

The logo consists of the word "SONAR" in a bold, blue, sans-serif font. The letter "O" is replaced by a white hexagon containing a blue shield with a white cross and a blue circle below it. The entire logo is set against a white hexagonal background with a blue border.

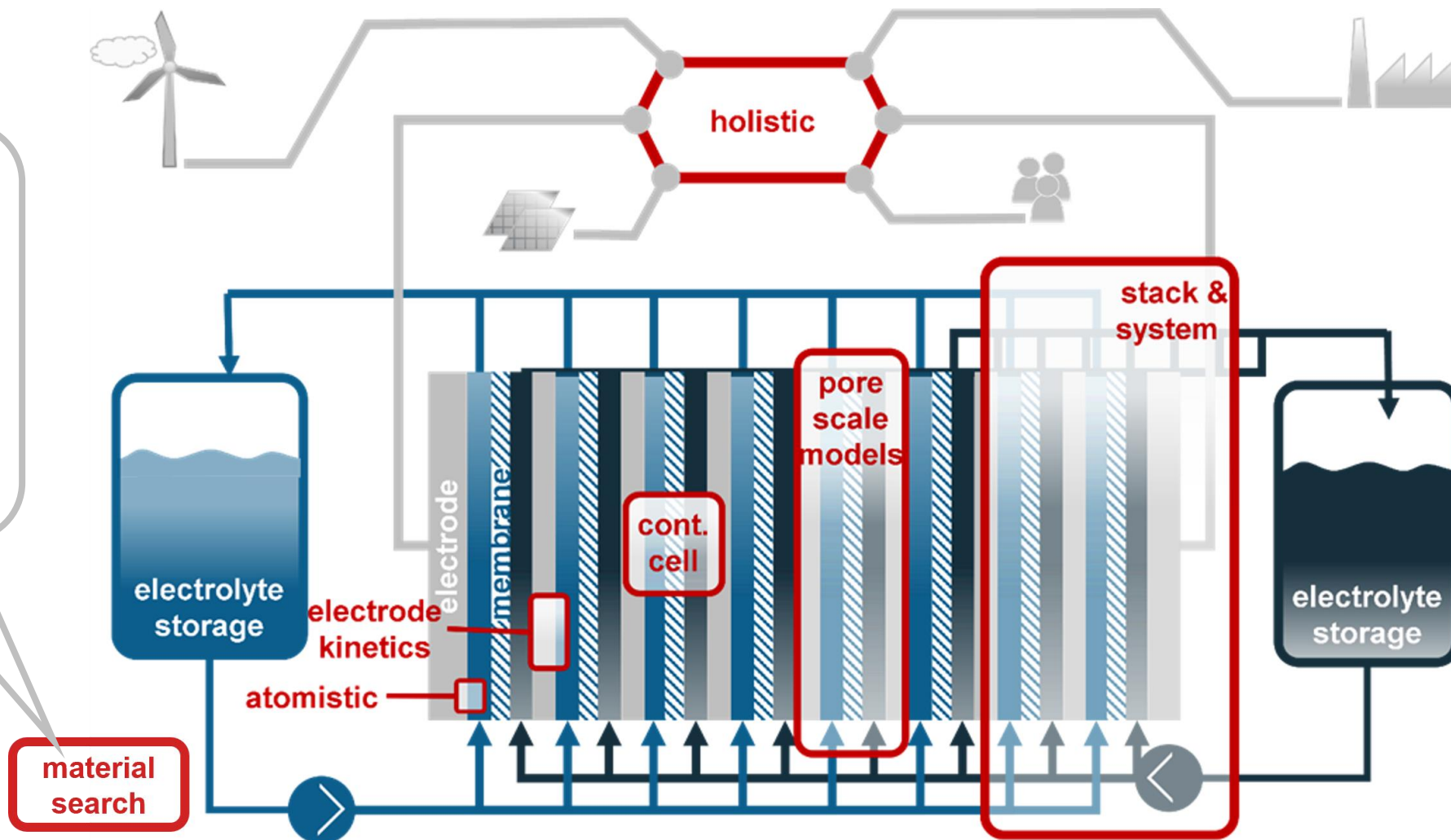
SONAR

AI & material search

Fraunhofer SCAI
Workshop 2023
St.Augustin, 21.09.2023

SONAR at a glance

task WP1:
tools to identify
organic
affordable
redox-active
materials



search

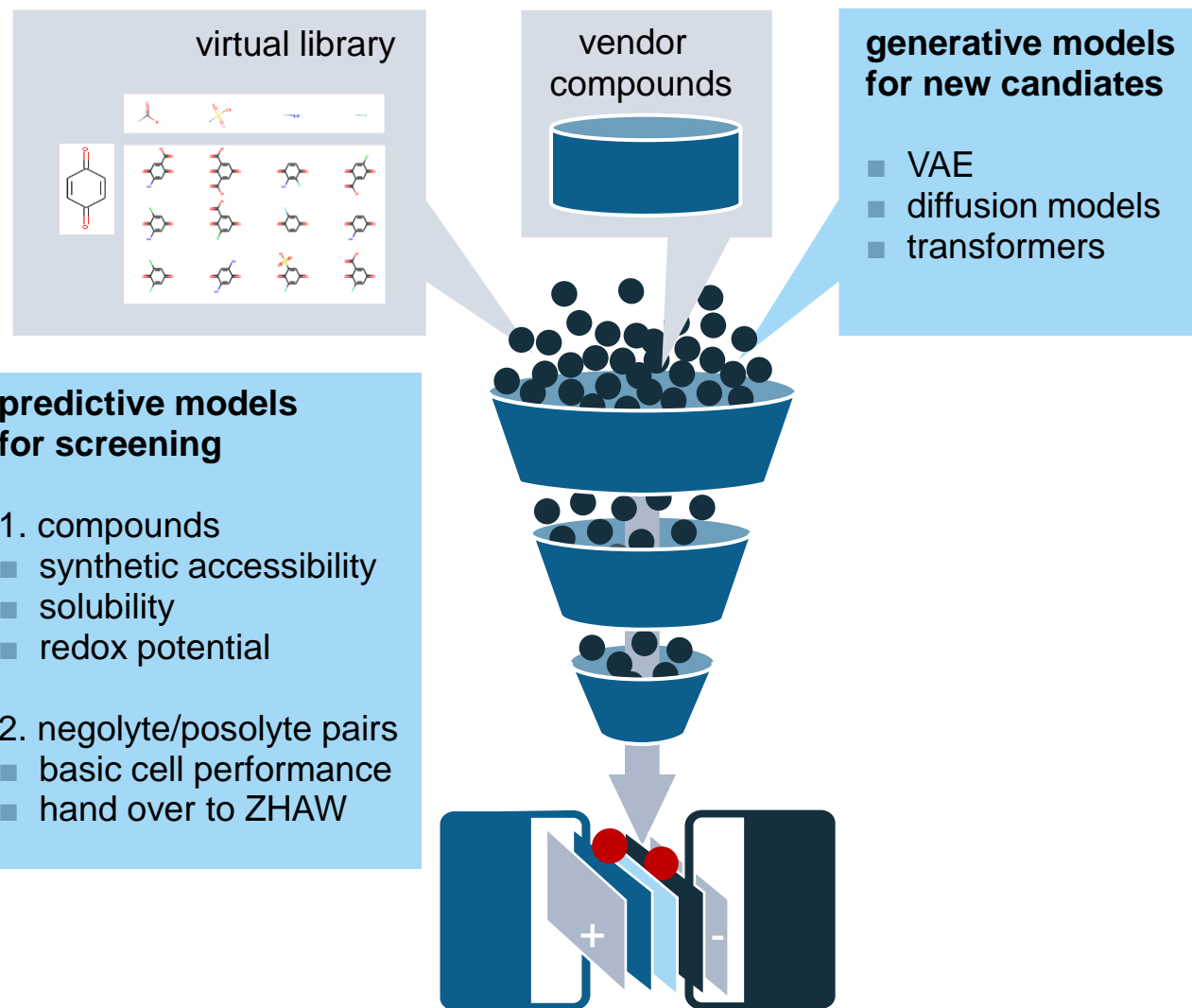
organic chemical space $\sim 10^{23} - 60$
for high energy density at low cost

