% ± 3.16	61.31% ± 3.51
5	Top-1	**24.53% ± 3.30**	22.66% ± 2.98	22.98% ± 3.95
	Top-2	**33.23% ± 3.36**	30.70% ± 3.01	31.46% ± 3.85
	Top-3	**36.93% ± 3.39**	34.24% ± 3.24	36.10% ± 3.78
6	Top-1	**57.04% ± 4.23**	55.05% ± 4.09	56.95% ± 4.83
	Top-2	**70.15% ± 3.82**	67.68% ± 4.03	69.84% ± 4.10
	Top-3	74.14% ± 3.59	71.28% ± 4.02	**74.21% ± 3.90**
7	Top-1	82.19% ± 11.96	74.40% ± 11.69	**83.41% ± 13.61**
	Top-2	**87.41% ± 8.20**	77.77% ± 10.01	86.93% ± 10.78
	Top-3	**88.99% ± 6.46**	80.29% ± 8.62	88.52% ± 8.37

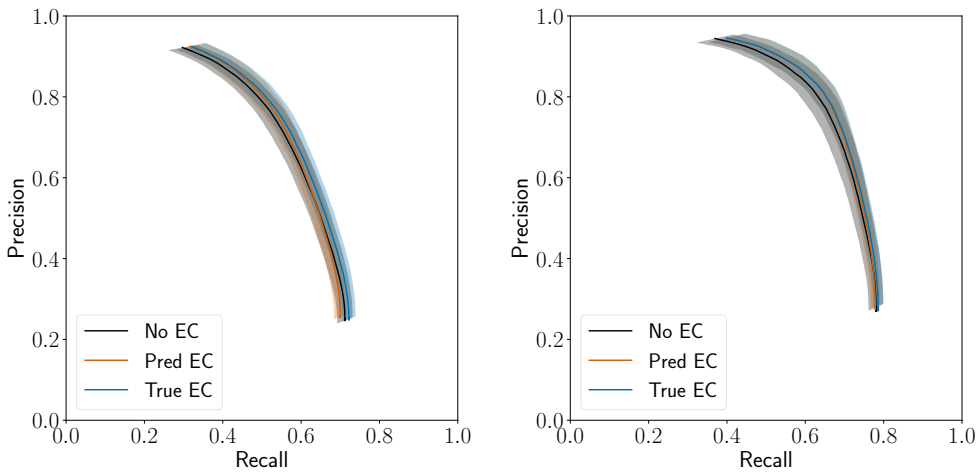
Table 6: Top-K accuracy comparison across True EC, Pred EC, and No EC models for the ECMap dataset.

EC have the most improvement in both figures over No EC in the middle of the PR curve.

Overall, including EC numbers in reaction prediction can slightly improve prediction performance. However, when using predicted EC numbers, it is clear that errors made by the ECLIPSE can significantly impact the product prediction results.

4.5 Case Study: enviPath

We use the enviPath dataset as a case study investigating how our method performs on a smaller dataset. We do this in two contexts, retraining with BBD and out-of-the-box inference with Soil and Sludge. We have to perform pure inference on Soil



(a) ECMap PR curve overall

(b) ECMap PR curve on EC two

Fig. 4: ECMap PR curves overall and on EC class two. Other PR curves by EC level one class are given in the supplementary material

and Sludge due to their highly limited amount of EC information. We split the BBD dataset into train and test sets using the same 10-fold cross-validation method we used with ECMap. We use only BBD for training as it is the sole enviPath package with EC numbers for the reactions. We again configure Algorithm 1 with a minimum of 40 samples and an EC level of three. We then retrain BECPred, F-ECLIPSE and H-ECLIPSE.

4.5.1 BBD Tuning Results

In Table 7 we give the parameters of the best F-ECLIPSE and H-ECLIPSE models on each cross-validation fold. Compared to the parameters for ECMap (Table 3), we see smaller numbers of estimators and smaller maximum depths being chosen. In the H-ECLIPSE parameters shown in Table 7, most estimators are either 20 or 50, and the max depth is mostly six or nine. F-ECLIPSE in Table 7 has generally higher estimator numbers, with a couple at 100 and 200, but has similar maximum depths. For the number of chains in the F-ECLIPSE, this is similar to the ECMap configurations, but

it does see three five-chain configurations appear. These lower parameter values lead to less complex models, perhaps avoiding over-fitting on the smaller dataset. Unlike the ECMap dataset, some models choose non-zero values for the tolerance. This suggests that more compounds in BBD have more than one EC number associated. Therefore, predicting multiple enzymes can lead to an increase in F1-score. However, even in these instances, the chosen tolerance value is small, with the most common value being 0.01. The features of these top models show that our tuning process is indeed adapting to the different datasets.

F-ECLIPSE					H-ECLIPSE		
Est	**Depth**	**Chains**	**Tol**	**Fold**	**Est**	**Depth**	**Tol**
500	6	20	0.010	0	20	15	0.000
200	6	5	0.010	1	100	9	0.010
200	15	10	0.100	2	20	6	0.025
500	3	5	0.000	3	50	9	0.000
100	6	5	0.000	4	500	9	0.000
20	15	5	0.000	5	50	15	0.000
50	9	10	0.100	6	200	6	0.000
50	15	10	0.000	7	50	15	0.000
50	3	40	0.010	8	50	6	0.025
100	9	20	0.000	9	20	6	0.000

Table 7: BBD Best Parameters. F-ECLIPSE paramters are on the left of the fold number and H-ECLIPSE are on the right. Est is the number of estimators, Depth is the maximum depth and Tol is the tolerance. Chains is the number of chains for F-ECLIPSE.

4.5.2 BBD ECLIPSE Evaluation

We performed the same evaluation of the models trained on BBD as we did on the ECMap dataset. Table 8 shows that F-ECLIPSE and H-ECLIPSE again outperform BECPred, with H-ECLIPSE performing the best.

H-ECLIPSE and F-ECLIPSE outperform BECPred with a mean F1-score, precision and recall of 7.5 percentage points higher in the hierarchical measures. H-ECLIPSE also achieves a slightly higher mean than F-ECLIPSE, approximately one percentage

point. We see results similar to those of the hierarchical level for the three EC levels. H-ECLIPSE achieves means of six, nine and six percentage points higher than BECPred for EC levels one, two and three, respectively. H-ECLIPSE is also consistently the top performing except for recall on level three, where F-ECLIPSE performs slightly better. These results show the same trend as the results on the ECMap dataset, with H-ECLIPSE performing the best and BECPred performing the worst.

Level	Metric	BECPred	F-ECLIPSE	H-ECLIPSE
H	F1	53.57% ± 4.03	60.02% ± 3.24	**61.09% ± 3.33**
	Precision	53.57% ± 4.03	59.98% ± 2.92	**61.62% ± 3.60**
	Recall	53.57% ± 4.03	60.08% ± 3.81	**60.59% ± 3.27**
L1	F1	69.70% ± 3.51	74.52% ± 4.13	**75.92% ± 2.80**
	Precision	71.51% ± 3.42	75.46% ± 4.21	**77.78% ± 2.81**
	Recall	68.00% ± 3.73	73.69% ± 4.85	**74.16% ± 3.00**
L2	F1	48.31% ± 3.99	55.87% ± 3.14	**57.45% ± 2.66**
	Precision	50.48% ± 3.98	57.05% ± 4.23	**59.91% ± 3.04**
	Recall	46.33% ± 4.07	55.03% ± 4.42	**55.20% ± 2.56**
L3	F1	44.63% ± 4.87	49.72% ± 3.64	**50.64% ± 3.55**
	Precision	47.22% ± 4.72	51.24% ± 5.18	**53.41% ± 3.67**
	Recall	42.34% ± 5.04	**48.62% ± 4.20**	48.18% ± 3.66

Table 8: Performance of BECPred, F-ECLIPSE and H-ECLIPSE on the BBD dataset. Results are split into hierarchical metrics and per EC level metrics. The highest mean for each metric is bolded.

Table 9 breaks down each method's hierarchical F1-score by EC level one. We give the full table with precision and recall in Table 2 of the supplementary material. The parameters used in Algorithm 1 meant that EC six and seven are removed as they contain too few instances. We see that F-ECLIPSE achieves the highest mean metrics on most ECs except EC one, where H-ECLIPSE achieves the highest. Grouping by EC greatly reduced the amount of data, leading to very high standard deviations across all methods and ECs. Despite that, we find that F-ECLIPSE and H-ECLIPSE outperform BECPred on every level one EC, with mean F1 scores of up to 27 percentage points higher in EC five. Between F-ECLIPSE and H-ECLIPSE, we see that H-ECLIPSE outperforms on EC by a mean of 3.2 percentage points. On all other ECs, F-ECLIPSE

outperforms H-ECLIPSE with a maximum increase of 4.9 percentage points on EC two and a minimum increase of 0.9 percentage points on EC four.

EC	BECPred	F-ECLIPSE	H-ECLIPSE
1	59.72% ± 5.25	63.74% ± 3.27	**66.94% ± 3.96**
2	29.22% ± 19.37	**40.08% ± 20.33**	35.18% ± 21.73
3	52.99% ± 4.29	**57.02% ± 12.24**	55.09% ± 12.04
4	33.71% ± 11.66	**47.30% ± 12.76**	46.41% ± 14.24
5	21.50% ± 7.43	**48.83% ± 21.90**	45.00% ± 19.89

Table 9: This table gives the hierarchical F1 grouped by EC level one for BECPred, F-ECLIPSE and H-ECLIPSE, on the BBD dataset. The highest mean for each EC level is bolded.

