## Methods

### Dataset Compilation

The dataset was compiled by manually collecting information about the primary sequence and substrate specificity of TEs from scientific literature. The final compiled dataset consisted of 106 TE sequences along with their corresponding substrate specificity. Based on their substrate specificity, the TEs were divided into three categories, 1) TEs with greater that 50% specificity for substrates above C12 and less that 10% specificity for C12 and below, 2) TEs with greater than 50% specificity for C12 or below and 3) TEs with mixed specificity. The first two categories accounted for about 48% and 35% of the training data while 17% of the training data belonged to the third category.

### Feature Extraction

In this work, the following three feature representation methods were employed to encode primary sequence information of the enzymes into fixed length vectors.

#### k-mer motif builder

The k-mer motif builder representation is similar to the N-gram representation for language models. A given sequence was scanned by a sliding window of a predefined length of characters, k, to generate all possible subsequences present within that sequence. The scanning procedure was carried out for all enzyme sequences in the training set to obtain a set of k-length subsequences or motifs that were present within these sequences. Only distinct subsequences which were present in at least two different enzymes in the training set were used to form an M-dimensional feature vector, where M represents the number of distinct motifs found in at least two different enzymes in the training set. For a specific sequence, the value of the feature vector will be 1 at the m-th coordinate () if the sequence contains the motif that corresponds to the m-th coordinate, 0 otherwise.

#### Grouped amino acid motif builder

The grouped amino acid motif builder performs an additional pre-processing step on the primary sequences of the enzymes before encoding them into a feature vector representation similar to the k-mer motif builder. In the preprocessing step, 20 amino acid types which can occur at a particular position in the sequence of an enzyme were categorized into five classes according to their physicochemical properties, hydrophobicity, charge and molecular size. The five classes were the aliphatic group (: GAVLMI), aromatic group (: FYW), positive charge group (: KRH), negative charged group (: DE) and uncharged group (: STCPNQ). Finally, each amino acid in the sequence was replaced by its respective class to create an encoded sequence with only five different types of values per position. For example, a primary sequence ‘MLTPWS’ is represented as . Using the encoded representation, the k-mer motif builder technique was used to construct a feature vector of length M, where M represents the number of distinct subsequences or motifs found in at least two different enzymes in the training set.

#### Positional feature builder

For positional feature builder method, the enzyme sequences were aligned using … . A total of 43 residue positions were selected from the aligned enzyme sequences based on their importance in prior acyl-ACP thioesterase mutagenesis studies as well as from structural analysis of the UcFatB binding pocket. Of these 43 positions, 15 were selected based on a previous study which successfully converted a long-chain acyl-ACP thioesterase from *Cuphea viscosissima* to have short-chain preference by interchanging the residues which varied in a sequence alignment. An additional 3 positions were selected based on an early mutagenesis study of the acyl-ACP thioesterase from *Umbellularia californica*, which converted the predominantly C12-specific thioesterase to a predominantly C14-specific thioesterase using sequence analysis of the homolog from *Cinnamomum camphora* to guide the design. Inspection of the *Umbellularia californica* structure, namely residues within the *α1* helix and the *β2*, *β4*, and *β5* helices, led to the identification of 5 additional positions. The remaining positions not encompassed within the criteria above were selected based on a sequence alignment among 6 thioesterases from 3 plant species. Each of the 3 plant species had a representative of a short-chain and a long-chain acyl-ACP thioesterase. The three species included were *Umbellularia californica* (accession numbers Q41635.1 and Q41634), *Cuphea palustris* (accession numbers Q39554 and Q39555), *Cuphea hookeriana* (accession numbers AAC49269.1 and AAC48990.1), and *Cuphea viscosissima* (accession numbers AEM72522.1 and AEM72523.1). After selecting the 43 positions, amino acids in those positions were represented as a one hot encoded feature vector. Each position apart from the 20 types of amino acids may also contain a gap (result of a multiple sequence alignment). Thus a 21-dimensional one hot encoded feature vector is used to represent each position.

### Support Vector Machine based learner of enzyme specificity prediction

The SVM based learner of enzyme specificity prediction included Principal Component Analysis for dimensionality reduction of the feature space followed by a Support Vector Classifier to predict enzyme specificity class. The PCA based dimensionality reduction step was carried out to decrease the number of parameters required to train an SVM model and make the model more generalizable. The number of PCA components, SVM model kernel, regularization parameter C and kernel coefficient gamma were selected by optimizing these hyperparameters using a 3-fold cross validation scheme described in the Model Training subsection.

### Neural Network based learner of enzyme specificity prediction

The NN based learner of enzyme specificity prediction included Principal Component Analysis for dimensionality reduction of the feature space followed by an Artificial Neural Network based classifier to predict enzyme specificity class. The PCA based dimensionality reduction step was carried out to decrease the number of parameters required to train an NN model and make the model more generalizable. The number of PCA components, hidden layer size of NN, initial learning rate and L2 regularization parameter alpha were selected by optimizing these hyperparameters using a 3-fold cross validation scheme described in the Model Training subsection.

### Ensemble Formulation

Our ensemble framework involved three base learners which provided an output to a meta learner that predicted the enzyme specificity class. Although all of our base learners were trained using the same principle (either SVM or NN), the heterogeneity among them was governed by the three different feature representation techniques described in the Feature Extraction section which were used to encode the set of enzyme sequences. The outputs of the base learners were passed on to the meta learner that used a majority voting scheme to predict the enzyme specificity category; if the predictions of the three base learners were respectively, then the prediction of the meta learner was . The workflow of our ensemble model is presented in Figure Y.

### Model Training

The model was trained using python’s numpy and scikit-learn modules. At first, a random seed was specified using numpy to reproduce results. The dataset of 106 TE enzyme sequences labeled according to their corresponding substrate specificity category was divided into a training set and a validation set by a 75-25 percentage split. The training set of sequences was encoded by the three different feature representation techniques, described in the Feature Extraction section, into three distinct feature vector representation of the sequences. The distinct feature vectors of the training set of sequences were used to train three separate base learners operating on the same principle (either PCA+SVM or PCA+NN). The hyperparameters of the base learners (number of components of PCA, kernel type, regularization parameter C, kernel coefficient in case of the SVM based learner or hidden layer size initial learning rate and regularization parameter in case of the NN based learner) were optimized using the GridSearchCV module of scikit-learn and setting the cross validation split to 3. The three base models with optimized set of hyperparameters were used to independently predict the substrate specificity category of enzymes in both training and validation sets. The output predictions of these base learners were passed on to a meta learner that used a hard-voting based majority vote classifier to output the final prediction of the enzyme substrate specificity class. In case there is a three-way tie between outputs of the base learners, the prediction of the positional feature builder based learners (which performed the best among all the base learners) was selected as the ensemble output. The training and validation accuracies of the three base learners and the ensemble model were recorded. This whole procedure was repeated 10,000 times by varying the random seed specified initially, which resulted in different training and cross validation set, thus affecting the model performance and yielding a distribution of training and validation set accuracies for the three base learners and the ensemble model. The objective of training our model multiple times was to check its robustness to the training set. In addition to the above mentioned hyperparameters, the k-mer motif and GAA encoded motif builder had an additional hyperparameter k that denotes the length of the motif to be considered. It was set to be 7 for both the models based on a separate validation study mentioned in detail in Appendix X.

### Automated Position Selection Method

The positional feature builder required expert level intervention and meticulous literature review to select the most relevant amino acid positions that affect the substrate specificity of TEs which can be time consuming and expensive. The automated position selection method tries to get rid of that limitation by automatically selecting the most important amino acid positions based on their statistical correlation with the labels (substrate specificity category of the enzymes). In this method, multiple sequence alignment of the enzymes was performed using the mafft tool. The aligned sequences were transformed into a one-hot encoded representation of each amino acid position. Theoretically, each position could be encoded by a 22 dimensional feature vector (20 types of amino acids, a gap and X, which represents an uncertain read) but in most cases a position was encoded by a vector of length 5 or even less since the types of amino acids in a specific position were usually conserved. The encoded feature space was used to select the best features which contributed the most towards the target variable or the labels variability. The feature selection technique was conducted using the SelectKBest module in scikit-learn and setting mutual information as the scoring function. Mutual information is selected as the correlation metric over Anova F-test or Chi-2 test because of its superior performance on capturing nonlinear interactions between dependent variables. The best ranked features were mapped back from the expanded one hot encoded feature space back to the original amino acid positions. These positions were recorded as the most important determinant of enzyme substrate specificity. The method had a parameter, n, that denotes the number of positions among total the aligned amino acid sequence positions that it must extract using the feature selection algorithm before stopping.