

# EnzymeHunter

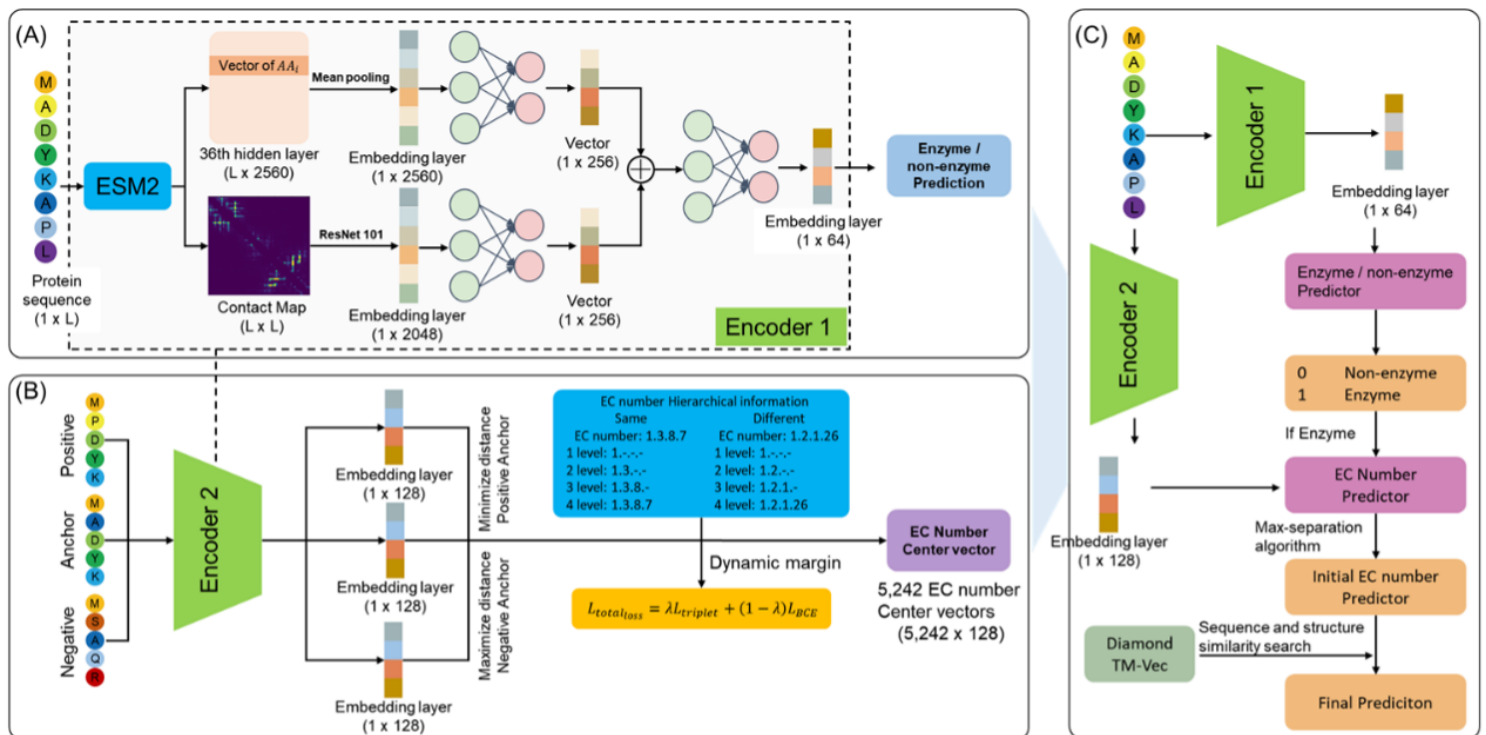
## Overview

**EnzymeHunter:** Achieving fine-grained enzyme function prediction with a hierarchically-aware contrastive learning framework.

This source code was developed and tested on **Linux (CentOS)** with **Python 3.9**.

## EnzymeHunter Architecture

Architecture and workflow of the EnzymeHunter model:



## Installation & Setup

### 1. Clone the repository

```
git clone https://github.com/cgxbio/EnzymeHunter.git
cd EnzymeHunter
```

### 2. Create and activate the virtual environment

```
conda env create -f environment.yml
conda activate EnzymeHunter
```

## Data and Model Download

Please download the pretrained data and model files from Zenodo:

👉 <https://zenodo.org/records/17048211>

After downloading, place the contents into the following directories:

```
EnzymeHunter/
├── data/          # Place dataset files here
├── model/         # Place model files here
├── EnzymeHunter.py
└── ...
```

## Usage Example

Here are two common examples of using the **EnzymeHunter** , corresponding to different running modes:

### 1. Normal Mode (Default Mode)

Use this mode when you need the program to automatically distinguish between enzymes and non-enzymes, and only predict EC numbers for proteins classified as enzymes:

```
# Example: run prediction on a dataset where not all proteins are enzymes
python EnzymeHunter.py --dataset example_test --all_are_enzymes False
```

Explanation:

- --example\_test is your test dataset name (without the .fasta extension)
- --The program will first perform enzyme/non-enzyme classification prediction
- --Only proteins predicted as enzymes (pred\_label=1) will undergo subsequent EC number prediction
- --Non-enzyme proteins will be marked with EC number 0.0.0.0

### 2. All-Enzymes Mode

Use this mode when you know all proteins in the dataset are enzymes and want to skip the enzyme/non-enzyme classification step, proceeding directly to EC number prediction:

```
# Example: run prediction assuming all proteins are enzymes
python EnzymeHunter.py --dataset example_test --all_are_enzymes True
```

Explanation:

- --example\_test is your enzyme dataset name (without the .fasta extension)
- --The --all\_are\_enzymes flag forces the program to treat all proteins as enzymes
- --Skips the enzyme/non-enzyme classification step and proceeds directly to EC number prediction
- --All proteins will have pred\_label set to 1 and pred\_prob set to 1.0

## Output

After running, the prediction results will be saved to:

```
./results/example_test_final_pred_results.csv
```