4.5.3 BBD Product Prediction

We perform production prediction on BBD using the same steps we used with ECMap in Section 4.4. Table 10 gives the overall and EC level one Top-K results. We see that True EC performs the best overall with a 2.3 percentage point gain over No EC. True EC also performs best on EC one, three and four. Looking at Pred EC, it outperforms No EC overall and on ECs three and four. Notably, Pred EC matches True EC's Top-1 performance on EC four. We see that No EC sometimes outperforms True EC, for example on EC two. However, for both True EC and No EC the product prediction performance is very low indicating that the transformer is has trouble learning this class regardless of EC information. This is likely due to the low amount of EC two data in BBD (see Figure 1b).

The BBD PR curves in Figure 5 again show True EC and Pred EC outperforming No EC. The overall PR curve in Figure 5a shows them both having better curves, achieving higher precision than No EC at all recall values. Interestingly, Pred EC maintains the same precision as True EC up to a recall of about 0.2 before its precision drops. Focussing on the EC three PR curve in Figure 5b, we again see True EC and Pred EC outperform No EC, with much higher precision across most of the curve.

EC	Top-K	True EC	Pred EC	No EC
Overall	Top-1	**20.70% ± 5.21**	18.41% ± 4.14	18.38% ± 4.06
	Top-2	**28.50% ± 6.18**	24.90% ± 5.26	24.01% ± 4.54
	Top-3	**30.77% ± 5.70**	26.66% ± 5.50	27.42% ± 4.22
1	Top-1	**23.53% ± 6.68**	20.35% ± 3.91	22.59% ± 4.96
	Top-2	**31.89% ± 7.81**	27.45% ± 6.25	28.00% ± 5.61
	Top-3	**34.30% ± 7.52**	29.14% ± 6.46	30.98% ± 5.79
2	Top-1	3.67% ± 7.77	3.67% ± 7.77	**4.43% ± 10.04**
	Top-2	5.10% ± 8.31	3.67% ± 7.77	**6.10% ± 10.60**
	Top-3	7.76% ± 10.84	6.33% ± 10.82	**9.19% ± 12.59**
3	Top-1	**20.32% ± 8.19**	19.18% ± 8.94	13.24% ± 6.53
	Top-2	**28.74% ± 9.14**	26.08% ± 8.60	20.81% ± 9.95
	Top-3	**30.32% ± 8.44**	27.67% ± 9.39	25.49% ± 9.73
4	Top-1	**9.77% ± 7.15**	**9.77% ± 7.15**	5.79% ± 6.69
	Top-2	**18.21% ± 11.98**	17.10% ± 10.89	10.88% ± 10.78
	Top-3	**19.75% ± 10.87**	17.87% ± 11.03	13.47% ± 9.46
5	Top-1	2.86% ± 6.02	1.43% ± 4.52	**5.93% ± 9.87**
	Top-2	**12.36% ± 16.35**	8.43% ± 11.27	8.43% ± 11.27
	Top-3	**16.36% ± 17.82**	8.43% ± 11.27	14.36% ± 15.89

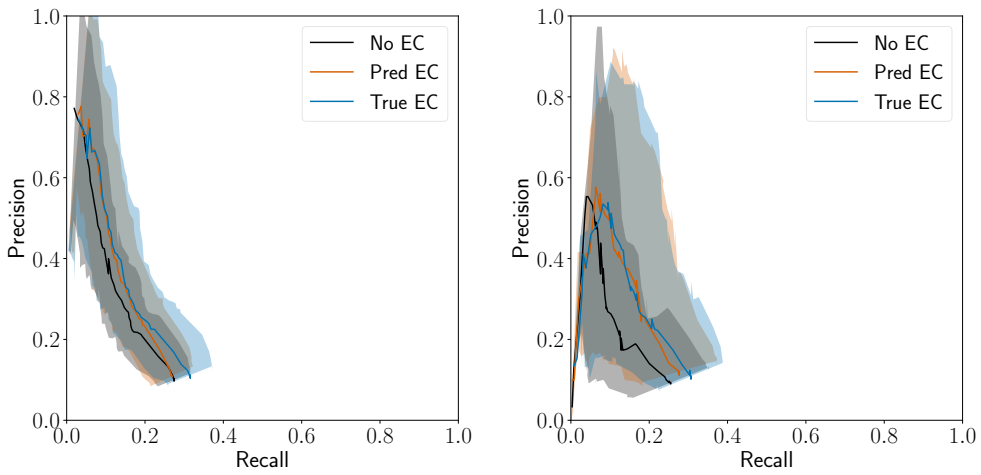
Table 10: Top-K accuracy comparison across True EC, Pred EC, and No EC models on the BBD dataset.

Overall, these results match those seen in the ECMap results. It is especially the case that the H-ECLIPSE performance greatly impacts product prediction performance with Pred EC's results on ECs two and five being particularly impacted.

4.5.4 Soil & Sludge Inference

For performing inference with Soil and Sludge, we first tune our H-ECLIPSE with the ECMap data following the tuning process in Section 3.3.4. Once the best parameters have been found, we retrain that configuration with all ECMap data, including the validation set usually left out in the tuning process.

Figure 6 shows the predicted EC classes for Soil (Figure 6a) and Sludge (6b). We see similar results for both datasets, with level one EC classes one and four being the majority of predictions. However, some differences exist, Soil sees EC two as the minority class, whereas EC three is the minority for Sludge. Interestingly, the model



(a) BBD PR curve overall

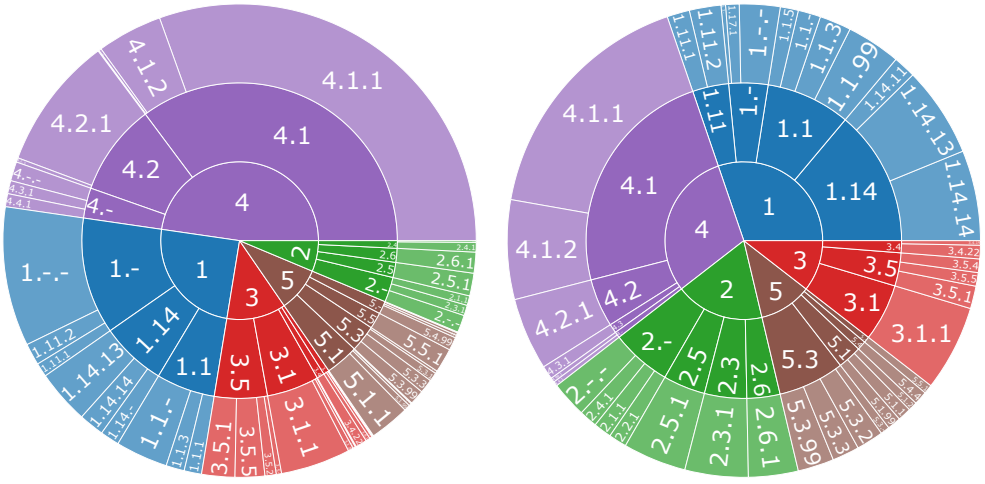
(b) BBD PR curve on EC three

Fig. 5: BBD PR curves overall and on EC class three. Other PR curves by EC level one class are given in the supplementary material

predicted no data in ECs six or seven. Soil and Sludge would likely have characteristics similar to BBD, which contains very few EC six and seven. Therefore, it seems reasonable that Soil and Sludge would have little or no compounds that interact with those EC classes.

In Figure 7 we show one example compound from Soil and Sludge with the true and predicted EC numbers. In the Soil example (Figure 7a) the ECLIPSE predicts two EC numbers 4.1.1 and 3.5.1 and the true EC is 3.5.1. For Sludge (Figure 7b) the ECLIPSE predicts only one EC number, 4.1.1 whereas the true set contains three EC numbers, 1.8.1, 6.4.1 and 4.1.1. In both these examples the predicted EC number is present in the true EC list.

After predicting EC numbers for Soil, we performed product prediction. We compare training our product prediction method with the predicted EC numbers and no EC numbers. Here, we have to train the product prediction model with the predicted EC numbers as no true EC numbers are available. We only do this on Soil, as Sludge



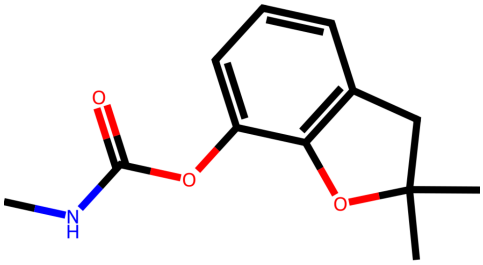
(a) Soil predicted EC class distribution (b) Sludge predicted EC class distribution

Fig. 6: Distribution of predicted EC numbers in Soil (a) and Sludge (b). We show the top three most common classes at the second and third level and group the rest into the dash. The colour segments each of the seven level one EC numbers.

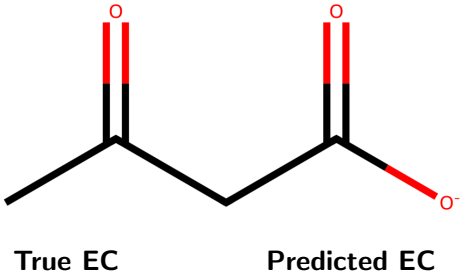
contains too few reactions. In Table 11, we see Pred EC outperforms No EC overall and all level one ECs by 1.5 percentage points in Top-1.

The PR curves in Figure 8 report similar results. Figure 8a shows that Pred EC outperforms No EC across the entire curve, with a higher peak precision of 0.7 compared to 0.55 and a higher peak recall of 0.55 compared to 0.45. Specifically looking at EC three (Figure 8b), Pred EC again achieves a higher precision of 0.4 compared to 0.3 and a higher recall of 0.5 compared to 0.4.

In the ECMap and BBD results, we noted that Pred EC performed best when the H-ECLIPSE performed best and could only outperform No EC in its best-performing ECs. Subsequently, seeing Pred EC consistently outperform No EC suggest that the EC classes predicted by the H-ECLIPSE for Soil are reasonable.



(a) Example compound in Soil alongside the True and Predicted EC classes.



(b) Example compound in Sludge alongside the True and Predicted EC classes.

