1.1 WP1 – AI & material search

The following instructions exemplifies a) how to generate some new molecules and b) how to submit them to a screening procedure. We assume a linux operating system.

1. Generate some compounds

Follow download and setup instructions given at <u>https://github.com/Fraunhofer-SCAI/llamol</u>. We suggest to use vscode to run the notebook.

- code.
 - 1. file explorer to the left \rightarrow click on **demonstrator.ipynb** to open notebook
 - 2. if necessary, choose python environment torch2-llamol
 - 3. hit 'Run All' button in top row, scroll down
 - 4. specify target properties
 - a) tick box for desired property
 - b) move slider and/or
 - c) type in SMILES fragment
 - 5. hit 'Generate' button at bottom of notebook. The output files *chemiscope_gen.json* and *gen_smiles.txt* will be generated . Scroll down to get a first overview.

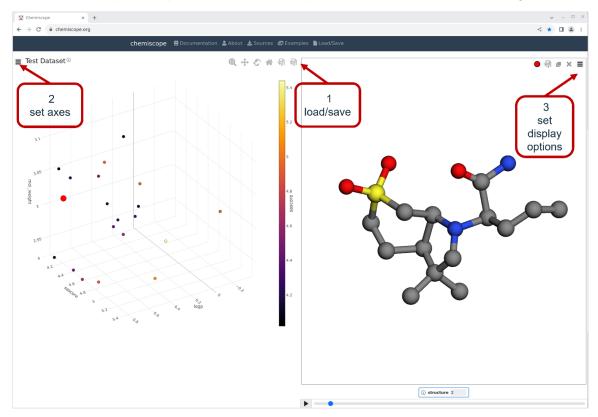
 ♦ demonstrator jpynb - wp1 - Visual Studio Code 							- • ×
File Edit	Selection View Go Run Terminal Help						
G	EXPLORER ····	🗙 Welcor	me 📑 demonstrator.ipynb •	💩 merge_jsons.py	<pre>{} chemiscope_gen.json</pre>	<pre>{} chemiscope_gen_1.json</pre>	☺ □ …
	✓ WP1						
	>pycache	+ Code			uts 🕅 Variables 📃 Outl		🚊 torch2-bachelor (Python 3.8.16)
، مح اللہ کے	> data {} all.json {} chemiscope_gen.json 5 @ demonstrator.jpynb 1 @ fragment_creator.py @ gen_smiles.txt 6 & llama2-W-full-RSS.pt	▷ ~	display(context_smi_input) display(temperature_slider display(generate_button) display(output) display(molecule_output)		mosnergnt		喧 🎠 💫 🖯 … 🝵
ш	merge_jsons.py	[3] 🗸	0.0s				Python
	🕏 model.py 🕏 plot_utils.py		🗹 logp	ſ	✓ sascore 3	🗆 mol.	weight
00	preprocess_dataset.py sample.py		logp:	3.00 sasco	re:	6.00 mol_weight:	3.00
	<pre>{} test_1.json {} test_2.json</pre>		Context SMI: C1=CSC=C1				
	<pre>{} test_3.json test_all.py </pre>		emperature:	0.80			
	 {} test.json de tokenizer.py de train.py 		Generate 4				
	🅏 trainer.py						
			ython_env ython_merge_jsons.p	y first_expe	eriment second_e	experiment … all_	experiments.json
8		PROBLEMS	S OUTPUT DEBUG CONSOLE TER	MINAL JUPYTER			() bash + ∨ □ □ □ ··· ^ ×
500	> OUTLINE > TIMELINE	<pre>o (base)</pre>	bash-4.4\$				1
*							Ln 120, Col 40 🔗 🔓 - { }

- in case you run several experiments, you have to rename the output files (else they will be overwritten), e.g. <u>mv chemiscope_gen.json first_experiment.json</u>
- to compile the results
 - enter python_env in the bottom terminal window
 - o python merge_jsons.py first_experiment second_experiment ... all_experiments.json

2. Inspect results

continue with file chemiscope_gen.json

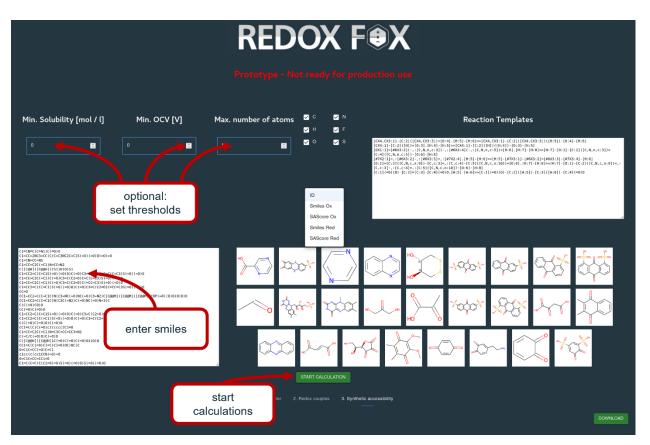
- open https://chemiscope.org in browser
 - o press 'load/save' button in middle of top row
 - upload chemiscope_gen.json
 - o optional: press menu icon in top left corner to adapt 'Map settings'



3. Carry out screening experiment

continue with file gen_smiles.txt

- mark and copy content of file, i.e. list of smiles
- open <u>www.redoxfox.scai.fraunhofer.de</u>
 - o replace default smiles in the white field on the left
 - mark & delete SMILES
 - paste list of copied SMILES
 - o optional: adjust threshold criteria (fields above)
 - hit green 'START CALCULATION' button
 - if posolyt/negolyte pairs survive. i.e. step 7 is reached:



4. Draw your own candidate or fragment SMILES

open cheminfo.org/flavor/malaria/Utilities/SMILES_generator___checker/index.html

- Draw molecule \rightarrow SMILES appears in box below
- Copy & paste into
 - \circ Redoxfox \rightarrow run screening –OR--
 - $\circ~$ Demonstrator: field 'Context Smi' \rightarrow use your molecule as seed structure for derivatives

When you are done: