Robust representation of small organic molecules with self-referencing embedded strings (SELFIES)

Florian Häse
Data Scientist – Computational Life Science

Bayer AG, Division Crop Science

Describing a chemical substance

Which representation is most useful?

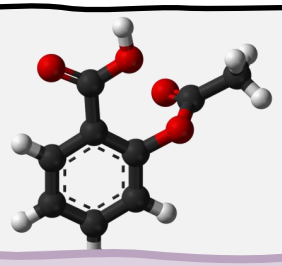
This white powder in ***that*** bottle!

Molar mass: **180.2 g/mol**
Density: **1.4 g/cm³**
Melting point: **136 °C**

This material with ***these*** properties!

CAS number: **50-78-2**
PubChem CID: **2244**
ChEMBL: **ChEMBL25**

This DB entry with ***these*** identifiers!

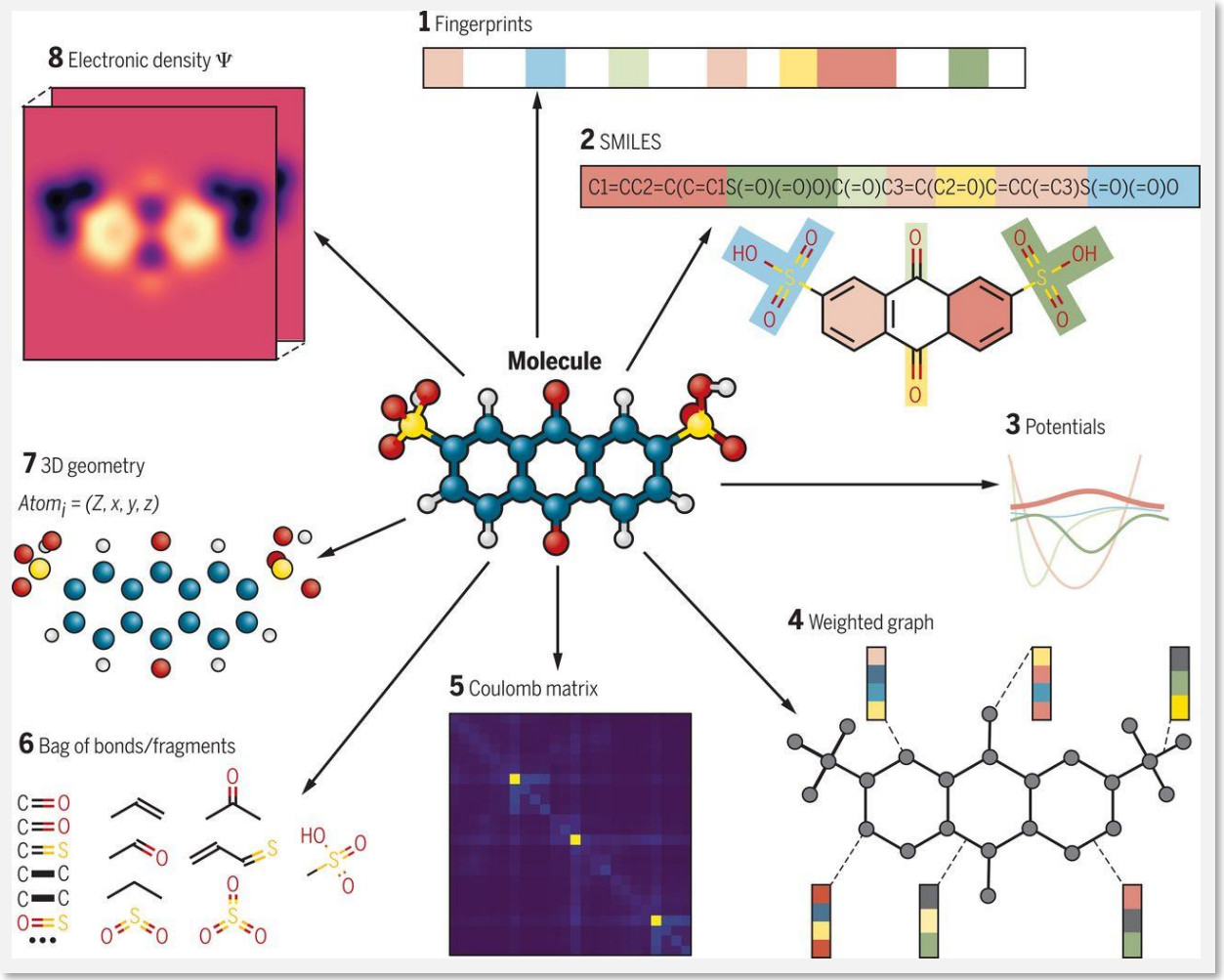


This chemical with ***this*** structure!

Different descriptions of the same chemical substance might be helpful in different situations.

Representations for molecular modeling

Which aspects of the molecular are most relevant to a given modeling task?



- Many different representations have been developed for molecular modeling tasks
- Each representation emphasizes certain aspects of a molecule, and obfuscates other aspects
- Every modeling task should start with the questions
  - which aspects about the molecule are fundamental to the task?
  - which representation highlights these aspects best?

Sanchez-Lengeling, Benjamin, and Alán Aspuru-Guzik. "Inverse molecular design using machine learning: Generative models for matter engineering." *Science* 361.6400 (2018): 360-365.

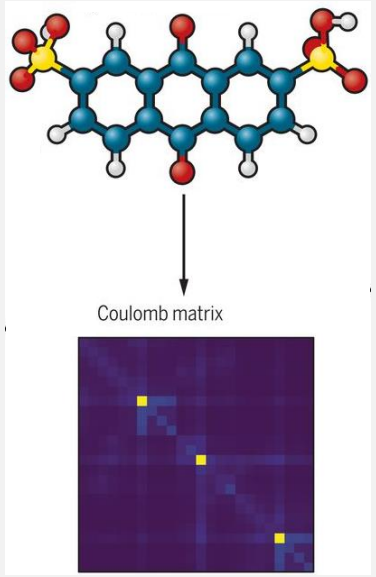
Example: Coulomb matrices to predict electronic properties

The **Coulomb Matrix** is a *global descriptor* based on the *pairwise electrostatic interaction* between nuclei,

$$M_{ij}^{\text{Coulomb}} = \begin{cases} Z_i^{2.4}/2 & \text{for } i = j \\ Z_i Z_j / R_{ij} & \text{for } i \neq j \end{cases}$$

where

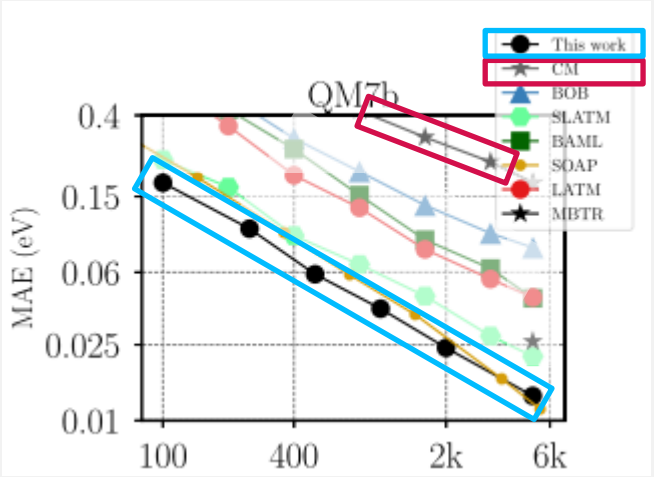
Z_i … nuclear charge
R_{ij} … interatomic distance



which

- uniquely encodes molecule structures
- satisfies translational and rotational invariance

Task:
Predict atomization/formation energies for QM7b



Multidimensional distributions of interatomic many-body expansions (Faber *et al.*) yield much more accurate prediction results

Better representations produce *better* results

Rupp, Matthias, et al. "Fast and accurate modeling of molecular atomization energies with machine learning." *Physical review letters* 108.5 (2012): 058301.
Faber, Felix A., et al. "Alchemical and structural distribution-based representation for universal quantum machine learning." *The Journal of chemical physics* 148.24 (2018): 241717.