Fig. 7: Example compounds from Soil (a) and Sludge (b) with the true and predicted EC numbers.

4.6 Results Summary

Summarising our results, we draw the following conclusions. Across the ECMap and BBD datasets, our H-ECLIPSE outperforms both F-ECLIPSE and BECPred. This shows that structuring EC number classification as a hierarchical classification problem is better than performing a 'flat' classification. Additionally, the performance over BECPred shows that tree-based methods like XGBoost can outperform the popular BERT-style methodologies.

Our product prediction results show that including enzyme information in the transformer model input can improve product prediction performance. This is especially the case on smaller datasets, as we see BBD and Soil gain the most from this additional information. These datasets are harder for the product prediction model to learn due to their limited size compared to larger ones, such as ECMap. Therefore, any relevant extra information, such as EC numbers, can be beneficial. We also note that when using predicted EC numbers, it is important to be aware of where the EC

EC	Top-K	Pred EC	No EC
Overall	Top-1	**34.30% ± 2.67**	32.79% ± 3.24
	Top-2	**46.65% ± 5.39**	43.23% ± 4.52
	Top-3	**51.52% ± 5.88**	48.62% ± 5.15
1	Top-1	**35.09% ± 4.07**	33.47% ± 5.41
	Top-2	**45.72% ± 5.81**	44.80% ± 3.93
	Top-3	**51.37% ± 5.88**	50.10% ± 3.23
2	Top-1	**25.38% ± 9.61**	23.78% ± 9.06
	Top-2	**39.81% ± 14.24**	38.97% ± 13.75
	Top-3	**45.32% ± 13.49**	42.88% ± 12.70
3	Top-1	**27.01% ± 8.50**	25.31% ± 10.56
	Top-2	**41.91% ± 15.59**	36.08% ± 15.41
	Top-3	**48.24% ± 16.74**	41.10% ± 17.23
4	Top-1	**38.65% ± 5.16**	37.42% ± 6.24
	Top-2	**51.92% ± 6.17**	45.34% ± 6.65
	Top-3	**55.47% ± 6.89**	52.23% ± 7.36
5	Top-1	**35.05% ± 13.48**	33.50% ± 12.77
	Top-2	**44.94% ± 13.70**	44.80% ± 12.30
	Top-3	**48.28% ± 12.93**	46.54% ± 14.48

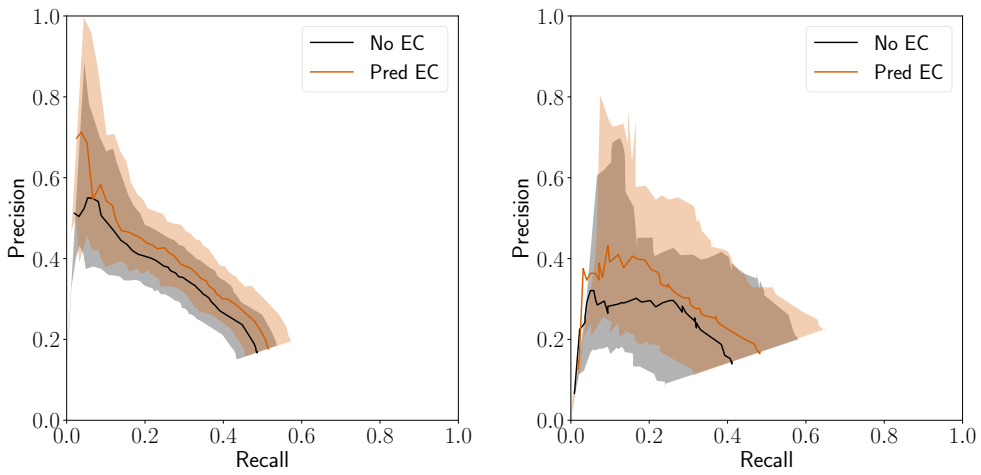
Table 11: Top-K accuracy comparison between Pred EC, and No EC models on the Soil dataset.

classification model performs poorly, as we find that when the predicted EC is wrong, the product prediction results are detrimentally impacted.

5 Limitations & Future Work

We only tune a small subset of hyperparameters, maximum depth, number of estimators, tolerance and number of chains (for F-ECLIPSE). This limitation is due to computational limitations preventing us from running the tens of thousands of combinations created by including other parameters. Future work could expand tuning to include more parameters and use a more intelligent tuning process that only focuses on the most promising hyperparameter combinations. Bayesian optimisation or genetic algorithms could be candidates.

Following the hyperparameters, we only used the Morgan fingerprint with the default values provided by RDKit. Future work could investigate whether other fingerprints,



(a) Soil PR curve overall

(b) Soil PR curve on EC three

Fig. 8: Soil PR curves overall and on EC class three. Other PR curves by EC level one class are given in the supplementary material

such as MACCS, or deep learning embeddings from methods such as ChemBerta [39], provide better information for the classifiers.

Concatenating EC numbers to the SMILES is a simplistic way to incorporate enzyme knowledge. Creating deep learning embeddings of the enzyme information to add to SMILES embeddings could be a better way of incorporating this information. Alternatively, using a mixture of experts' style method, where different product prediction models predict reactions caused by different ECs, could also improve overall product prediction performance.

6 Conclusion

We show that our proposed framework, ECLIPSE, achieves high statistical results and is highly computationally efficient at predicting EC classes that chemical compounds are likely to interact with. Our method outperforms the existing enzyme prediction method, BECPred, with a mean hierarchical F1-score of up to 93.2% on the ECMap

dataset. We also alleviate the reliance on product information to predict EC numbers for chemical reactions. Additionally, our H-ECLIPSE method is highly efficient with an inference time of less than $\frac{1}{16}$ compared to BECPred and our F-ECLIPSE. For product prediction, we find that incorporating enzyme information is beneficial to increasing product prediction performance, especially in smaller datasets such as enviPath's BBD and Soil, where we see the most improvement. Our research here aids the ongoing goal of reducing the environmental impact of chemicals by improving the knowledge and performance of chemical reaction prediction methods.

List of Abbreviations

EC - Enzyme Commission

SMILES - Simplified Molecular Input Line Entry System

ECLIPSE - Enzyme Class Learning and Interaction Prediction SystEm

LCL - Local Classifier per Level

LCPN - Local Classifier per Parent Node

LCN - Local Classifier per Node

BBD - Biocatalysis/Biodegradation Database

H-ECLIPSE - Hierarchical ECLIPSE

F-ECLIPSE - Flat ECLIPSE

ECC - Ensemble of Classifier Chains

PR - Precision-recall

7 Declarations

Availability of Data and Material

The data for reproducing the results is available at https://doi.org/10.5281/zenodo.17489235 and the code is available at https://github.com/MyCreativityOutlet/chem-eclipse.

Competing Interests

JSW is a co-founder and CEO of enviPath Limited, a scientific software development company that develops and maintains enviPath. LBB is an AI engineer and researcher at enviPath Limited. The authors declare no competing interests.

Funding

LBB is receiving the University of Auckland Doctoral Scholarship.

Author's Contributions

LBB proposed the initial methodology, carried out the experiments and wrote the initial manuscript draft. GD, KT and JSW continually provided expertise for refining the details during the entire research, as well as editing and improving the manuscript. All authors read and approved the final manuscript.

Acknowledgements

Thank you to Jasmin Hafner from the Swiss Federal Institute of Aquatic Science and Technology for her helpful discussion and finding the data used in Figure 7.

Thank you to the University of Auckland for the financial support through the University of Auckland Doctoral Scholarship.

References

[1] Testa, B., Pedretti, A., Vistoli, G.: Reactions and enzymes in the metabolism of drugs and other xenobiotics. Drug Discovery Today **17**(11), 549–560 (2012) https://doi.org/10.1016/j.drudis.2012.01.017 . Accessed 2025-02-10

[2] European Union: Regulation (EU) No 528/2012 of the European Parliament and of the Council of 22 May 2012 concerning the making available on the market and use of biocidal products Text with EEA relevance. Legislative Body: CONSIL, EP (2012). http://data.europa.eu/eli/reg/2012/528/oj/eng Accessed 2024-12-02

[3] International Union of Biochemistry and Molecular Biology. Nomenclature Committee: Enzyme Nomenclature 1992 : Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the Nomenclature and Classification of enzymes. Published for the International Union of Biochemistry and Molecular Biology by Academic Press, San Diego (1992). http://archive.org/details/enzymenomenclatu0000inte_d6c2 Accessed 2025-04-28

[4] Daylight Chemical Information Systems, Inc.: Daylight Theory: SMILES (2019). https://www.daylight.com/dayhtml/doc/theory/theory.smiles.html Accessed 2024-01-22

[5] Heid, E., Probst, D., H. Green, W., H. Madsen, G.K.: EnzymeMap: curation, validation and data-driven prediction of enzymatic reactions. Chemical Science **14**(48), 14229–14242 (2023) https://doi.org/10.1039/D3SC02048G . Publisher: Royal Society of Chemistry. Accessed 2025-02-10

[6] Probst, D., Manica, M., Nana Teukam, Y.G., Castrogiovanni, A., Paratore, F., Laino, T.: Biocatalysed synthesis planning using data-driven learning. Nature

Communications **13**(1), 964 (2022) https://doi.org/10.1038/s41467-022-28536-w . Publisher: Nature Publishing Group. Accessed 2025-03-18

[7] Qian, W., Wang, X., Kang, Y., Pan, P., Hou, T., Hsieh, C.-Y.: A general model for predicting enzyme functions based on enzymatic reactions. Journal of Cheminformatics **16**(1), 38 (2024) https://doi.org/10.1186/s13321-024-00827-y . Accessed 2025-03-18

[8] Zeng, Z., Guo, J., Jin, J., Luo, X.: CLAIRE: a contrastive learning-based predictor for EC number of chemical reactions. Journal of Cheminformatics **17**(1), 2 (2025) https://doi.org/10.1186/s13321-024-00944-8 . Accessed 2025-05-15