speed & focus

efficient down-selection

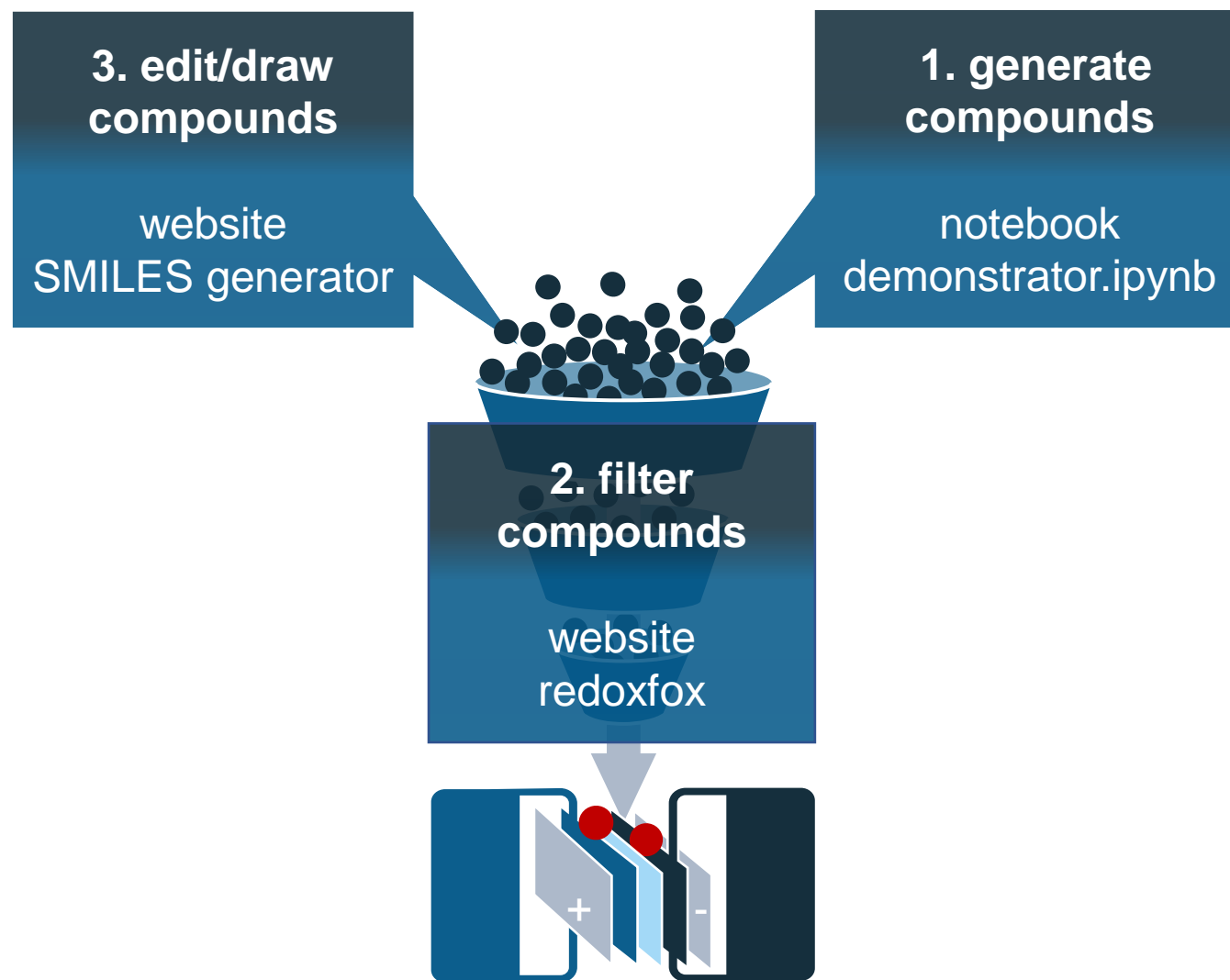
cheap filters first
then expensive

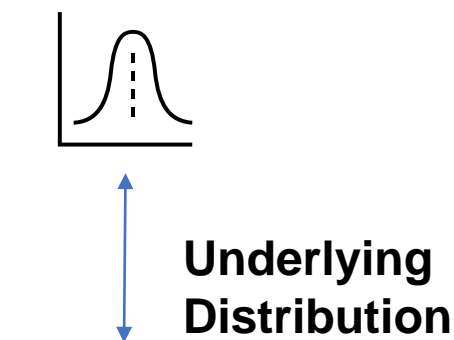
all fast



Hands on session

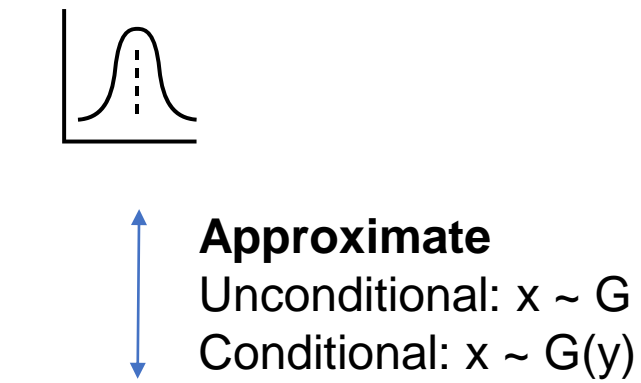
- + page 20 in handout
- + jupyter notebook
- + webinterface
 - chromium-browser &



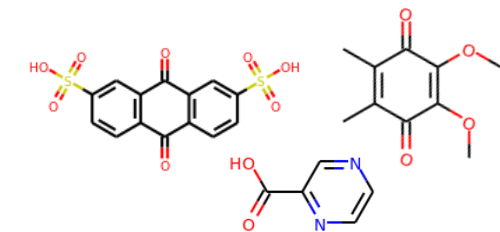


Dataset: (x,y) pairs
 x - Molecule
 y - Property

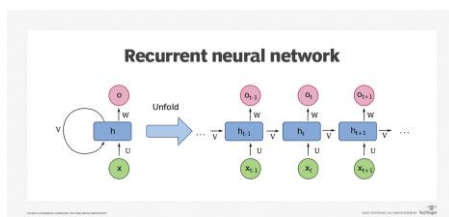
Train



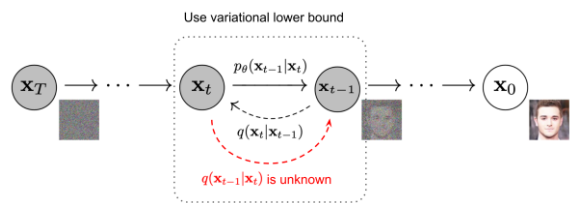
Sample



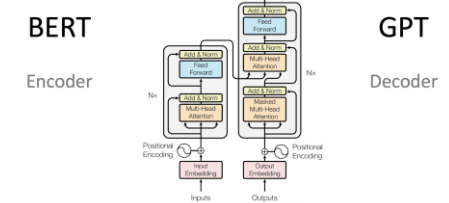
RNNs



Diffusion

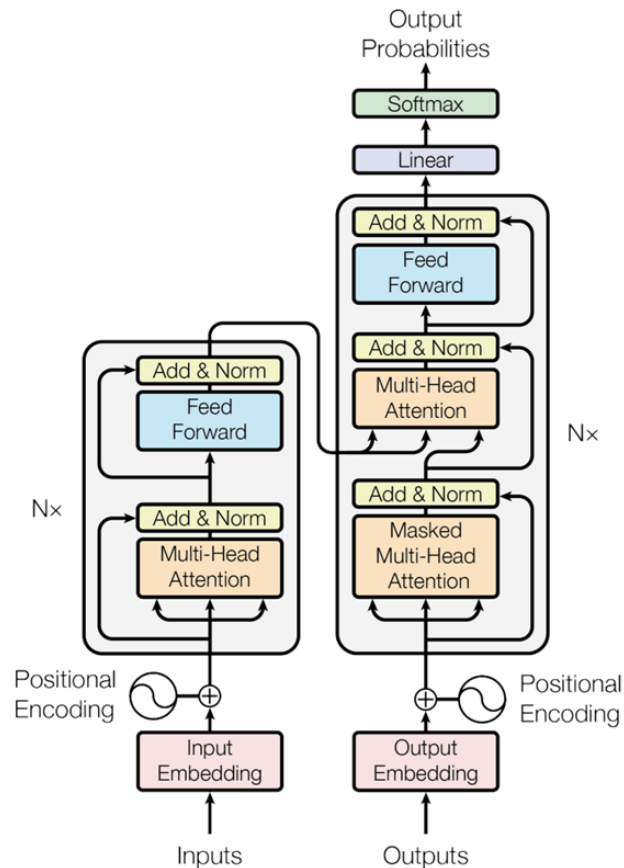


Transformer



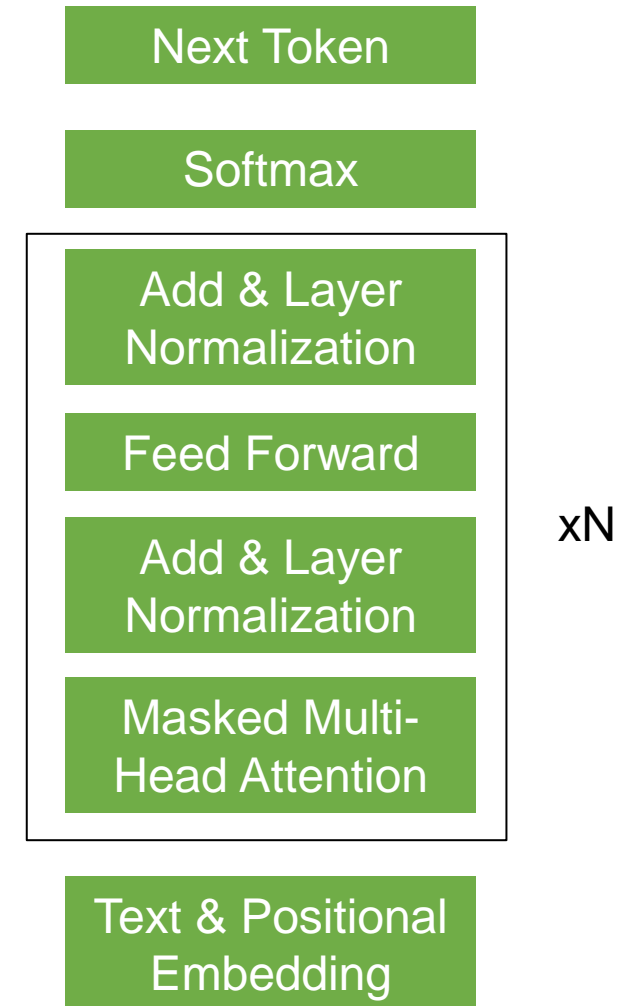
BERT

Encoder

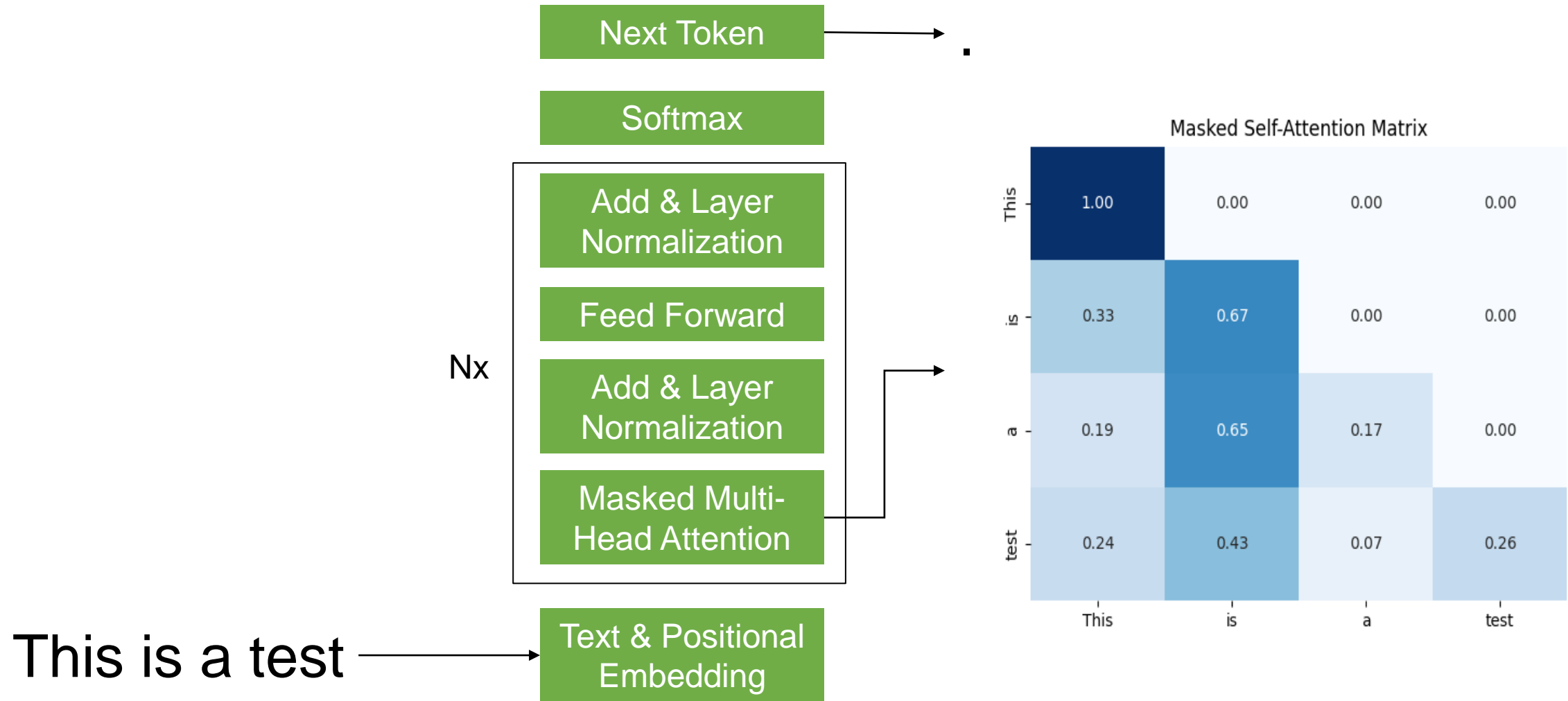


GPT

Decoder



Transformer – Basic Generation



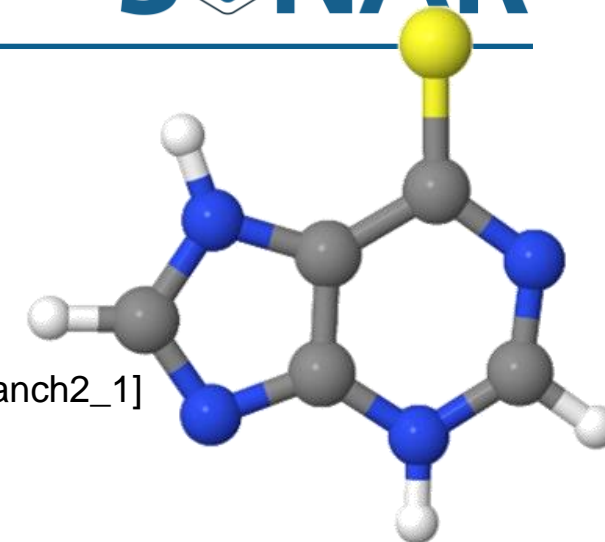
Molecule representation

2D

InChi InChI=1S/C5H4N4S/c10-5-3-4(7-1-6-3)8-2-9-5/h1-2H,(H2,6,7,8,9,10)