Molecules as text

Sequential approaches to molecular structures

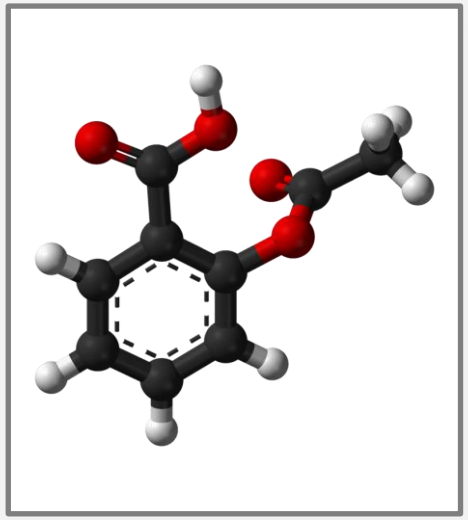
De-novo generation of small organic molecules

How can we generate molecule structures from scratch?

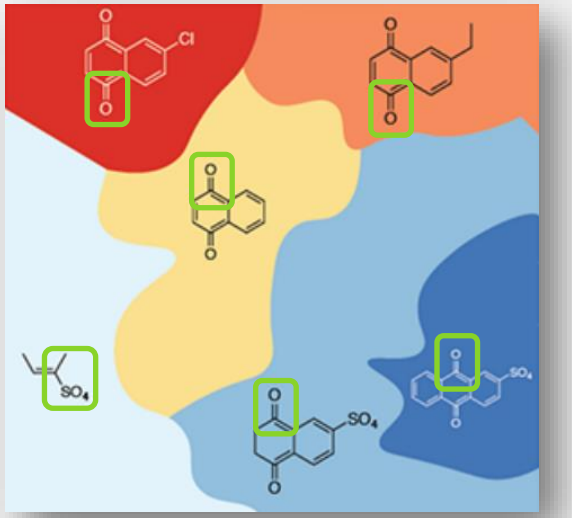
Goals:

- machine-generated molecule structures
- which satisfy basic chemical constraints

"valid" molecule



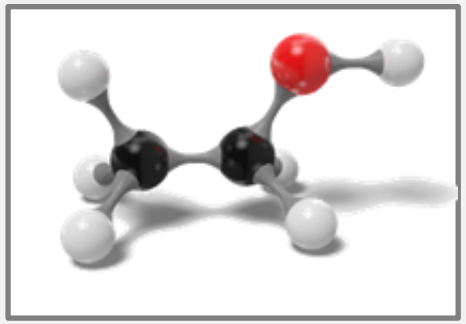
"invalid" molecules



Idea:

- get inspired by educational toys – ***chemistry sets***!
- accept the constraints of *your* scientific kit
- only attach those atoms that fit the current structure

Example:



Algorithm (ignoring H):

- Take one carbon (C)
- Take another carbon (C)
- Take one oxygen (O)

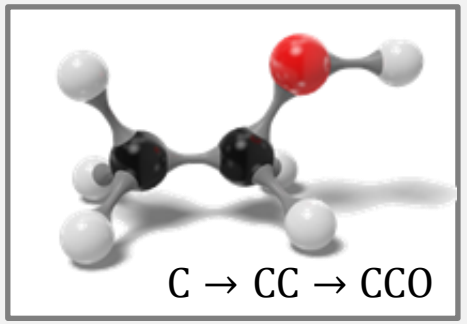
$C \rightarrow CC \rightarrow CCO$

Molecule can be "written as a string"

Formalizing the generation of molecular structures

Which set of rules enables the sequential generation of molecular graphs?

Generation process
is governed by **rules**
and **representations**



Formal grammars
are composed of

- terminal symbols ***T***
- nonterminal symbols ***N***
- production rules ***P***
- start symbols ***S***

Example

T = {**C**, **O**}

N = {*[molecule]*}

S = {*[molecule]*}

Production rules:

- **(i)** [molecule] → >> empty <<
- **(ii)** [molecule] → **C** [molecule]
- **(iii)** [molecule] → **O** [molecule]

Execution of, e.g., [(ii), (ii), (iii), (i)] leads to

[molecule] -> **C [molecule]** -> **C C [molecule]** -> **C C O [molecule]** -> **C C O**

Observations:

- This grammar generates, e.g., alkanes and alcohols
  CCO, **CCCC**, **OCCC**, **C**
- This grammar cannot generate all molecules
  CNO is a syntactical mistake (N is not part of the vocabulary)
- Some of the generated strings are not molecules
  OOOO is not a valid molecule

Short interlude:

Simplified **M**olecular **I**nput **L**ine **E**ntry **S**ystem *(SMILES)*

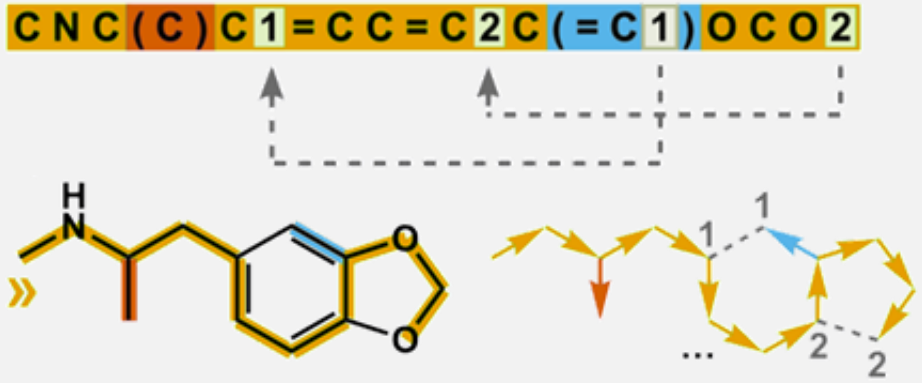
Excursion to SMILES strings

What is the *de-facto* standard for text-based molecular representations?

SMILES:
Simplified **M**olecular **I**nput **L**ine **E**ntry **S**ystem

Atoms: C, O, F, N, P, S, …
Bonds: =, #, :, -
Branches: (,)
Rings: 1, 2, 3, ...

Example: 3,4-methylenedioxymethamphetamine (MDMA)



Benefits

- capable of representing molecules as strings
- molecular representations are intuitive

Disadvantage

- arbitrary sequences of SMILES characters do not necessarily encode valid molecules

Examples: *Syntactic* violations

CC1CO1CCC2CCC	Ring not closed
CC(CCO(CCC)CC	Branch not closed

Examples: *Semantic* violations

C=C=O=CCOC	Incorrect valency
C=CCCFOC	Incorrect valency

Weininger, David. "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules." *Journal of chemical information and computer sciences* 28.1 (1988): 31-36.

Self-referencing embedded strings *(SELFIES)*

Requirements on grammars for sequential molecular generation

Which properties does a grammar need to satisfy to enable sequential molecular generation?

Why did we fail **syntactically**?

CC1CO1CCC**2**CCC	Ring not closed
CC**(**CCO(CCC)CC	Branch not closed

Some generated symbols cannot be interpreted in the given context

Mitigation: Grammar must be *context-free*

Why did we fail **semantically**?

