Mol-Fragmentation: Molecular Fragmentation Tutorial

This tutorial will guide you through installing and using a Python tool for **molecular fragmentation** using RDKit. The script processes a CSV of molecules and generates complementary fragments with fingerprints.

Step 1: Set Up Your Environment

We recommend using **Conda** to manage dependencies.

1. Install Conda from Miniconda if you don't have it.
2. Save the following content as `environment.yml`:

```yaml
name: mol-fragmentation
channels:
  - rdkit
  - conda-forge
  - defaults
dependencies:
  - python=3.9
  - rdkit
  - matplotlib
  - pandas
  - numpy
  - pip
  - pip:
      - umap-learn
```

3. Create the environment:

```
conda env create -f environment.yml
conda activate mol-fragmentation
```

4. Check that it works:

```
python -c "import rdkit, pandas, numpy, matplotlib; print('Environment ready!')"
```

Step 2: Prepare Your CSV File

Your CSV should contain at least two columns:

ID	SMILES
Mol1	CC(=O)NC1=CC=CC=C1
Mol2	C1=CC=CC=C1O

- **ID**: unique identifier for each molecule.
- **SMILES**: canonical SMILES representation.

Step 3: Run the Fragmentation Script

The script is called `fragmentation.py`. Run it in your terminal:

```
python fragmentation.py
```

Step 3a: Interactive Prompts

You will be asked for:

1. **CSV file path** – Full path to your CSV file.
2. **CSV separator** – Usually `,` or `;`.
3. **SMILES column name** – Column with SMILES strings.
4. **Identifier column name** – Column with molecule IDs.

Step 3b: Fragmentation Process

The script will:

- Parse each SMILES.
- Apply **RECAP**, **BRICS**, and custom fragmentation rules.
- Avoid breaking fused rings and optionally neighboring atoms.
- Keep only **maximum complementary fragment sets**.
- Generate **ECFP4 fingerprints** for each fragment.

Step 4: Output

After completion, the script creates `fragments.csv`:

Identifier	Original_SMILES	Fragment_SMILES	Rule	Mode	ECFP4	UMAP_1	UMAP_2
Mol1	CC(=O)NC1=CC=CC=C1	CC	Urea	ALL	[0,1,0,…,0]	17.12	8.27
Mol1	CC(=O)NC1=CC=CC=C1	NC1=CC=CC=C1	Urea	CRUSH, RECAP or BRICS	[1,0,0,…,1]	7.12	14.17

- `Rule` shows which fragmentation rule was applied.
- `Mode` shows which fragmentation set was applied.
- `ECFP4` contains the Morgan fingerprint of the fragment (radius=2, 2048 bits).

Step 5: Tips & Notes

- **Minimum fragment size**: Default is 3 heavy atoms.
- **Expanded rules**: Set to True to use CRUSH, RECAP, and BRICS custom SMIRKS.
- **Debug mode**: Prints warnings for unrecognized SMILES.

You are now ready to fragment molecules like a pro! 🎯