SMILES C1=NC2=C(N1)C(=S)N=CN2

SELFIES [C][=N][C]=[C][Branch1_1][Ring2][N][Ring1][Branch1_1][C][Branch1_2][C]=[S][N]=[C][N][Ring1][Branch2_1]



3D

Coordinates

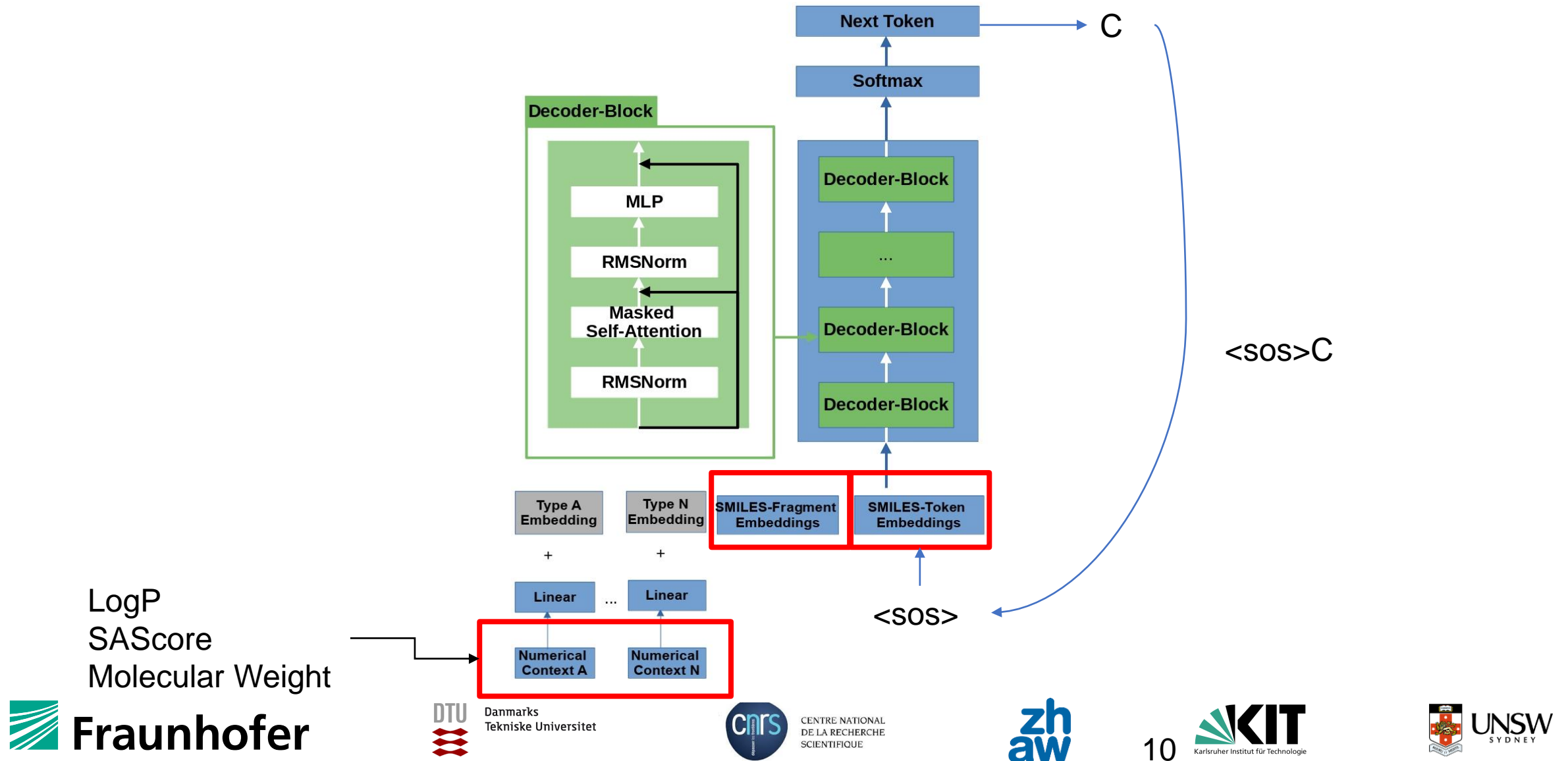
C	-2.28752	-0.09279	0.00131
N	-1.52093	-1.16587	0.00160
C	-0.26385	-0.66545	0.00005
C	-0.23058	0.70099	-0.00110
N	-1.55044	1.06537	0.00000
⋮	⋮	⋮	⋮

Descriptors

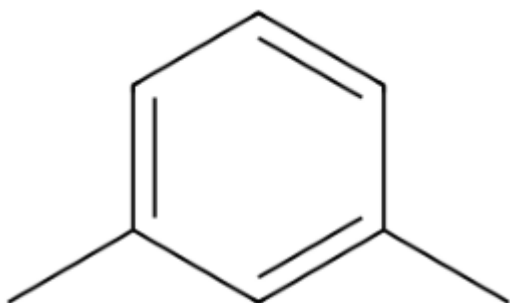
- matter of choice
- meaningful
- Cheap
- LogP, SAScore and Molecular Weight

- + **LogP** [-4, 8]: How hydrophobic is a molecule?
 - Negative: soluble in water
 - Positive: attracted to non-polar environments
- + **SAScore** [0,10]: ease(0) or difficulty(10) to synthesize a molecule
- + **Molecular Weight** [0, 10]: proxy for the size of a molecule
 - We divide it by the constant 100 to „normalize“
- + **SMILES Fragment**: Can be a full molecule or a part of a SMILES
e.g. „c1cccc1“ or „csc“

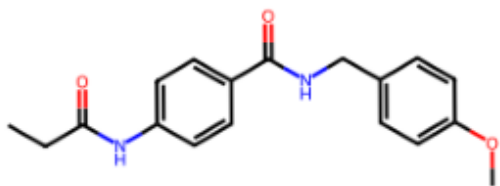
Transformer – Molecule Generation



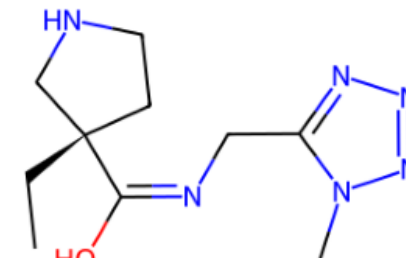
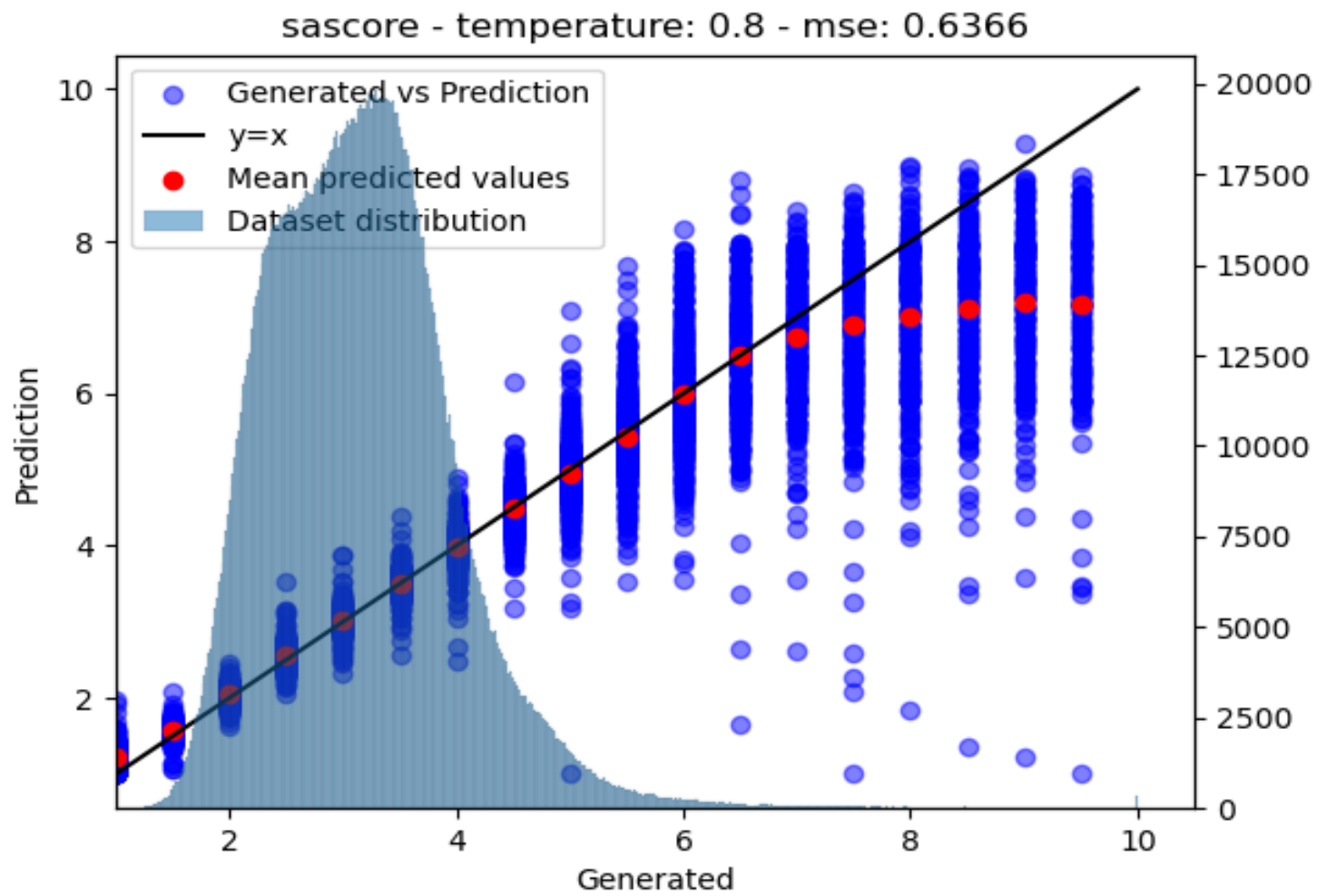
Transformer – Performance SAScore



