User Guide of Flour-predictor

The initial interactive interface of Flour-predictor is shown below, which mainly includes two functional modules: "Retrieval and Similarity" and "Property Prediction." Users can simply select the corresponding module based on their needs. Flour-predictor does not require registration or login, and there are no limitations on usage. It is a simple and efficient standalone software. Users can also directly use the provided GUI.py to add and modify software features. The interface upon entering is shown in Figure 1.

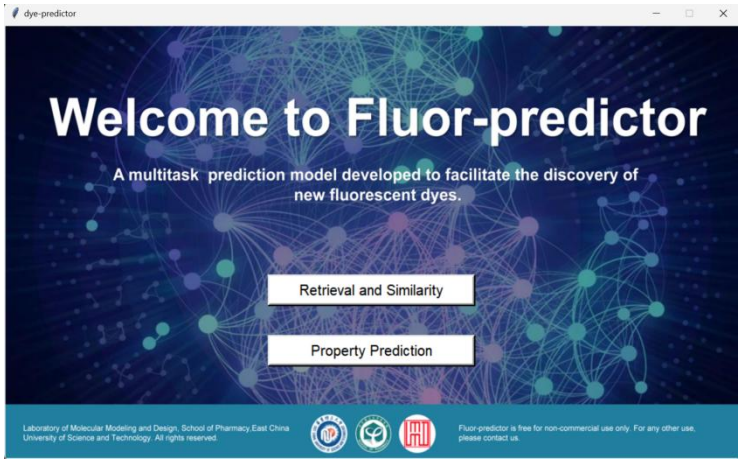

Figure 1. The initial interface of Flour-predictor

■ Function 1: Dye Retrieval and Similarity Search.

This section includes two specific applications, allowing users to choose based on their needs. The first is the dye molecule retrieval function, where users need to input the SMILES of the dye molecule. The model will automatically convert it to a standard SMILES and search our organized database for the optical information related to the target dye-solvent pairs, saving the results in the Results folder under target_search.csv. This functionality greatly facilitates researchers in retrieving information about related molecules and quickly determining whether relevant experimental data and specific information are available for the target dye.

The second part is the similarity search function. This feature is particularly useful for new compounds, as it can identify existing compounds that are similar to the new compound, providing a general understanding of its optical information.

When users input the SMILES of the new compound, the software will automatically generate its Morgan fingerprint. By calculating the Tanimoto similarity of this fingerprint with those of other dyes in the database, the model will provide the top 100 most similar entries, saving the results in Similarity_search.xlsx in the Results folder. The number of recommendations given by the model can be defined by the user in GUI.py.

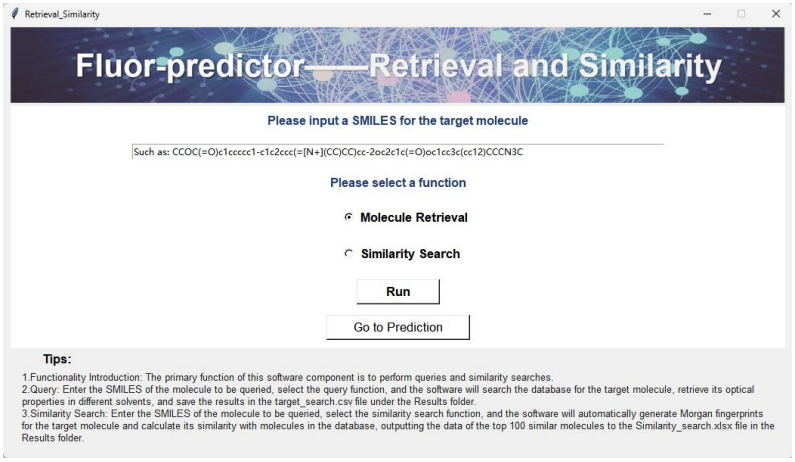

Figure 2. The interface of Function 1

■ Function 2: Dye Property Prediction.

We used three trained prediction models: one for general dye prediction, and two specifically for xanthene and cyanine dyes. To enhance the model's generalization ability, we employed models trained on dye molecule-based splitting. Users need to select the appropriate model based on their molecular type; the xanthene and cyanine models are trained exclusively on these two classes of data and should not be used for predicting other categories. When using the models, please first select the prediction model for the target molecule, then input the SMILES of the dye molecule and the solvent molecule. Additionally, the model supports multiple entry inputs, allowing you to input multiple dye and corresponding solvent SMILES, separated by commas. Please ensure that the number of dye molecules corresponds to the number of solvents. After running the prediction, a results display box will pop up, showing the predicted values for the four properties of the target molecule. The results will also be saved in the Results folder under pred-results.csv.

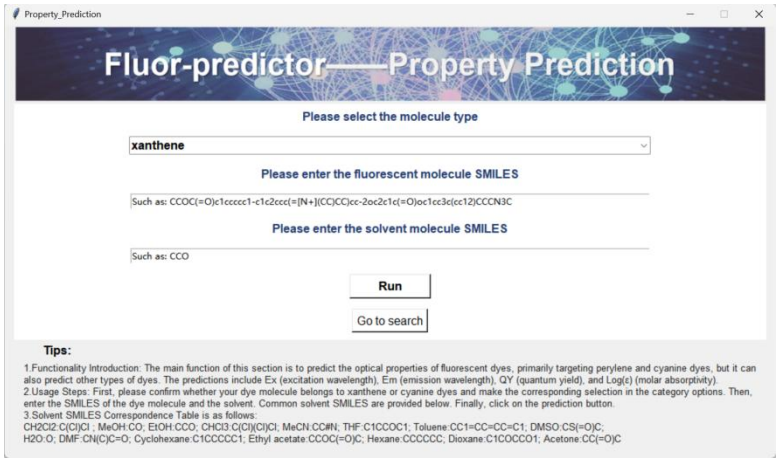

Figure 3. The interface of Function 2

■ Creation of the Software Usage Environment：

Since the software requires calling deep learning libraries, we have attempted packaging, but the resulting software is very large and slow to load. Therefore, we recommend using it in your own Conda virtual environment. Below is the complete environment creation code; follow the steps to install:

1. To Create the Environment:

conda create -n dye37 python=3.7

2. Library Installation:

conda install pytorch==1.13.1 torchvision==0.14.1 torchaudio==0.13.1 pytorch-cuda=11.7 -c pytorch -c nvidia
pip install pandas==1.3.0
pip install dgllife==0.2.8
pip install rdkit-pypi
pip install dgl==1.1.2+cu117 -f https://data.dgl.ai/wheels/cu117/repo.html
pip install packaging
conda install xlsxwriter
conda install -c anaconda scikit-learn

3. Correction of Source Code Errors:

In GUI.py, jump to the code in attentivefp.py through AttentiveFPGNN, rename

copy_edge to copy_e; rename src_mul_edge to u_mul_e. After the installation is complete, you can run it directly.