[9] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is All you Need. In: Advances in Neural Information Processing Systems, vol. 30 (2017). https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html Accessed 2023-04-28

[10] Irwin, R., Dimitriadis, S., He, J., Bjerrum, E.J.: Chemformer: a pre-trained transformer for computational chemistry. Machine Learning: Science and Technology **3**(1), 015022 (2022) https://doi.org/10.1088/2632-2153/ac3ffb . Publisher: IOP Publishing. Accessed 2023-05-28

[11] Schwaller, P., Laino, T., Gaudin, T., Bolgar, P., Hunter, C.A., Bekas, C., Lee, A.A.: Molecular Transformer: A Model for Uncertainty-Calibrated Chemical Reaction Prediction. ACS central science **5**(9), 1572–1583 (2019) https://doi.org/10.1021/acscentsci.9b00576

[12] Brydon, L., Zhang, K., Dobbie, G., Taškova, K., Wicker, J.S.: Predictive modeling

of biodegradation pathways using transformer architectures. Journal of Cheminformatics **17**(1), 1–16 (2025) https://doi.org/10.1186/s13321-025-00969-7 . Number: 1 Publisher: BioMed Central. Accessed 2025-02-19

[13] Wang, L., Zhang, C., Bai, R., Li, J., Duan, H.: Heck reaction prediction using a transformer model based on a transfer learning strategy. Chemical Communications **56**(65), 9368–9371 (2020) https://doi.org/10.1039/D0CC02657C . Publisher: The Royal Society of Chemistry. Accessed 2023-04-28

[14] Wu, Z., Cai, X., Zhang, C., Qiao, H., Wu, Y., Zhang, Y., Wang, X., Xie, H., Luo, F., Duan, H.: Self-Supervised Molecular Pretraining Strategy for Low-Resource Reaction Prediction Scenarios. Journal of Chemical Information and Modeling **62**(19), 4579–4590 (2022) https://doi.org/10.1021/acs.jcim.2c00588 . Publisher: American Chemical Society. Accessed 2024-01-08

[15] Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs] (2019). https://doi.org/10.48550/arXiv.1810.04805 . http://arxiv.org/abs/1810.04805 Accessed 2024-02-19

[16] Probst, D.: An explainability framework for deep learning on chemical reactions exemplified by enzyme-catalysed reaction classification. Journal of Cheminformatics **15**(1), 113 (2023) https://doi.org/10.1186/s13321-023-00784-y . Accessed 2025-10-19

[17] Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70. ICML'17, pp. 3145–3153, Sydney, NSW, Australia (2017)

[18] Valentini, G.: Hierarchical Ensemble Methods for Protein Function Prediction. International Scholarly Research Notices **2014**(1), 901419 (2014) https://doi.org/10.1155/2014/901419 . _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2014/901419. Accessed 2025-09-02

[19] Stojanova, D., Ceci, M., Malerba, D., Dzeroski, S.: Using PPI network autocorrelation in hierarchical multi-label classification trees for gene function prediction. Bmc Bioinformatics **14**, 285 (2013) https://doi.org/10.1186/1471-2105-14-285 . Num Pages: 18 Place: London Publisher: BMC Web of Science ID: WOS:000325434300001

[20] Silla, C.N., Freitas, A.A.: A survey of hierarchical classification across different application domains. Data Mining and Knowledge Discovery **22**(1), 31–72 (2011) https://doi.org/10.1007/s10618-010-0175-9 . Accessed 2025-09-02

[21] Wicker, J., Lorsbach, T., Gütlein, M., Schmid, E., Latino, D., Kramer, S., Fenner, K.: enviPath–The environmental contaminant biotransformation pathway resource. Nucleic Acids Research **44**(D1), 502–508 (2016) https://doi.org/10.1093/nar/gkv1229

[22] Hafner, J., Lorsbach, T., Schmidt, S., Brydon, L., Dost, K., Zhang, K., Fenner, K., Wicker, J.: Advancements in biotransformation pathway prediction: enhancements, datasets, and novel functionalities in enviPath. Journal of Cheminformatics **16**(1), 93 (2024) https://doi.org/10.1186/s13321-024-00881-6 . Accessed 2024-08-25

[23] Chang, A., Jeske, L., Ulbrich, S., Hofmann, J., Koblitz, J., Schomburg, I., Neumann-Schaal, M., Jahn, D., Schomburg, D.: BRENDA, the ELIXIR core data resource in 2021: new developments and updates. Nucleic Acids Research **49**(D1),

498–508 (2021) https://doi.org/10.1093/nar/gkaa1025 . Accessed 2024-02-29

[24] Kanehisa, M., Furumichi, M., Sato, Y., Matsuura, Y., Ishiguro-Watanabe, M.: KEGG: biological systems database as a model of the real world. Nucleic Acids Research **53**(D1), 672–677 (2025) https://doi.org/10.1093/nar/gkae909 . Accessed 2025-09-30

[25] Moretti, S., Tran, V., Mehl, F., Ibberson, M., Pagni, M.: MetaNetX/MNXref: unified namespace for metabolites and biochemical reactions in the context of metabolic models. Nucleic Acids Research **49**(D1), 570–574 (2021) https://doi.org/10.1093/nar/gkaa992 . Accessed 2025-09-30

[26] Landrum, G., Tosco, P., Kelley, B., Rodriguez, R., Cosgrove, D., Vianello, R., sriniker, Gedeck, P., Jones, G., Kawashima, E., NadineSchneider, Nealschneider, D., Dalke, A., Swain, M., Cole, B., Turk, S., Savelev, A., tadhurst-cdd, Vaucher, A., Wójcikowski, M., Take, I., Walker, R., Scalfani, V.F., Faara, H., Ujihara, K., Probst, D., Lehtivarjo, J., godin, g., Pahl, A., Maeder, N.: Rdkit: 2025_03_2 (Q1 2025) Release (2025). https://doi.org/10.5281/zenodo.15286010 . https://zenodo.org/records/15286010 Accessed 2025-04-29

[27] Latino, D.A.R.S., Wicker, J., Gütlein, M., Schmid, E., Kramer, S., Fenner, K.: Eawag-Soil in enviPath: a new resource for exploring regulatory pesticide soil biodegradation pathways and half-life data. Environmental Science. Processes & Impacts **19**(3), 449–464 (2017) https://doi.org/10.1039/c6em00697c

[28] Gao, J., Ellis, L.B.M., Wackett, L.P.: The University of Minnesota Biocatalysis/Biodegradation Database: improving public access. Nucleic Acids Research **38**(Database issue), 488–491 (2010) https://doi.org/10.1093/nar/gkp771

[29] Trostel, L., Coll, C., Fenner, K., Hafner, J.: Combining predictive and analytical methods to elucidate pharmaceutical biotransformation in activated sludge. Environmental Science: Processes & Impacts **25**(8), 1322–1336 (2023) https://doi.org/10.1039/D3EM00161J . Publisher: Royal Society of Chemistry. Accessed 2024-01-29

[30] Morgan, H.L.: The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. Journal of Chemical Documentation **5**(2), 107–113 (1965) https://doi.org/10.1021/c160017a018 . Publisher: American Chemical Society. Accessed 2025-09-02

[31] Stepišnik, T., Škrlj, B., Wicker, J., Kocev, D.: A comprehensive comparison of molecular feature representations for use in predictive modeling. Computers in Biology and Medicine **130**, 104197 (2021) https://doi.org/10.1016/j.compbiomed.2020.104197 . Accessed 2025-11-14

[32] Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery And Data Mining. KDD '16, pp. 785–794, New York, NY, USA (2016). https://doi.org/10.1145/2939672.2939785 . https://dl.acm.org/doi/10.1145/2939672.2939785 Accessed 2025-09-02

[33] Lowe, D.M.: Extraction of chemical structures and reactions from the literature. Doctor of Philosophy (PhD), University of Cambridge (October 2012). http://www.dspace.cam.ac.uk/handle/1810/244727 Accessed 2023-10-15

[34] Schwaller, P., Gaudin, T., Lányi, D., Bekas, C., Laino, T.: "Found in Translation": predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. Chemical Science **9**(28), 6091–6098 (2018) https://doi.org/10.1039/C8SC02339E . Publisher: The Royal Society of Chemistry.

Accessed 2023-08-01

[35] Kiritchenko, S., Matwin, S., Nock, R., Famili, A.F.: Learning and Evaluation in the Presence of Class Hierarchies: Application to Text Categorization. In: Lamontagne, L., Marchand, M. (eds.) Advances in Artificial Intelligence, Berlin, Heidelberg, pp. 395–406 (2006). https://doi.org/10.1007/11766247_34

[36] Sechidis, K., Tsoumakas, G., Vlahavas, I.: On the Stratification of Multi-label Data. In: Gunopulos, D., Hofmann, T., Malerba, D., Vazirgiannis, M. (eds.) Machine Learning and Knowledge Discovery in Databases, Berlin, Heidelberg, pp. 145–158 (2011). https://doi.org/10.1007/978-3-642-23808-6_10

[37] Miranda, F.M., Köhnecke, N., Renard, B.Y.: HiClass: a Python Library for Local Hierarchical Classification Compatible with Scikit-learn. Journal of Machine Learning Research **24**(29), 1–17 (2023). Accessed 2025-10-12

[38] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library. arXiv:1912.01703 [cs, stat] (2019). https://doi.org/10.48550/arXiv.1912.01703 . http://arxiv.org/abs/1912.01703 Accessed 2024-02-28

[39] Ahmad, W., Simon, E., Chithrananda, S., Grand, G., Ramsundar, B.: ChemBERTa-2: Towards Chemical Foundation Models. arXiv:2209.01712 (2022). https://doi.org/10.48550/arXiv.2209.01712 . http://arxiv.org/abs/2209.01712 Accessed 2024-10-09

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- chemeclipsesupp.pdf