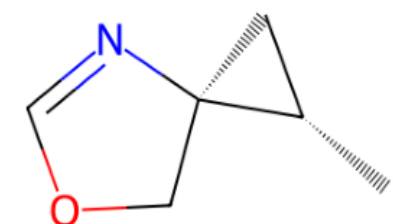
sascore: 0.5 vs 1.14



sascore: 1.5 vs 1.54

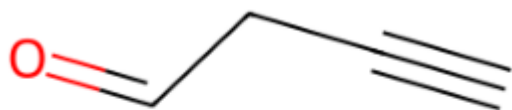


sascore: 4.0 vs 3.93

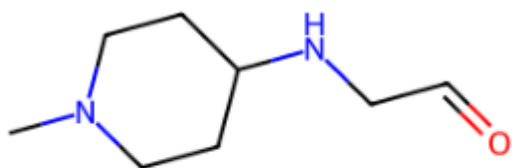


sascore: 5.5 vs 5.53

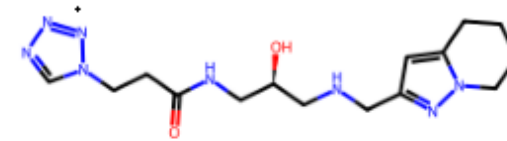
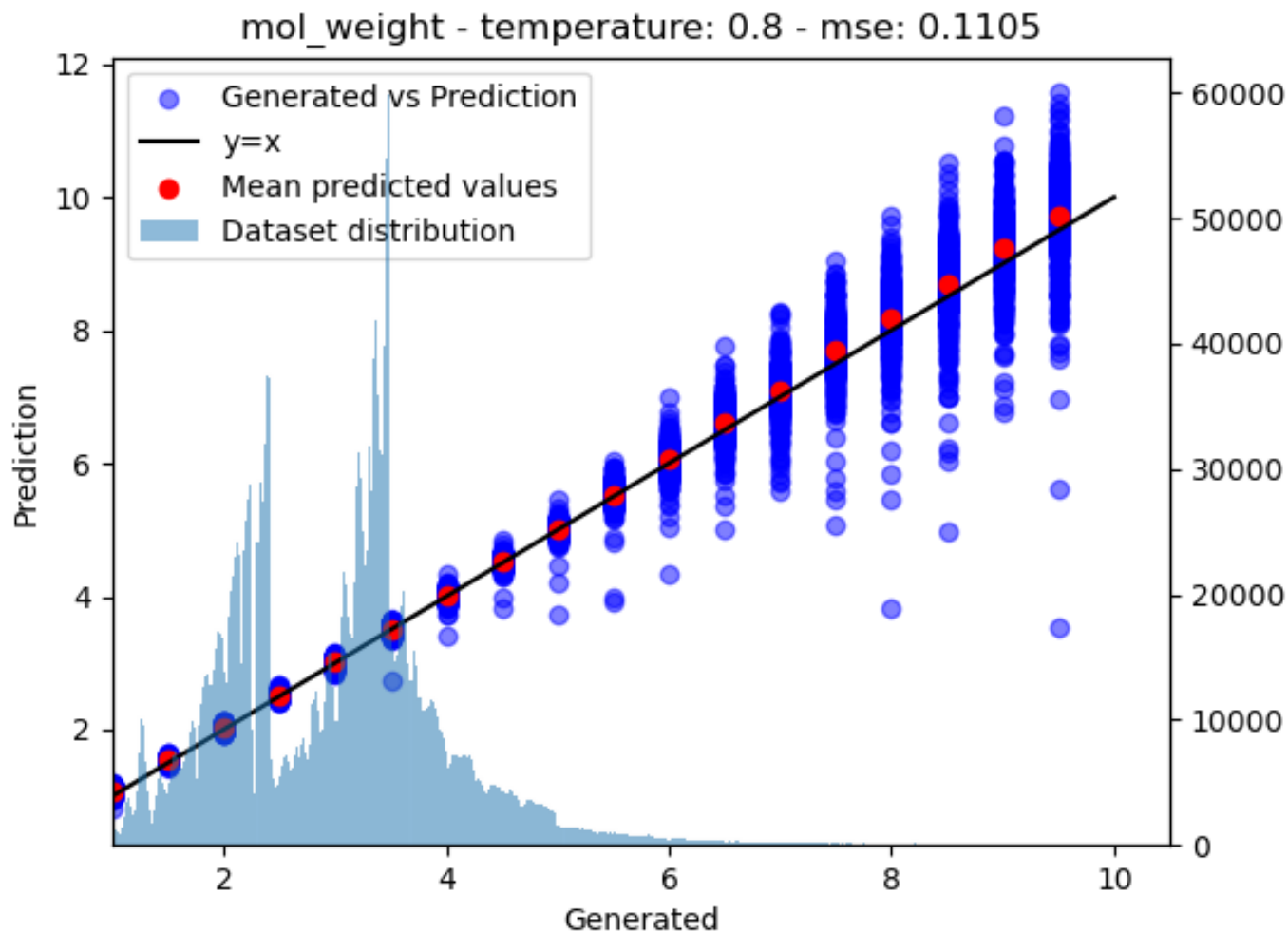
Transformer – Performance Molecular Weight



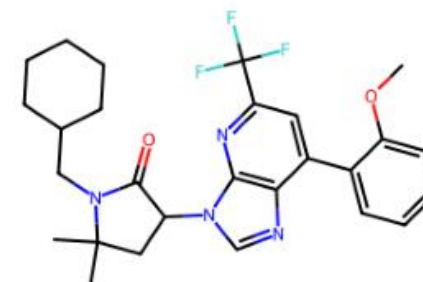
mol_weight: 0.5 vs 0.68



mol_weight: 1.5 vs 1.56



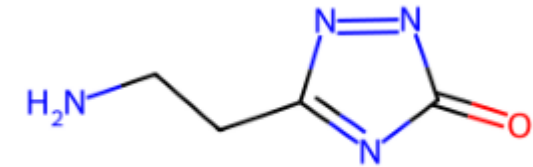
mol_weight: 3.5 vs 3.48



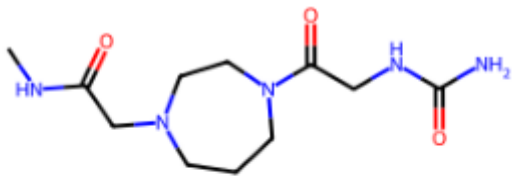
mol_weight: 5.0 vs 5.0

Transformer – Performance Multi Condition

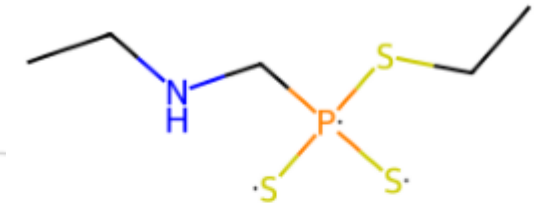
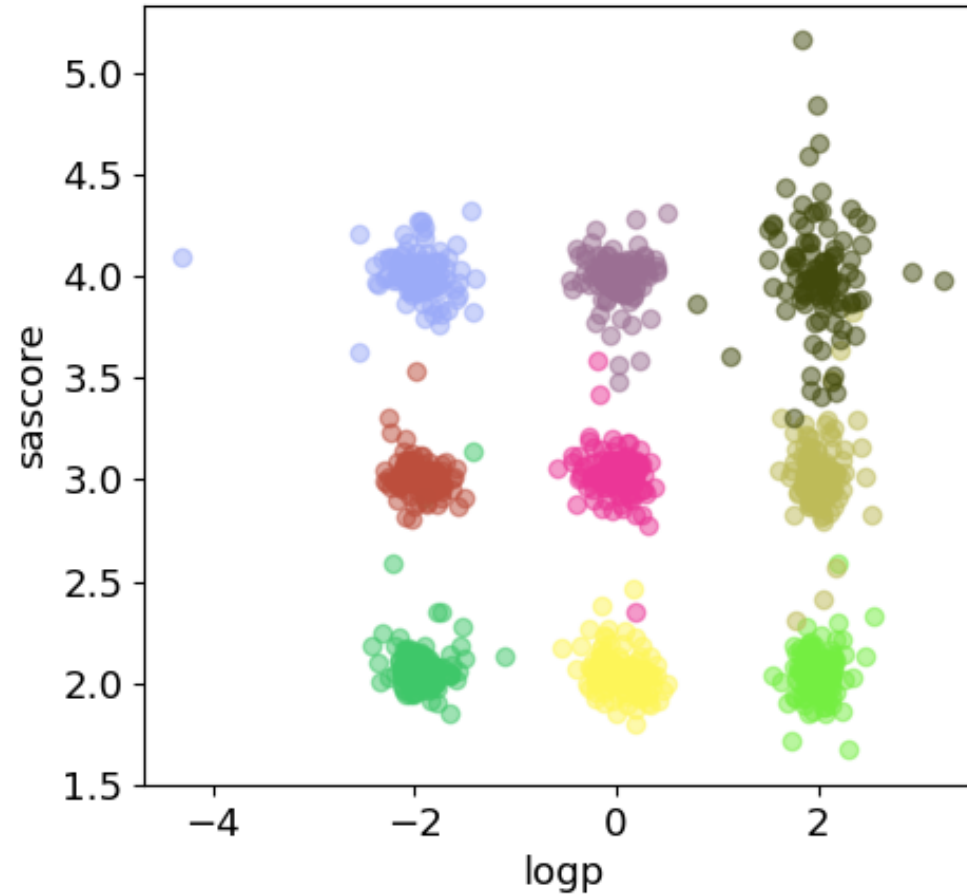
Multi Property Distribution of Generated Molecules