C=C**=O=**CCOC	Incorrect valency
C=CCC**F**OC	Incorrect valency

Some generated symbols do not satisfy physical / chemical constraints

Mitigation: Grammar must account for *chemistry*

Categorizing formal grammars (Chomsky)

Class	Grammars	Automaton
Type 0	unrestricted	Turing machine
Type 1	context sensitive	Linear-Bound
Type 2	context free	Pushdown
Type 3	regular	Finite

Chomsky Type 2 Grammar

- uses non-terminal symbols Z and terminal symbols t
- uses production rules of the form $Z \rightarrow t\ Z$
- can apply production rules regardless of the context
- but does - *a priori* - not produce chemically relevant sequences

Hopcroft, John E., Rajeev Motwani, and Jeffrey D. Ullman. "Automata theory, languages, and computation." *International Edition* 24.2 (2006).

Simple Chomsky type-2 grammar to generate FC=C=N

How do we design Chomsky type-2 grammars for molecular generation?

Simple grammar

Terminals

T = { F, C, =N, =C }

Non-terminals

$N = \{Z_0, Z_1, Z_2, Z_3\}$

Start symbol

$S = \{Z_0\}$

Production rules

// [F]: $Z_0 \rightarrow \mathbf{F}\ Z_1$

// [C]: $Z_1 \rightarrow \mathbf{C}\ Z_3$

// [N]: $Z_2 \rightarrow =\mathbf{N}$

// [C]: $Z_3 \rightarrow =\mathbf{C}\ Z_2$

Pushdown automaton generates string based on collection of specific production rules

State of derivation	$[F]$	$[C]$	$[N]$
Z_0			
Z_1	FZ_1		
Z_2		CZ_3	
Z_3			=N
		=CZ_2	

Example: Consider the collection of production rules

[F] [C] [C] [N]

Starting from Z_0, the automaton generates:

$$Z_0 \xrightarrow{[F]} \mathbf{F}\ Z_1 \xrightarrow{[C]} \mathbf{F\ C}\ Z_3 \xrightarrow{[C]} \mathbf{F\ C =C}\ Z_2 \xrightarrow{[N]} \mathbf{F\ C =C =N}$$

Keeping track of the current state, the pushdown automaton will only generate chemically valid structures if the production rules satisfy these constraints

Introducing SELFIES

Which features do we need to generate molecular graphs?

We require a set of decoding rules to reliably write (molecular) graphs from character-based sequences

- writing ***vertices:*** non-terminal Z_r and rule A_n ***generates*** vertex $t_{r,n}$ and non-terminal $Z_{h_{r,n}}$
- writing ***rings:*** non-terminal Z_r and rule A_{n+m} ***closes*** N-membered ring and non-terminal $Z_{i_{r,n}}$
- writing ***branches:*** non-terminal Z_r and rule A_{n+n+p} ***opens*** N-membered branch and non-terminal $Z_{h_{r,n}}$

State of derivation	A_0	A_1	A_n	A_{n+1}	A_{n+m}	A_{n+m+1}	A_{n+m+p}
Z_0	ε	$t_{0,1}Z_{h_{0,1}}$	$t_{0,n}Z_{h_{0,n}}$	$R(N)Z_{i_{0,1}}$	$R(N)Z_{i_{0,m}}$	$B(N,Z_{j_{0,1}})Z_{k_{0,1}}$	$B(N,Z_{j_{0,p}})Z_{k_{0,p}}$
Z_1	ε	$t_{1,1}Z_{h_{1,1}}$	$t_{1,n}Z_{h_{1,n}}$	$R(N)Z_{i_{1,1}}$	$R(N)Z_{i_{1,m}}$	$B(N,Z_{j_{1,1}})Z_{k_{1,1}}$	$B(N,Z_{j_{1,p}})Z_{k_{1,p}}$
Z_r	ε	$t_{r,1}Z_{h_{r,1}}$	$t_{r,n}Z_{h_{r,n}}$	$R(N)Z_{i_{r,1}}$	$R(N)Z_{i_{r,m}}$	$B(N,Z_{j_{r,1}})Z_{k_{r,1}}$	$B(N,Z_{j_{r,p}})Z_{k_{r,p}}$
N	0	1	n	$n+1$	$n+m$	$n+m+1$	$n+m+p$

However: These decoding rules do not account for ***chemical / physical semantics*** (yet)!

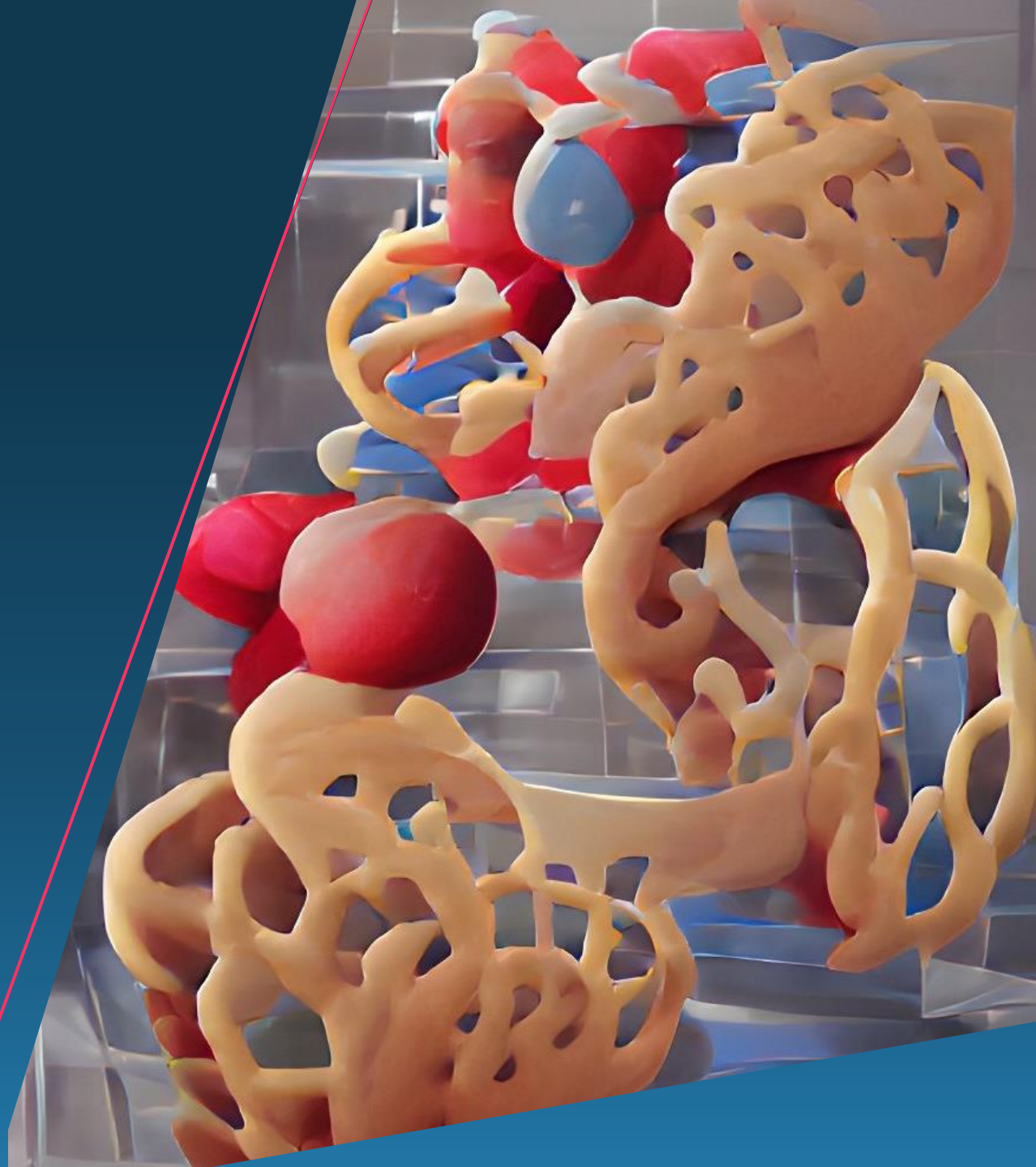
Introducing SELFIES

Which features do we need to generate ***plausible*** molecular graphs?

	$[\varepsilon]$	$[F]$	$[=O]$	$[\#N]$	$[O]$	$[N]$	$[=N]$	$[C]$	$[=C]$	$[\#C]$	$[Ring]$	[Branch1]	[Branch2]	[Branch3]
Z_0	Z_0	FZ_1	OZ_2	NZ_3	OZ_2	NZ_3	NZ_3	CZ_4	CZ_4	CZ_4	ign Z_0	ign Z_0	ign Z_0	ign Z_0
Z_1	ε	F	O	N	OZ_1	NZ_2	NZ_2	CZ_3	CZ_3	CZ_3	$R(N)$	ign Z_1	ign Z_1	ign Z_1
Z_2	ε	F	=O	=N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	=CZ_2	$R(N)Z_1$	$B(N,Z_5)Z_1$	$B(N,Z_5)Z_1$	$B(N,Z_5)Z_1$
Z_3	ε	F	=O	#N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	#CZ_1	$R(N)Z_2$	$B(N,Z_5)Z_2$	$B(N,Z_6)Z_1$	$B(N,Z_5)Z_2$
Z_4	ε	F	=O	#N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	#CZ_1	$R(N)Z_3$	$B(N,Z_5)Z_3$	$B(N,Z_7)Z_1$	$B(N,Z_6)Z_3$
Z_5	C	F	O	N	OZ_1	NZ_2	NZ_2	CZ_3	CZ_3	CZ_3	Z_5	Z_5	Z_5	Z_5
Z_6	C	F	=O	=N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	=CZ_2	Z_6	Z_6	Z_6	Z_6
Z_7	C	F	=O	#N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	#CZ_1	Z_7	Z_7	Z_7	Z_7
N	1	2	3	4	5	6	7	8	9	10	11	12	13	14

// This set of decoding rules covers the generation of all ***non-ionic*** molecules of the ***QM9*** dataset

Hands-on exercise with SELFIES



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

	$[\varepsilon]$	$[F]$	$[=O]$	$[\#N]$	$[O]$	$[N]$	$[=N]$	$[C]$	$[=C]$	$[\#C]$	$[Ring]$	[Branch1]	[Branch2]	[Branch3]
Z_0	Z_0	FZ_1	OZ_2	NZ_3	OZ_2	NZ_3	NZ_3	CZ_4	CZ_4	CZ_4	ign Z_0	ign Z_0	ign Z_0	ign Z_0
Z_1	ε	F	O	N	OZ_1	NZ_2	NZ_2	CZ_3	CZ_3	CZ_3	$R(N)$	ign Z_1	ign Z_1	ign Z_1
Z_2	ε	F	=O	=N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	=CZ_2	$R(N)Z_1$	$B(N,Z_5)Z_1$	$B(N,Z_5)Z_1$	$B(N,Z_5)Z_1$
Z_3	ε	F	=O	#N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	#CZ_1	$R(N)Z_2$	$B(N,Z_5)Z_2$	$B(N,Z_6)Z_1$	$B(N,Z_5)Z_2$
Z_4	ε	F	=O	#N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	#CZ_1	$R(N)Z_3$	$B(N,Z_5)Z_3$	$B(N,Z_7)Z_1$	$B(N,Z_6)Z_3$
Z_5	C	F	O	N	OZ_1	NZ_2	NZ_2	CZ_3	CZ_3	CZ_3	Z_5	Z_5	Z_5	Z_5
Z_6	C	F	=O	=N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	=CZ_2	Z_6	Z_6	Z_6	Z_6
Z_7	C	F	=O	#N	OZ_1	NZ_2	=NZ_1	CZ_3	=CZ_2	#CZ_1	Z_7	Z_7	Z_7	Z_7
N	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Task: Decode the given SELFIES string using (a subset of) the provided set of decoding rules!

Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_0

Current rule symbol: **[O]**

Decoding rule:

[O] → **O**Z_2

Decoded string:

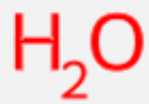
OZ_2

Decoded SMILES

O

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_2

Current rule symbol: [=C]

Decoding rule:

[=C] → =CZ_2

Decoded string:

O=CZ_2

Decoded SMILES

O=C

	[ε]	[$= O$]	[O]	[N]	[C]	[$= C$]	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	=CZ_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule

Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] **[Branch2]** [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_2

Current rule symbol: **[Branch2]**

Decoding rule:

[Branch2] → $B(N, Z_5)Z_1$

Decoded string:

O=C $B(N, Z_5)Z_1$

Decoded SMILES

O=C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N, Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N, Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule

Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] **[eps]** [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: N

Current rule symbol: **[eps]**

Decoding rule:

[eps] → 1

Decoded string:

O=C $B(1, Z_5)Z_1$

Decoded SMILES

O=C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N, Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N, Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule

Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] **[C]** [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_5

Current rule symbol: **[C]**

Decoding rule:

[C] → CZ_3

Decoded string:

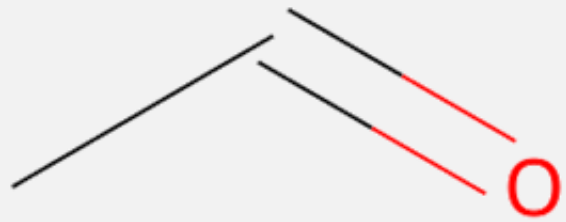
O=C $B(1,$**C**$Z_3)$ Z_1

Decoded SMILES

O=C(C)

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] **[O]** [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_1

Current rule symbol: **[O]**

Decoding rule:

[O] → **O**Z_1

Decoded string:

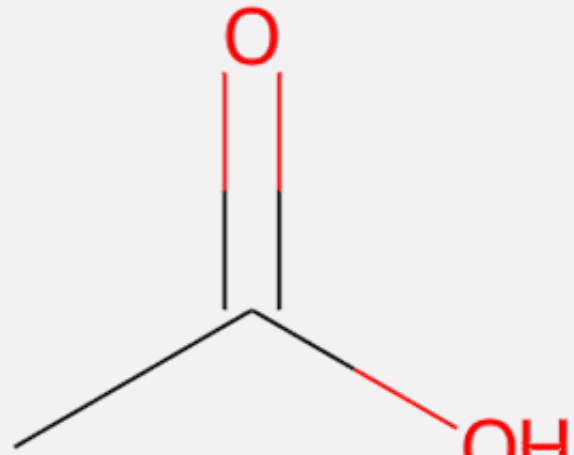
O=C $B(1,$**C**$Z_3)$ **O**Z_1

Decoded SMILES

O=C(C)O

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] **[C]** [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_1

Current rule symbol: **[C]**

Decoding rule:

[C] → CZ_3

Decoded string:

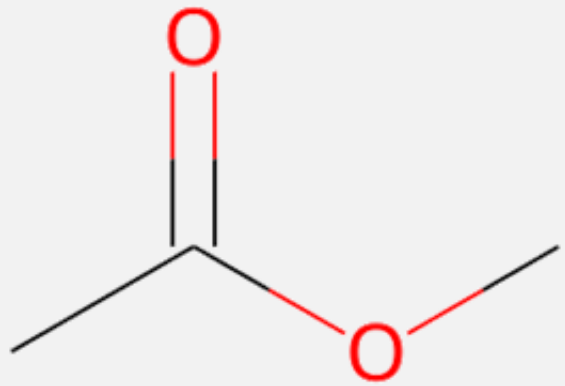
O=C $B(1,\mathbf{C}Z_3)$ **O**$\mathbf{C}Z_3$

Decoded SMILES

O=C(C)OC

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] **[=C]** [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_3

Current rule symbol: **[=C]**

Decoding rule:

[=C] → **=C**Z_2

Decoded string:

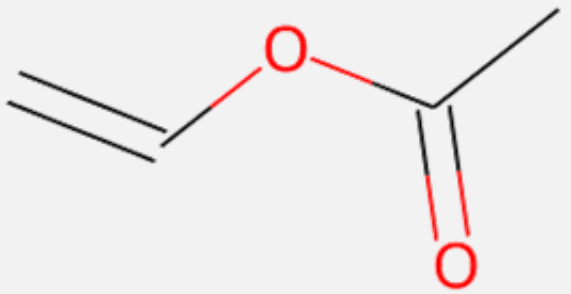
O=C $B(1,$**C**$Z_3)$ **OC=C**Z_2

Decoded SMILES

O=C(C)OC=C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] **[C]** [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_2

Current rule symbol: **[C]**

Decoding rule:

[C] → CZ_3

Decoded string:

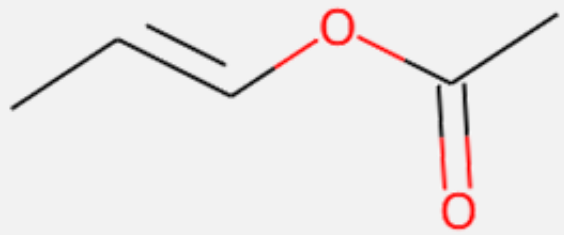
O=C $B(1,$**C**$Z_3)$ **OC=CC**Z_3

Decoded SMILES

O=C(C)OC=CC

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] **[=C]** [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_3

Current rule symbol: **[=C]**

Decoding rule:

[=C] → **=C**Z_2

Decoded string:

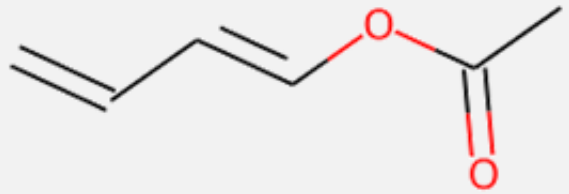
O=C $B(1,$**C**$Z_3)$ **OC=CC=C**Z_2

Decoded SMILES

O=C(C)OC=CC=C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] **[C]** [=C] [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_2

Current rule symbol: **[C]**

Decoding rule:

[C] → CZ_3

Decoded string:

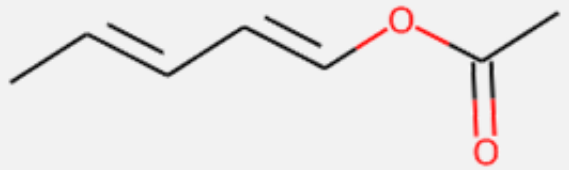
O=C $B(1,$**C**$Z_3)$ **OC=CC=C**$\mathbf{C}Z_3$

Decoded SMILES

O=C(C)OC=CC=CC

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] **[=C]** [Ring] [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_3

Current rule symbol: **[=C]**

Decoding rule:

[=C] → **=C**Z_2

Decoded string:

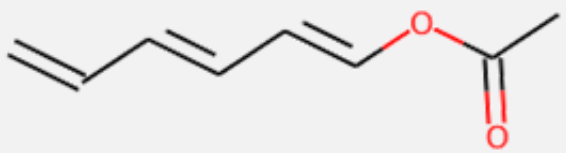
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C**Z_2

Decoded SMILES

O=C(C)OC=CC=CC=C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] **[Ring]** [N] [C] [Branch2] [eps] [=O] [O]

Current state: Z_2

Current rule symbol: **[Ring]**

Decoding rule:

[Ring] → $R(N)Z_2$

Decoded string:

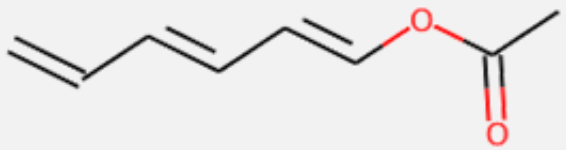
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $R(N)Z_2$

Decoded SMILES

O=C(C)OC=CC=CC=C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] **[N]** [C] [Branch2] [eps] [=O] [O]

Current state: N

Current rule symbol: **[N]**

Decoding rule:

[N] → 6

Decoded string:

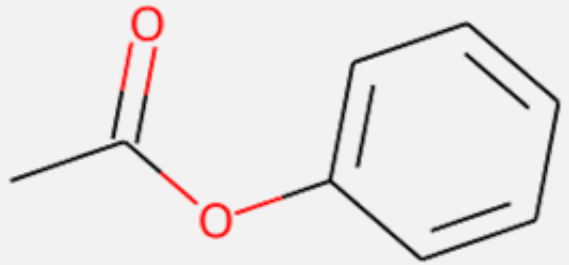
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $R(6)Z_2$

Decoded SMILES

O=C(C)OC1=CC=CC=C1

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] **[C]** [Branch2] [eps] [=O] [O]

Current state: Z_2

Current rule symbol: **[C]**

Decoding rule:

[C] → CZ_3

Decoded string:

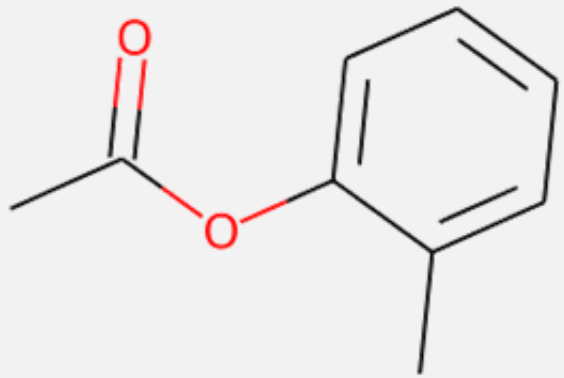
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $\boldsymbol{R}(6)$ **C**Z_3

Decoded SMILES

O=C(C)OC1=CC=CC=C1C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] **[Branch2]** [eps] [=O] [O]

Current state: Z_3

Current rule symbol: **[Branch2]**

Decoding rule:

[Branch2] → $B(N, Z_6)Z_1$

Decoded string:

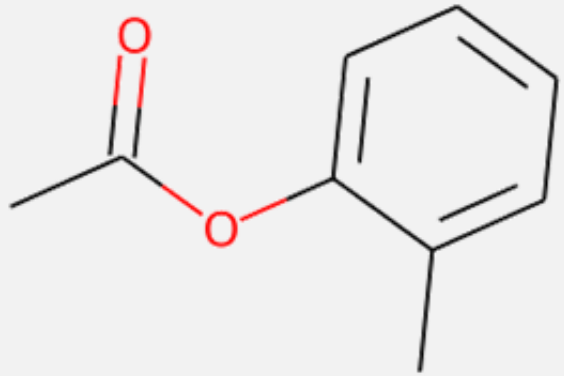
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $\boldsymbol{R}(6)$ **C** $B(N, Z_6)Z_1$

Decoded SMILES

O=C(C)OC1=CC=CC=C1C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N, Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N, Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] **[eps]** [=O] [O]

Current state: N

Current rule symbol: **[eps]**

Decoding rule:

[eps] → 1

Decoded string:

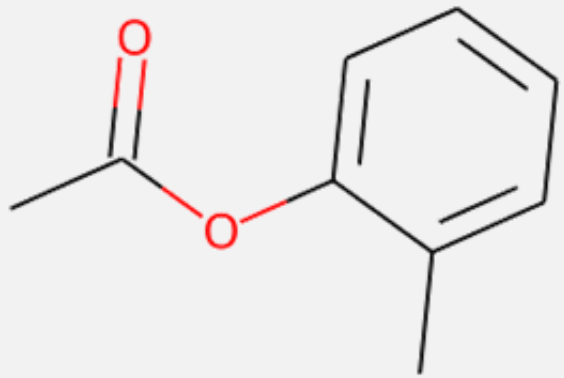
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $\boldsymbol{R}(6)$ **C** $B(1,Z_6)Z_1$

Decoded SMILES

O=C(C)OC1=CC=CC=C1C

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	=O	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] **[=O]** [O]

Current state: Z_6

Current rule symbol: **[=O]**

Decoding rule:

[=O] → O

Decoded string:

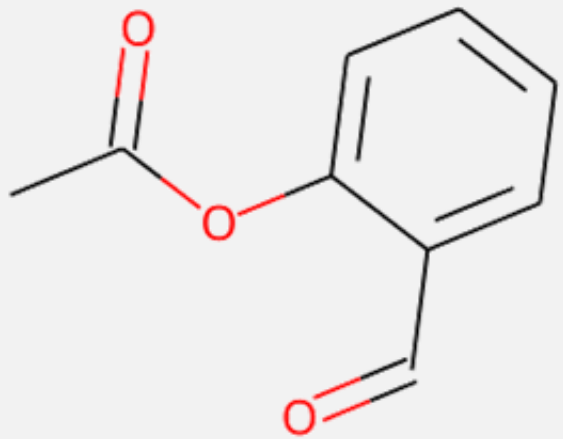
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $\boldsymbol{R}(6)$ **C** $B(1,$**O**$)Z_1$

Decoded SMILES

O=C(C)OC1=CC=CC=C1C(=O)

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

Decoded molecule



Exercise - Decoding a SELFIES string (with selfies v0.1.1)

[O] [=C] [Branch2] [eps] [C] [O] [C] [=C] [C] [=C] [C] [=C] [Ring] [N] [C] [Branch2] [eps] [=O] **[O]**

Current state: Z_1

Current rule symbol: **[O]**

Decoding rule:

[O] → **O**Z_1

Decoded string:

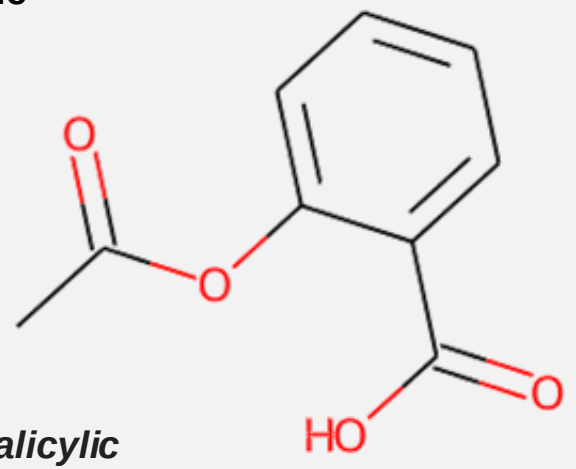
O=C $B(1,$**C**$Z_3)$ **OC=CC=CC=C** $\boldsymbol{R}(6)$ **C** $B(1,$**O**$)$ **O**Z_1

Decoded SMILES

O=C(C)OC1=CC=CC=C1C(=O)O

	$[\varepsilon]$	$[=O]$	$[O]$	$[N]$	$[C]$	$[=C]$	[Ring]	[Branch2]
Z_0	Z_0	OZ_2	**O**Z_2	NZ_3	CZ_4	CZ_4	$R(N)$	ign Z_0
Z_1	ε	O	**O**Z_1	NZ_2	**C**Z_3	CZ_3	$R(N)$ Z_1	ign Z_1
Z_2	ε	=O	**O**Z_1	NZ_2	**C**Z_3	**=C**Z_2	$R(N)Z_2$	$B(N,Z_5)Z_1$
Z_3	ε	=O	OZ_1	NZ_2	CZ_3	**=C**Z_2	$R(N)$ Z_3	$B(N,Z_6)Z_1$
Z_5	ε	O	OZ_1	NZ_2	**C**Z_3	CZ_3	Z_5	Z_5
Z_6	ε	**=O**	OZ_1	NZ_2	CZ_3	=CZ_2	Z_5	Z_6
N	1	3	5	6	8	9	11	13

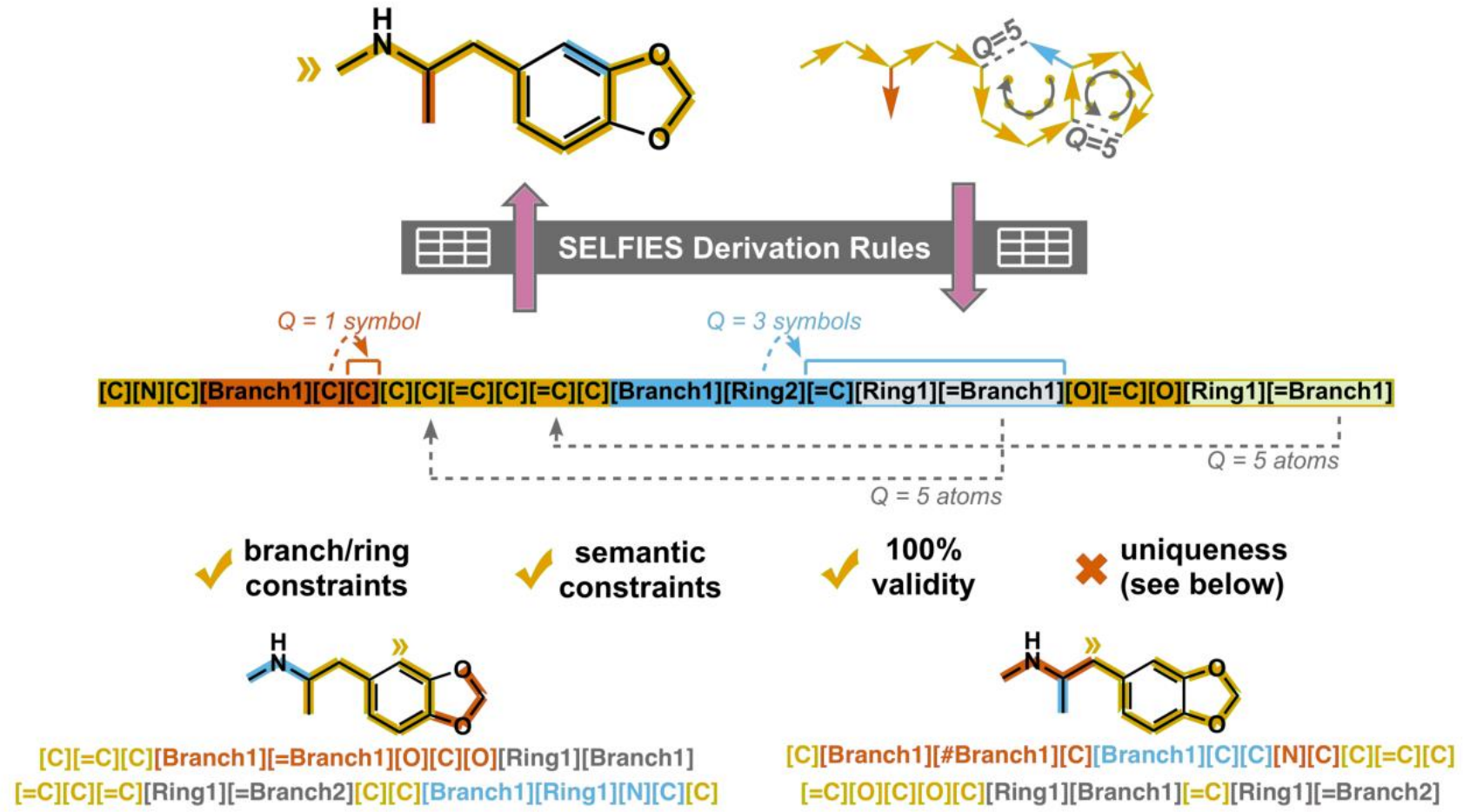
Decoded molecule



acetylsalicylic acid

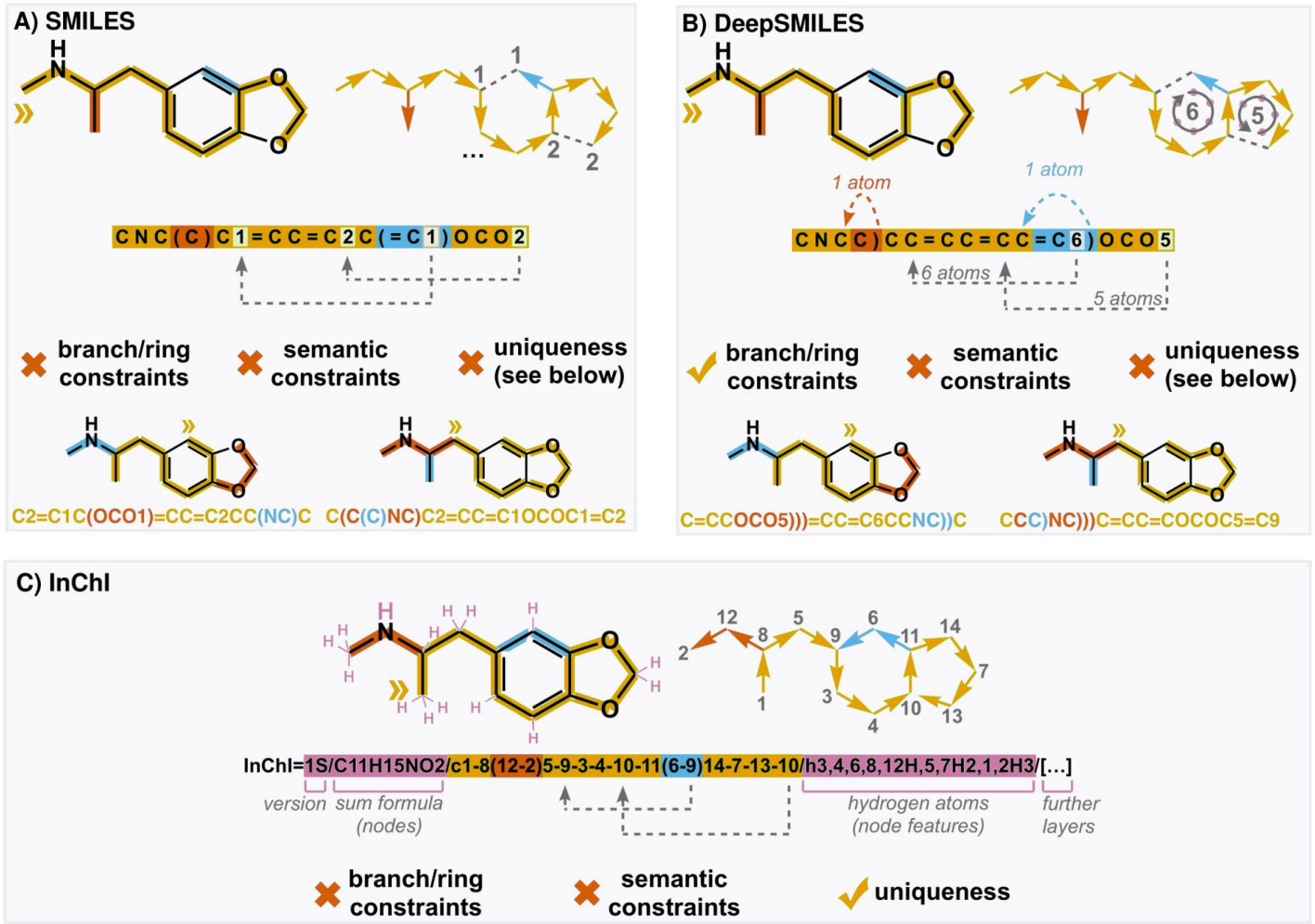
Comparing SELFIES to other text-based representations

Summarizing the capabilities and limitations of SELFIES



Krenn, Mario, et al. "SELFIES and the future of molecular string representations." *arXiv preprint arXiv:2204.00056* (2022).

SELFIES in comparison to other text-based representations



SMILES offer

- ✓ **human readability**

InChI offer

- ✓ **uniqueness**

DeepSMILES offer

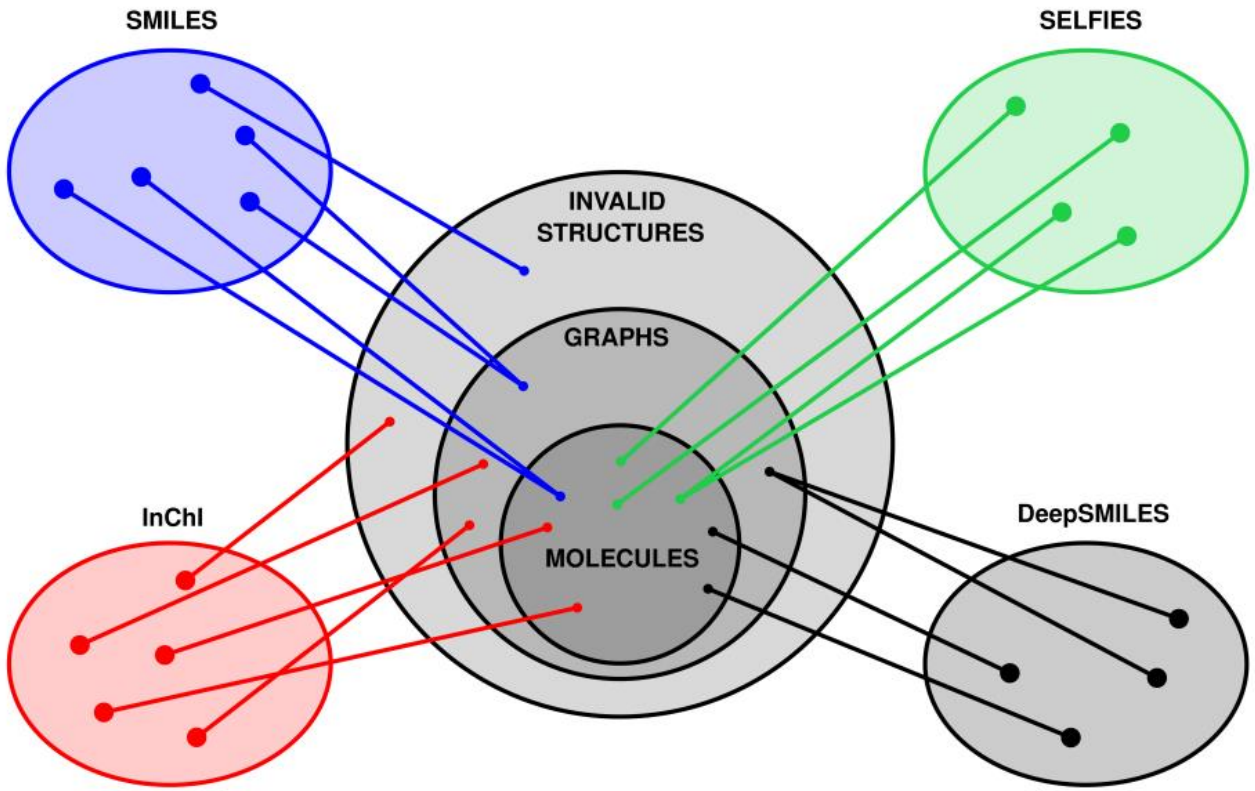
- ✓ **human readability**
- ✓ **branch / ring constraints**

SELFIES offers

- ✓ **100 % validity**
- ✓ **branch / ring constraints**
- ✓ **semantic constraints**

Krenn, Mario, et al. "SELFIES and the future of molecular string representations." *arXiv preprint arXiv:2204.00056* (2022).

