Summary of our code

Our project performs virtual screening of small molecules against the alpha-synuclein protein related to Parkinson's disease which involves downloading compound data, preparing ligand files, docking ligands with a receptor, and finally generating complexes of receptor-ligand pairs for analysis.

Step 1: Compound Data Download and Preprocessing (sdf_to_pdbqt.py)

3D structures of compounds, their structurally similar molecules, and substructure matches are downloaded from PubChem

Further steps involved here:

1. **Input Data**:
    - Provide a list of PubChem IDs of interest.
    - Specify the number (`N`) of similar and substructure compounds to download.
2. **Files/Directories Created**:
    - Creates an `SDF` directory to store downloaded files.
3. **Download 3D Structures**:
    - For each PubChem ID:
        i. Check if the `.sdf` file exists.
        ii. If not, download the 3D structure from PubChem and save it in the `SDF` directory.
    - Implements error handling for missing structures and failed requests.
4. **Fetch Similar and Substructure Compounds**:
    - Queries PubChem for:
        i. Compounds with high 3D structural similarity.
        ii. Compounds matching substructure patterns.
    - Downloads these additional structures.

Step 2: Converting SDF to PDBQT(download_sdf.py)

The downloaded `.sdf` files are converted to `.pdbqt` format using Open Babel for molecular docking

Further steps involved here:

1. **Files/Directories Created**:
    - Creates a `PDBQT` directory to store the converted files.
2. **Conversion Process**:
    - Iterates through all `.sdf` files in the `SDF` directory.
    - Converts each file to `.pdbqt` format using Open Babel.
    - Stores converted files in the `PDBQT` directory.
    - Prints success or error messages for each conversion.

Step 3: Molecular Docking with AutoDock Vina(perform_docking.py)

The docking simulations of ligands are performed against the alpha-synuclein receptor

Further steps involved here:

1. **Files/Directories Created**:
    - Creates directories: `OUT` (output files), `LOG` (log files), and `Config` (configuration files).
2. **Configuration**:
    - Reads receptor configuration parameters from `.txt` files in the `Config` directory (e.g., box center and size).
3. **Docking Process**:
    - Iterates through all `.pdbqt` ligand files.
    - Runs AutoDock Vina for each ligand using receptor and configuration data.
    - Stores output in the `OUT` directory and logs in the `LOG` directory.
4. **Result Parsing**:
    - Extracts docking results (affinity scores, RMSD values) from the log files.
    - Summarizes the top 3 results per configuration and overall top 3 results.
    - Saves a summary as `docking_summary.csv`.

Step 4: Receptor-Ligand Complex Creation(create_complex_from_ligand.py)

Receptor-ligand complex structures are generated in `.pdb` format for further analysis.

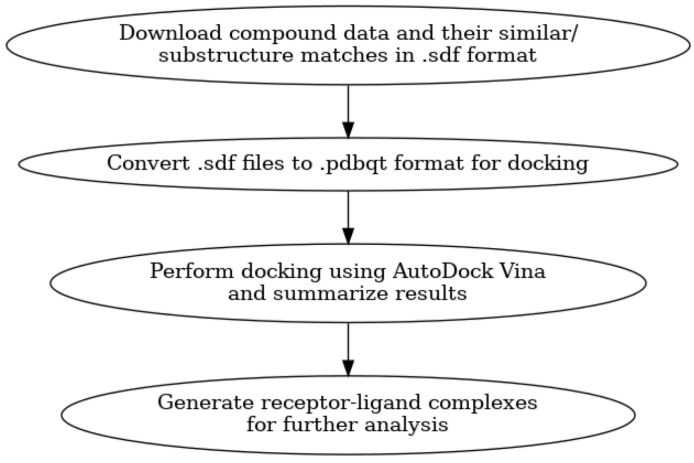
Further steps involved here:

1. **Files/Directories Created**
    - Creates a `Complexes` directory to store generated complexes.

2. **Process Ligands**:
    - Reads top-performing ligands from the `Top` directory.
3. **Generate Complexes**:
    - Converts receptor and ligand `.pdbqt` files to `.pdb` using Open Babel.
    - Extracts only `Model 1` from both receptor and ligand files.
    - Combines receptor and ligand into a single `.pdb` file.
4. **Output**:
    - Writes combined files to the `Complexes` directory.
5. **Files/Directories Deleted**:
    - Deletes temporary `.pdb` files after processing.

Summary and flow of execution:

We have completely automated the process, by combining the 4 python files into a bash script which runs these 4 files on its own. Initially, one needs to provide the pubchem ids of the ligands of interest and the uniprot id of the target molecule, and rest everything else is taken care of.



Dependencies

- **Python Libraries Used**: `os`, `requests`, `subprocess`, `csv`, `time`, `concurrent.futures`, `pathlib`

- **External Tools Used**:
    - Open Babel: For file format conversion.
    - AutoDock Vina: For molecular docking simulations.

List of output Files/Directories

- **Directories**:

    - `SDF`: Stores `.sdf` files of compounds.
    - `PDBQT`: Stores `.pdbqt` files for docking.
    - `OUT`: Stores docking results (`.pdbqt`).
    - `LOG`: Stores log files from docking simulations.
    - `Complexes`: Stores receptor-ligand `.pdb` complexes.

- **Summary File**: `docking_summary.csv` with docking statistics.