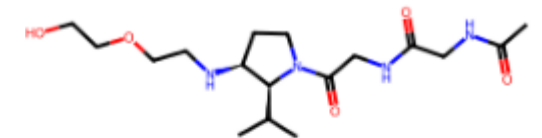
logp: 0.5 vs 0.32
sascore: 4.0 vs 4.12



logp: -2.0 vs -2.06
sascore: 2.0 vs 2.13

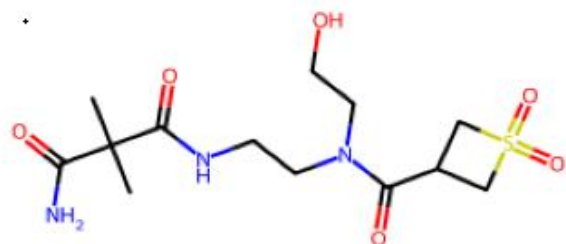


logp: 3.0 vs 3.46
sascore: 5.5 vs 4.45



logp: -1.5 vs -1.54
sascore: 3.5 vs 3.53

Transformer – Performance Multi Condition



logp: -2.5 vs -2.52

sascore: 3.0 vs 2.95

mol_weight: 3.5 vs 3.49

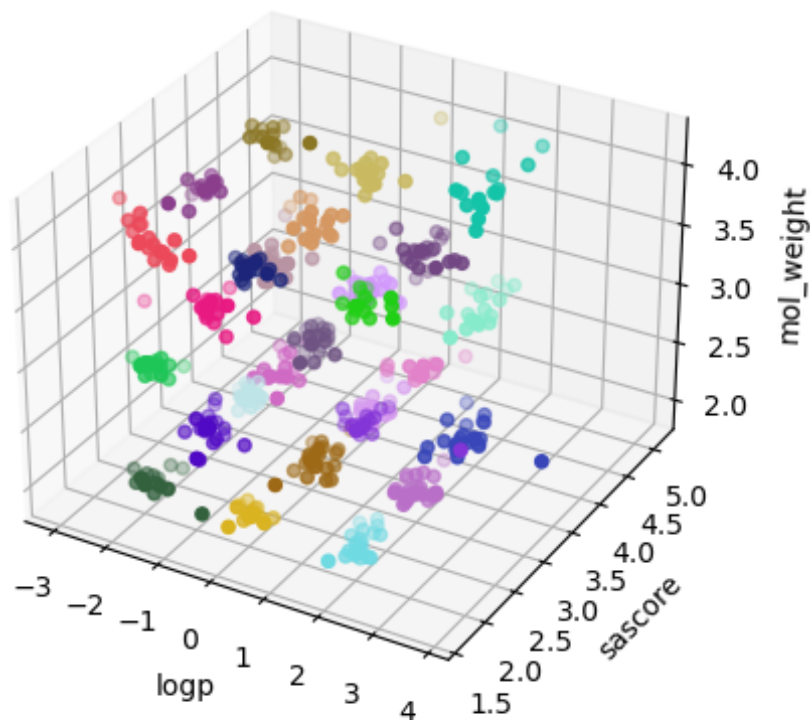


logp: 3.0 vs 3.07

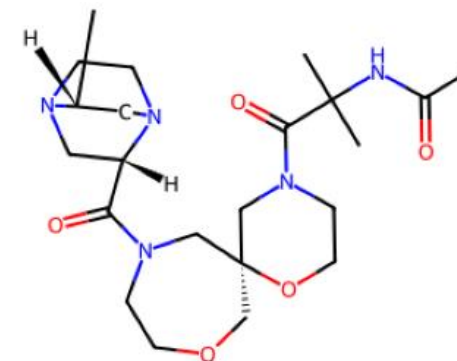
sascore: 6.0 vs 6.07

mol_weight: 6.5 vs 6.66

Multi Property Distribution of Generated Molecules



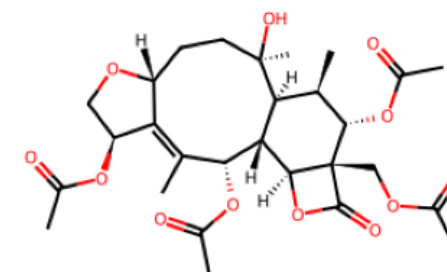
- (-2.0, 2.0, 4.0)
- (-2.0, 3.0, 2.0)
- (-2.0, 3.0, 3.0)
- (-2.0, 3.0, 4.0)
- (-2.0, 4.0, 2.0)
- (-2.0, 4.0, 3.0)
- (-2.0, 4.0, 4.0)
- (0.0, 2.0, 2.0)
- (0.0, 2.0, 3.0)
- (0.0, 2.0, 4.0)
- (0.0, 3.0, 2.0)
- (0.0, 3.0, 3.0)
- (0.0, 3.0, 4.0)
- (0.0, 4.0, 2.0)
- (0.0, 4.0, 3.0)
- (0.0, 4.0, 4.0)
- (2.0, 2.0, 2.0)
- (2.0, 2.0, 3.0)
- (2.0, 2.0, 4.0)
- (2.0, 3.0, 2.0)
- (2.0, 3.0, 3.0)
- (2.0, 3.0, 4.0)
- (2.0, 4.0, 2.0)



logp: -1.0 vs -1.25

sascore: 5.5 vs 5.61

mol_weight: 4.5 vs 4.51



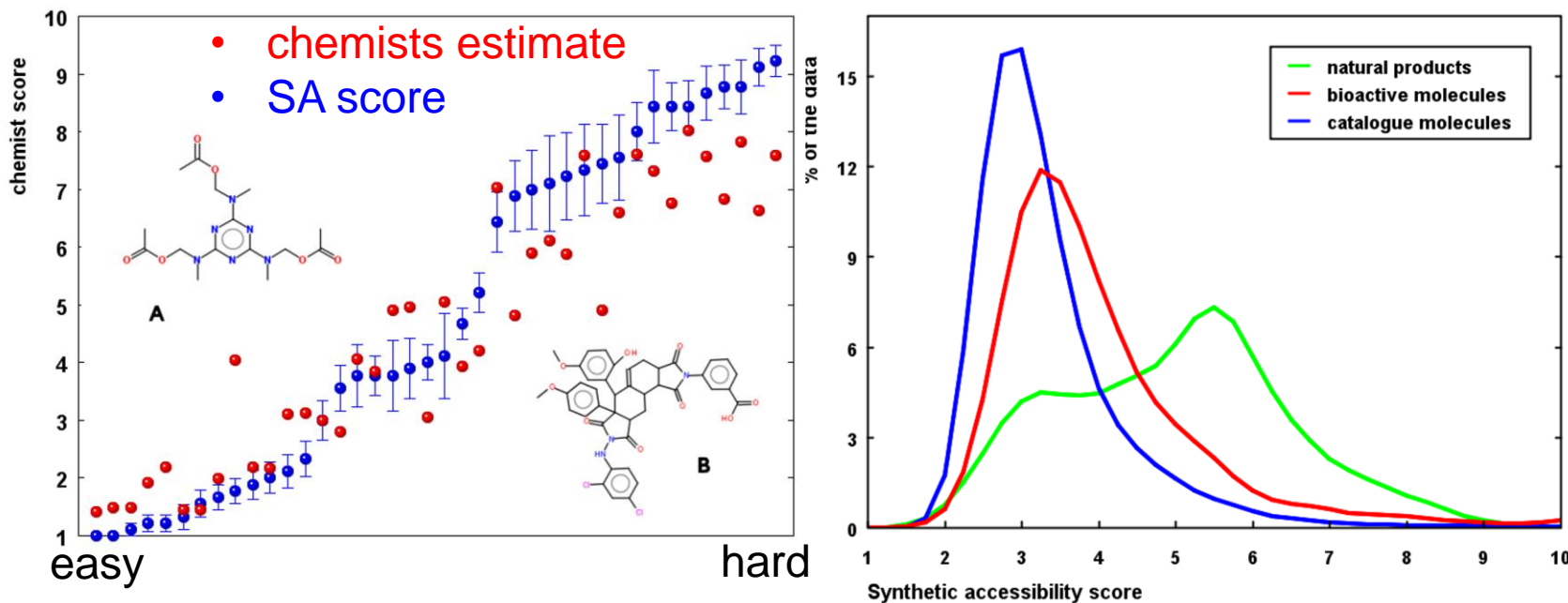
logp: 2.0 vs 1.4

sascore: 6.0 vs 5.92

mol_weight: 5.5 vs 5.66

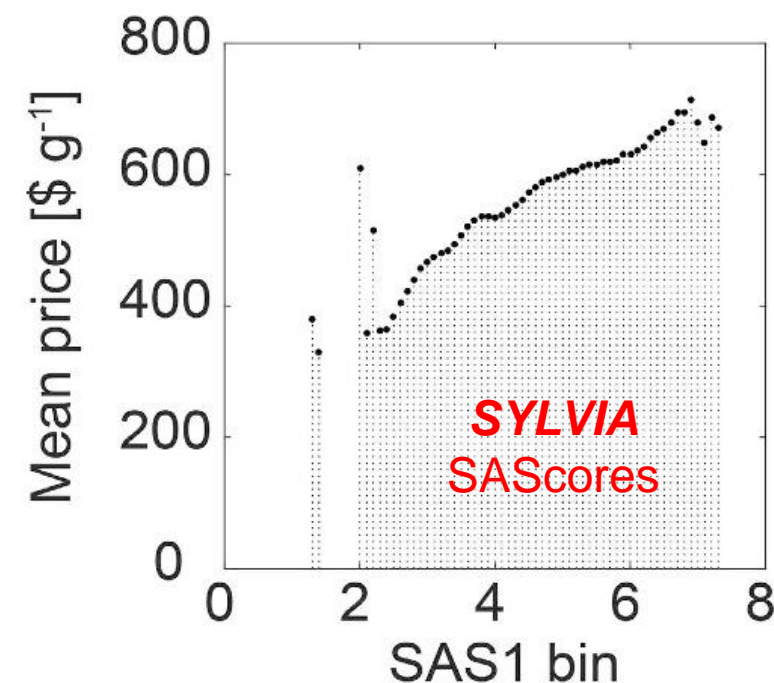
Synthetic accessibility score

Estimate SA score of drug-like molecules based on molecular complexity and fragment contributions



Molecular descriptor data explain market prices of a large commercial chemical compound library

Polanski et al, 2016

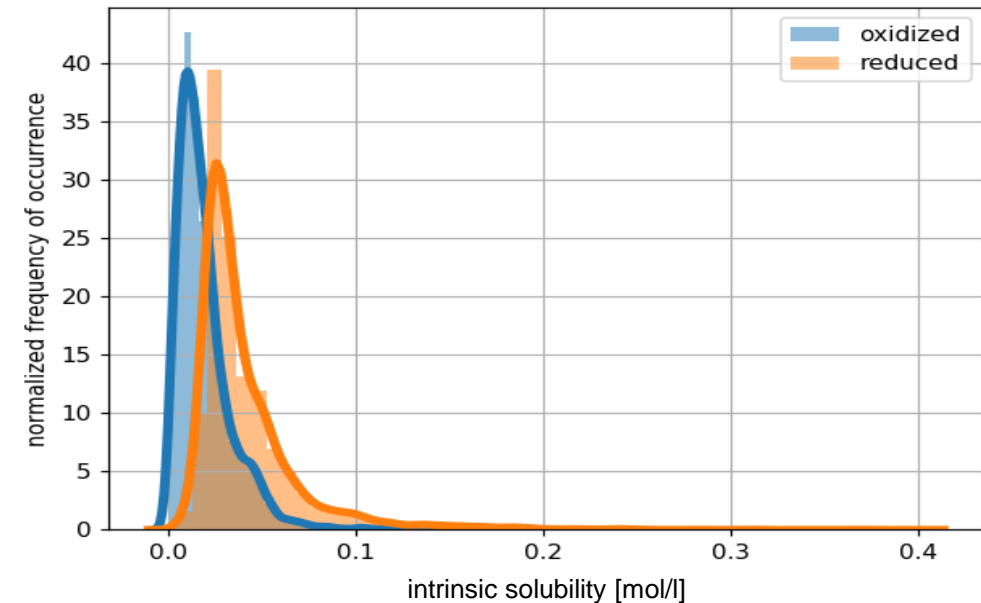
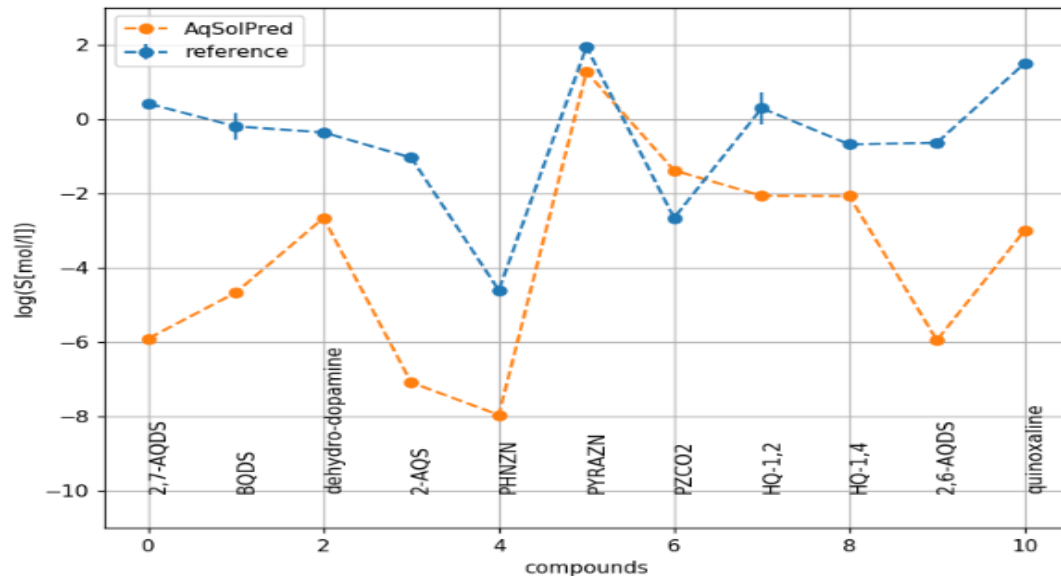


Ertl & Schuffenhauer 2009. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1, 1-11.

Solubility by AqSolPred v1.0

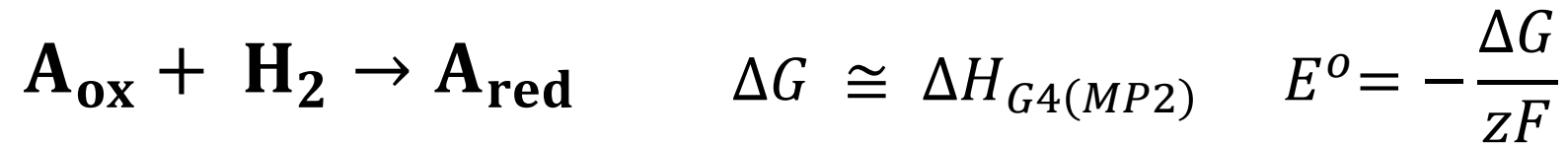
consensus model (Sorkun, Vianney, Koelman & Er 2021, Pushing the limits of solubility prediction via quality-oriented data selection' *science* 24.1 : 101961)

trained on ~10k experimental data (Sorkun, Khetan & Er. 2019, 'qSolDB, a curated reference set of aqueous solubility and 2D descriptors for a diverse set of compounds, *Scientific data* 6.1, 1-8)

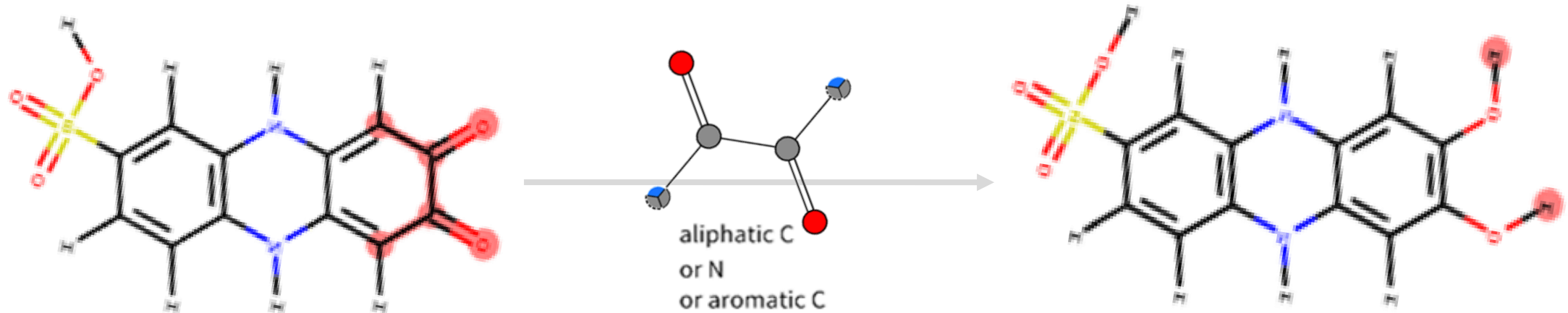


- *intrinsic* solubility prediction neglects (de)protonation processes
- oxidized states considered less soluble than reduced ones

Redox potential via ΔH



neutral, organic, closed shell
compounds in gas phase

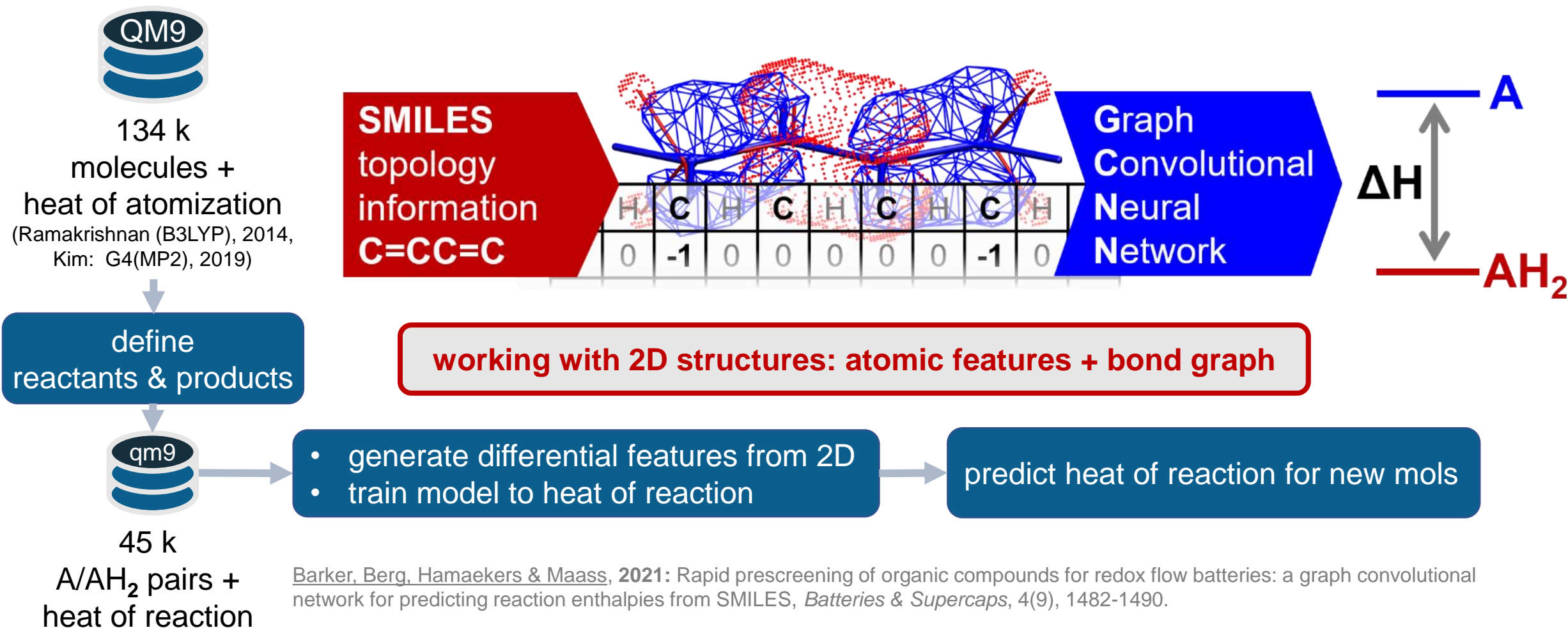


SMARTS reaction templates

[OX1:1]=[CX3:2](-,:[C,N,n,c:3])-,:[CX3:4](-,:[C,N,n,c:5])=[O:6].[H:7]-[H:8]
>> [H:7]-[O:1]-[C:2]([C,N,n,c:3])=[C:4]([C,N,n,c:5])-[O:6]-[H:8]

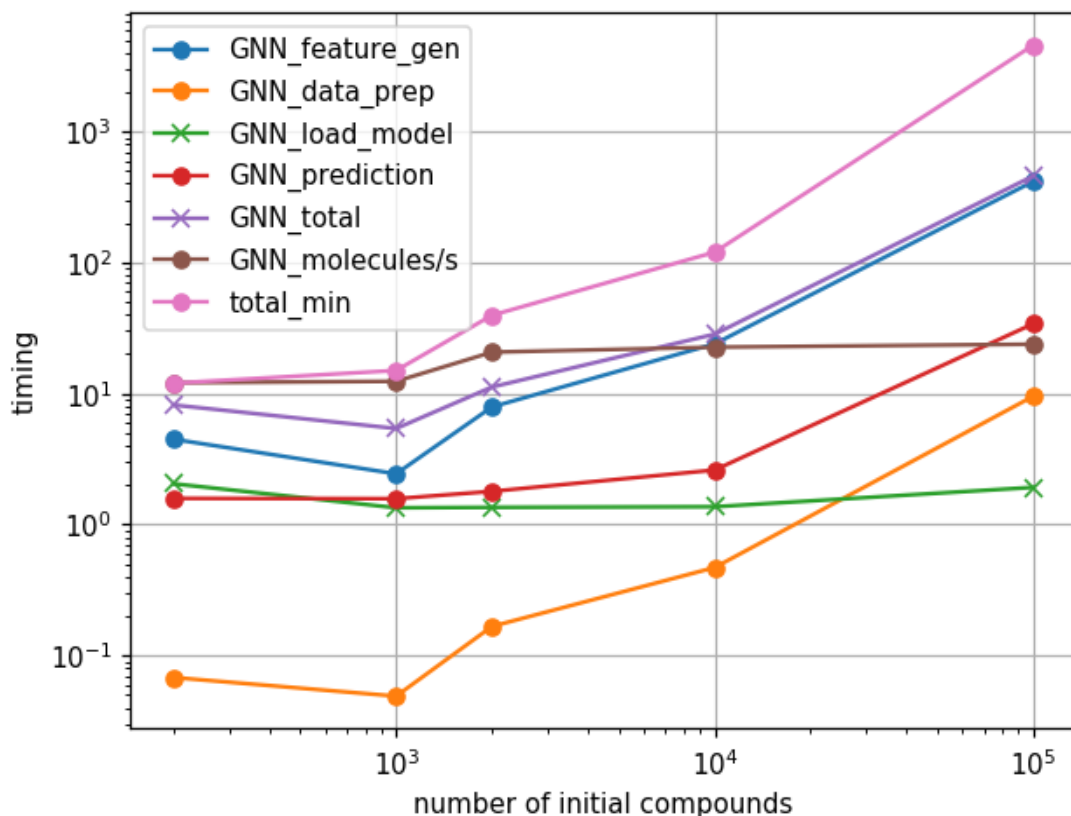
SMARTSviewer smartsview.zbh.uni-hamburg.de, ZBH Center for Bioinformatics, University of Hamburg

Predicting ΔH

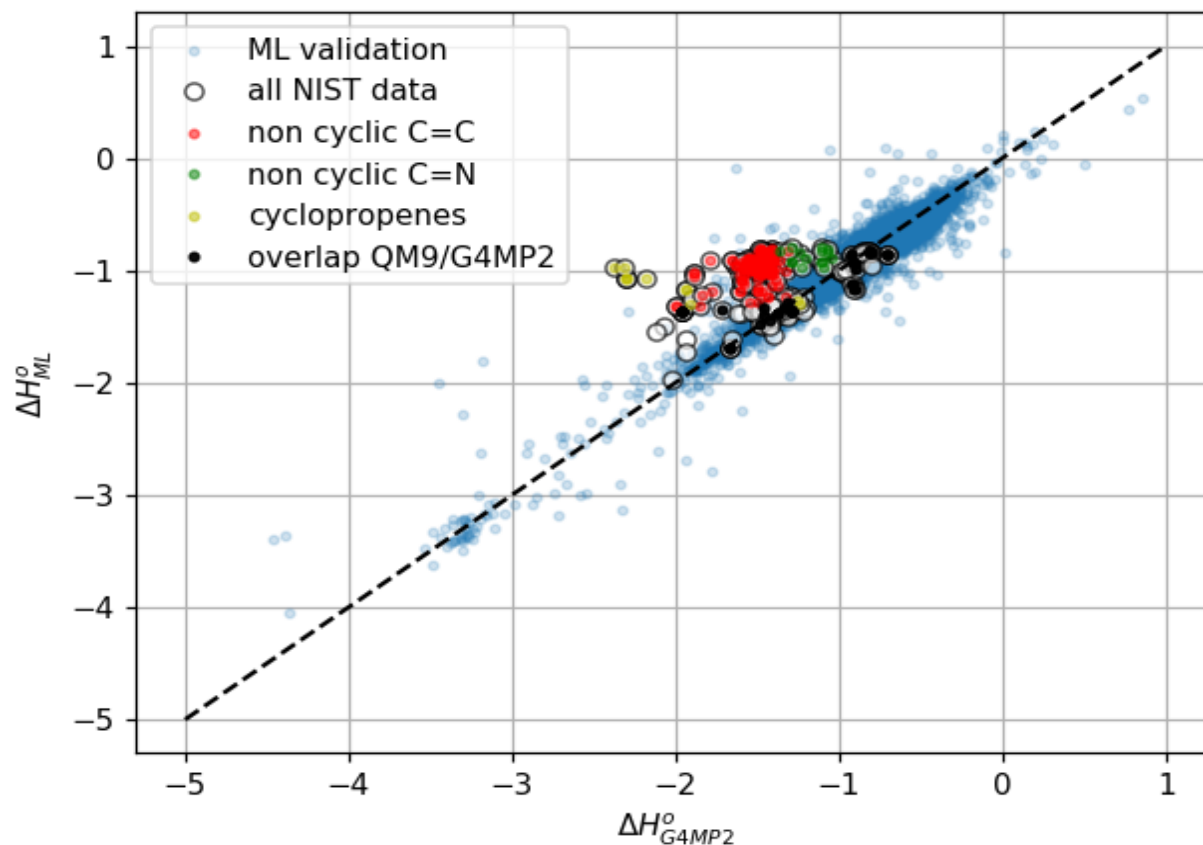


Barker, Berg, Hamaekers & Maass, 2021: Rapid prescreening of organic compounds for redox flow batteries: a graph convolutional network for predicting reaction enthalpies from SMILES, Batteries & Supercaps, 4(9), 1482-1490.

Model performance

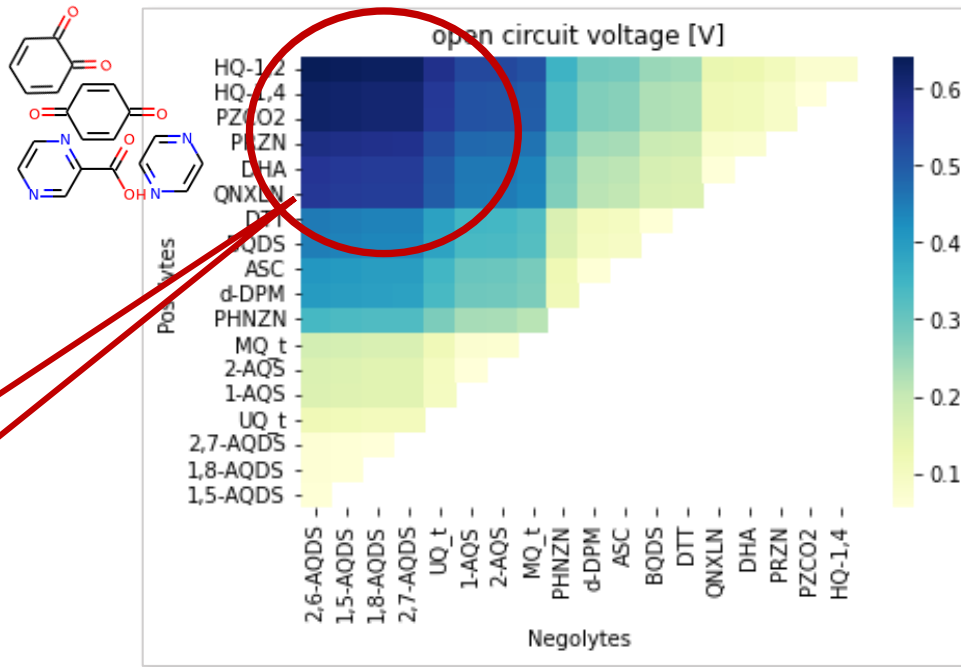


- 50 mols/sec – GNN feature generation most time consuming step
- 10e+5 compounds ≤ 2 hours on std desktop PC (4 cores, 8 GB memory)

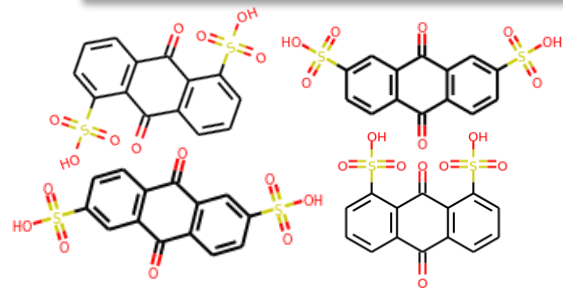


- limitation of the chemical space represented within the training set
→ provide larger training set & train again

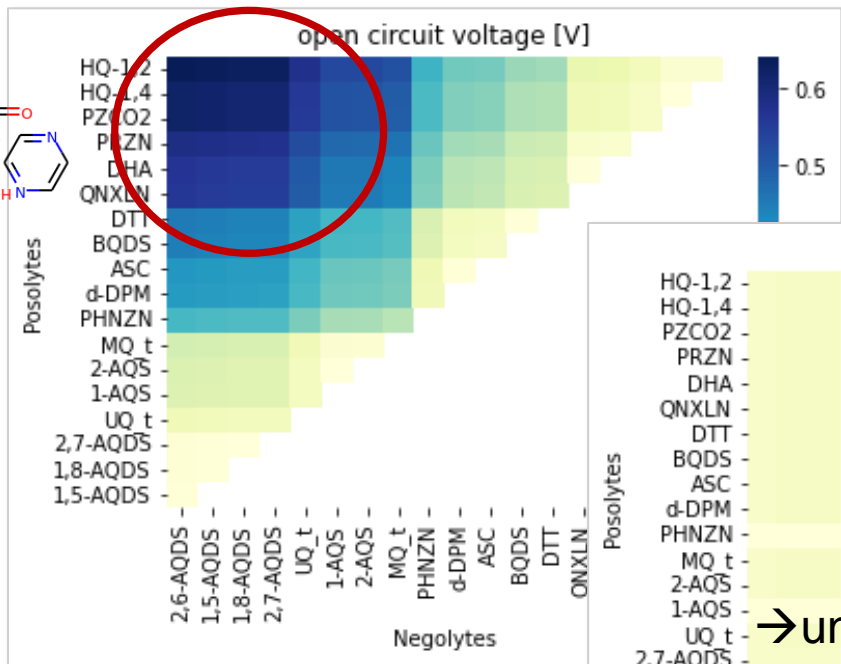
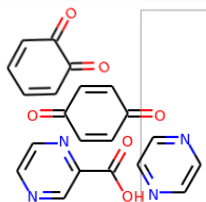
Full cell performance: OCV



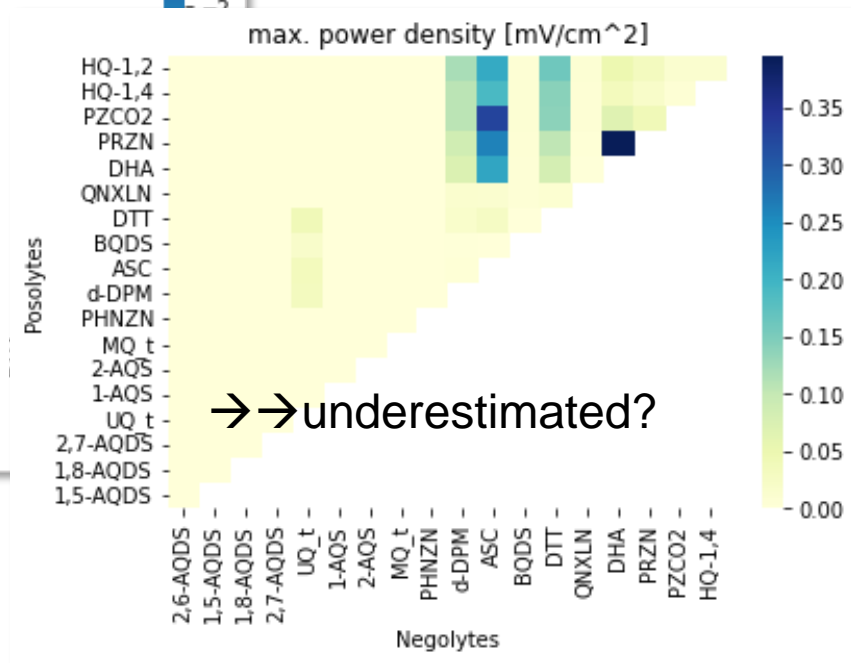
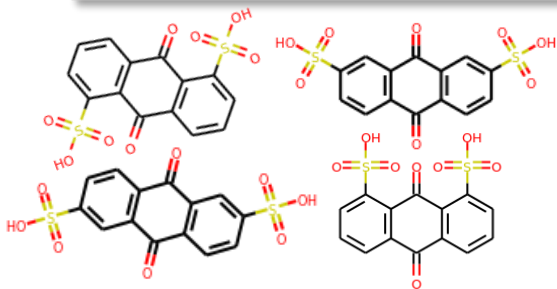
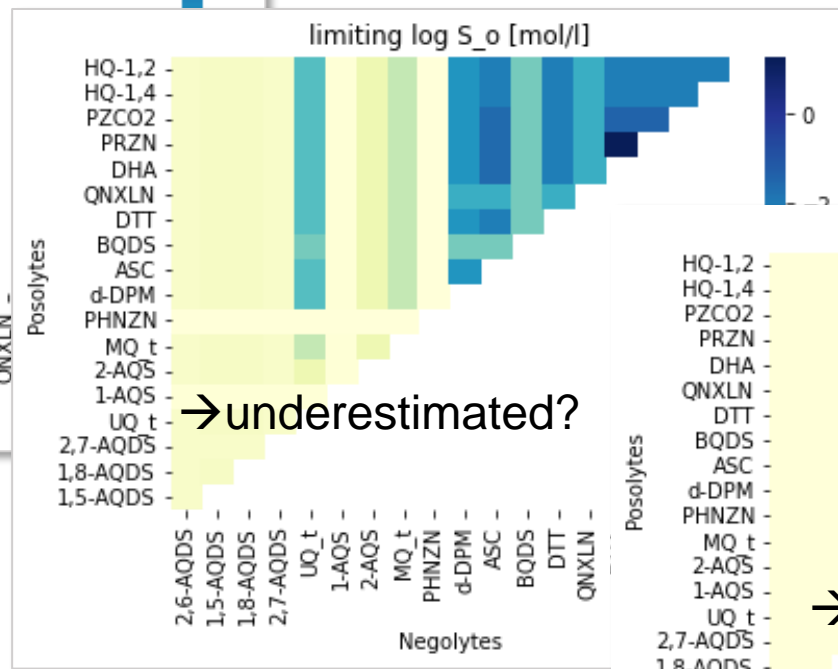
reasonable matches



Full cell performance



→ reasonable matches found
 → logS accuracy critical – consider pH



Generate compounds

File Edit Selection View Go Run Terminal Help

EXPLORER WP1

- ..pycache_
- data
- all.json
- chemiscope_gen.json 5
- demonstrator.ipynb 1
- fragment_creator.py
- gen_smiles.txt 6
- llama2-M-Full-RSS.pt
- merge_jsons.py
- model.py
- plot_utils.py
- preprocess_dataset.py
- sample.py
- test_1.json
- test_2.json
- test_3.json
- test_all.py
- test.json
- tokenizer.py
- train.py
- trainer.py

Code + Run All 2 Restart Clear All Outputs Variables Outline

```
import ipywidgets as widgets

display(context_smi_input)
display(temperature_slider)
display(generate_button)
display(output)
display(molecule_output)
```

logp: 3.00 sascore: 3 mol_weight: 3.00

Context SMI: C1=CSC=C1

Temperature: 0.80

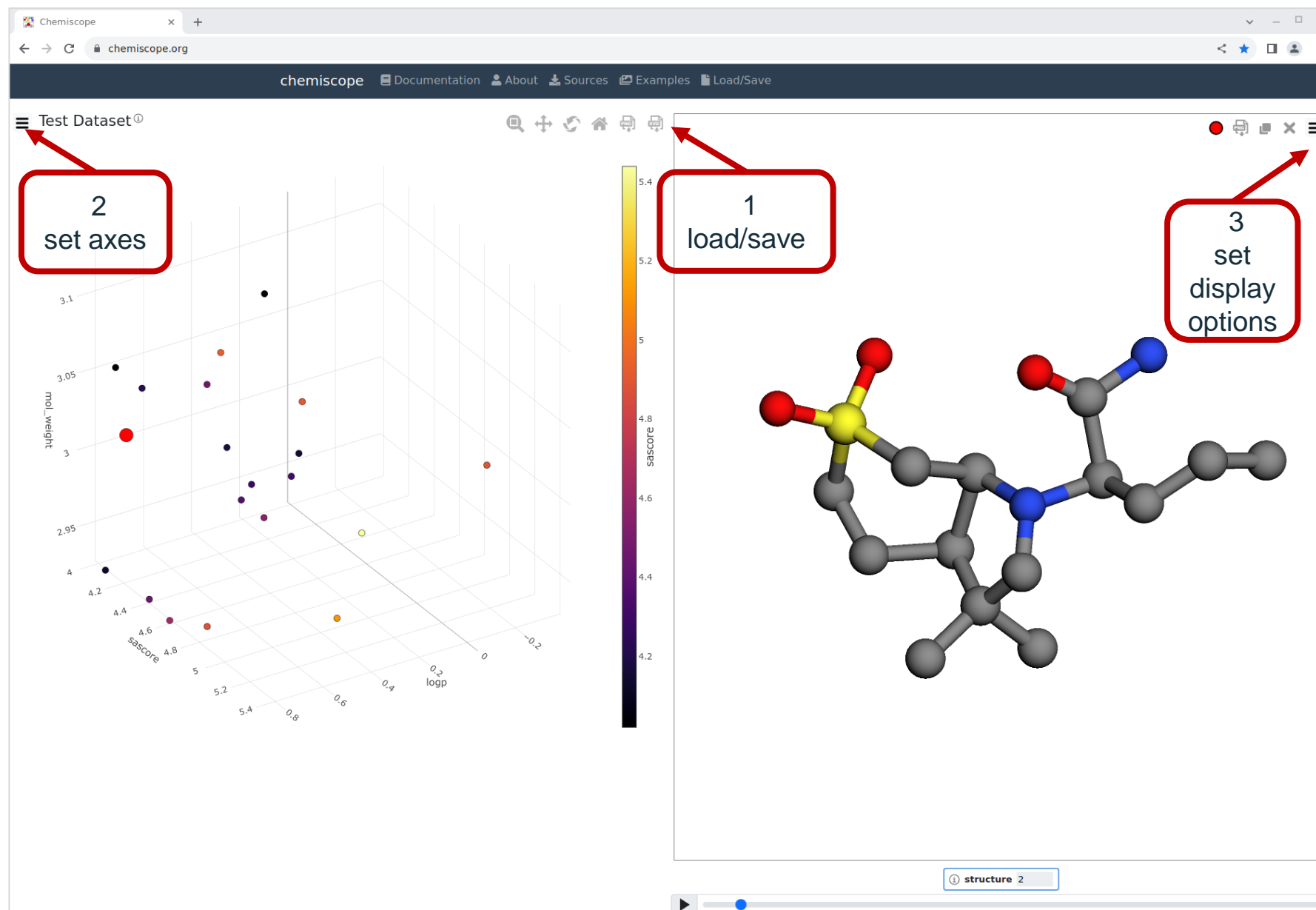
Generate 4

```
python merge_jsons.py first_experiment second_experiment ... all_experiments.json
```

TERMINAL (base) bash-4.4\$

- code .
- open *demonstrator.ipynb*
- >> Run All
- scroll down
- [specify targets]
- Generate**
- scroll down
- continue further analyses with output *chemiscope_gen.json* and/or *gen_smiles.txt*

View results



- open Tab Chemiscope in chromium-browser &
- upload *chemiscope_gen.json*

REDOX FOX

Prototype - Not ready for production use

Min. Solubility [mol / l] Min. OCV [V] Max. number of atoms

optional: set thresholds

Reaction Templates

```
[CX4,CX3:1]-[C:2]([CX4,CX3:3])=[O:4],[H:5]-[H:6]>>[CX4,CX3:1]-[C:2]([CX4,CX3:3])([H:5])-[O:4]-[H:6]
[CX4:1]-[C:2]([H])=[O:3],[H:4]-[H:5]>>[CX4:1]-[C:2]([H])([H:4])-[O:3]-[H:5]
[OX1:1]=[#OX3:2](-,[C,N,n,c:3])(-,[#OX3:4](-,[C,N,n,c:5])=[O:6],[H:7]-[H:8])>>[H:7]-[O:1]-[C:2]([C,N,n,c:3])=
[C:4]([C,N,n,c:5])-[O:6]-[H:8]
[#PX2:1]#-[#OX3:2](-,[#OX3:3]#-[#PX2:4],[H:5]-[H:6])>>[H:5]-[#PX3:1]-[#OX3:2]-[#OX3:3]-[#PX3:4]-[H:6]
[O:1]=[C:2]([C,N,n:9])-[C,c:3]#-[C,c:4]-[C:5]([C,N,n:10])=[O:6],[H:7]-[H:8]>>[H:7]-[O:1]-[C:2]([C,N,n:9])#-[
[C,c:3]#-[C,c:4]#-[C:5]([C,N,n:10])-[O:6]-[H:8]
[C:1]([O])-[C:2]([C:3]-[C:4])=[O:1],[H:5]-[H:6]>>[C:1]([O])-[C:2]([H:5])-[C:3]([H:6])-[C:4]([O])
```

ID
Smiles Ox
SAScore Ox
Smiles Red
SAScore Red

1. enter smiles

2. start calculations

START CALCULATION

DOWNLOAD

1. enter smiles

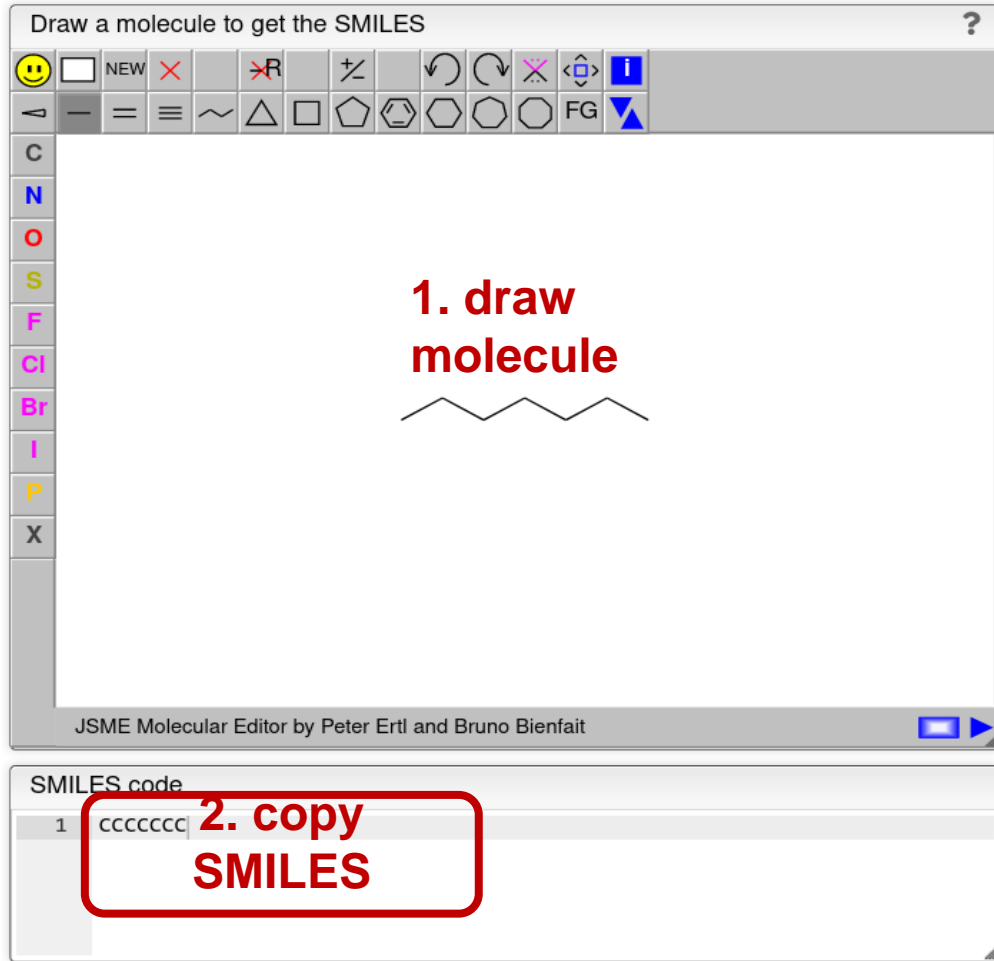
2. start calculations

- open Tab Redoxfox in chromium-browser & cat gen_smiles.txt
- copy & paste
- [apply filters?]
- start calculation
- scroll down
- choose OCV as Y-label
- click on point of interest in diagram → data row

Define target molecule/fragment

HOME UTILITIES ▾ DISPLAY DATA 3D MODELS

Draw a molecule to get the SMILES ?



1. draw molecule

SMILES code

1 CCCCCC 2. copy SMILES

JSME Molecular Editor by Peter Ertl and Bruno Bienfait

paste into

a) demonstrator

b) redoxfox

Play around...

3. edit/draw
compounds

website
SMILES generator

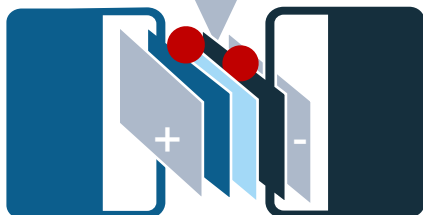
1. generate
compounds

notebook
demonstrator.ipynb



2. filter
compounds

website
redoxfox



Play around...

3. edit/draw
compounds

website
SMILES generator

1. generate
compounds

notebook
demonstrator.ipynb

2. filter
compounds

website
redoxfox