SELFIES in comparison to other text-based representations



SMILES offer

- ✓ **human readability**

InChI offer

- ✓ **uniqueness**

DeepSMILES offer

- ✓ **human readability**
- ✓ **branch / ring constraints**

SELFIES offers

- ✓ **100 % validity**
- ✓ **branch / ring constraints**
- ✓ **semantic constraints**

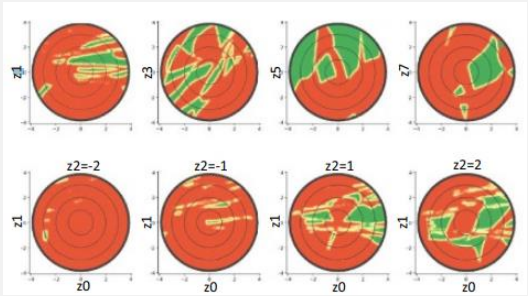
Krenn, Mario, et al. "SELFIES and the future of molecular string representations." *arXiv preprint arXiv:2204.00056* (2022).

Modeling opportunities with SELFIES

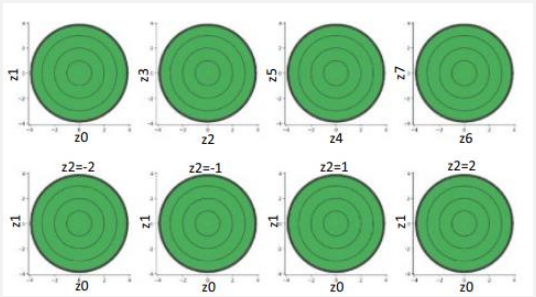
Which tasks could benefit from SELFIES' robustness?

Compact structural domains

Latent space of SMILES-based autoencoder



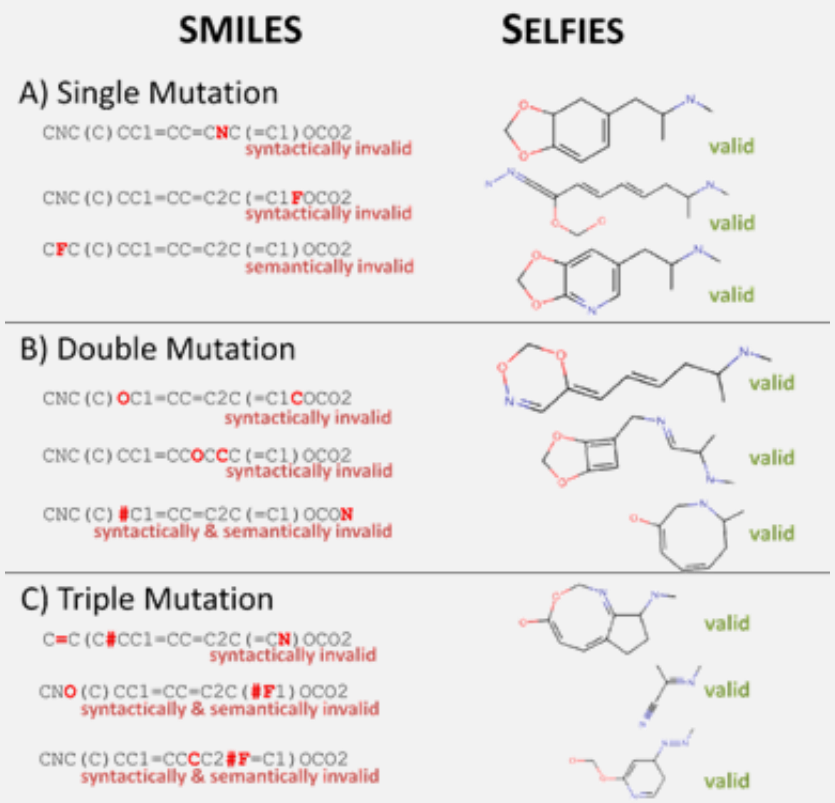
Latent space of SELFIES-based autoencoder



Syntactically and semantically valid

Syntactically or semantically invalid

Robust to character-wise mutations

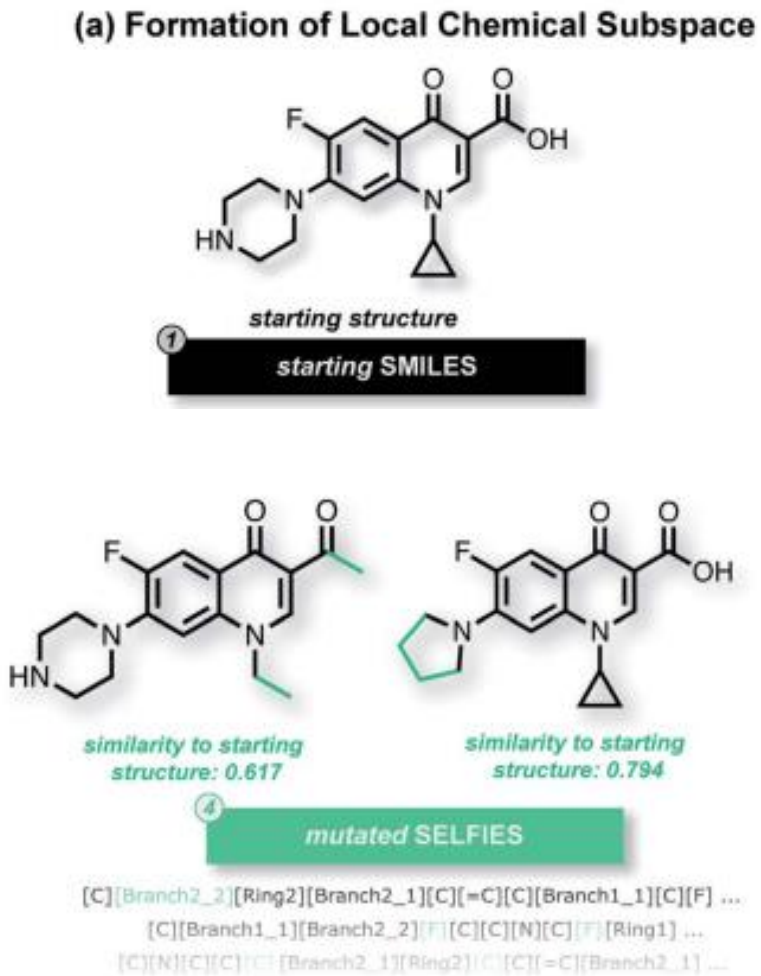


Krenn, Mario, et al. "Self-referencing embedded strings (SELFIES): A 100% robust molecular string representation." *Machine Learning: Science and Technology* 1.4 (2020): 045024.

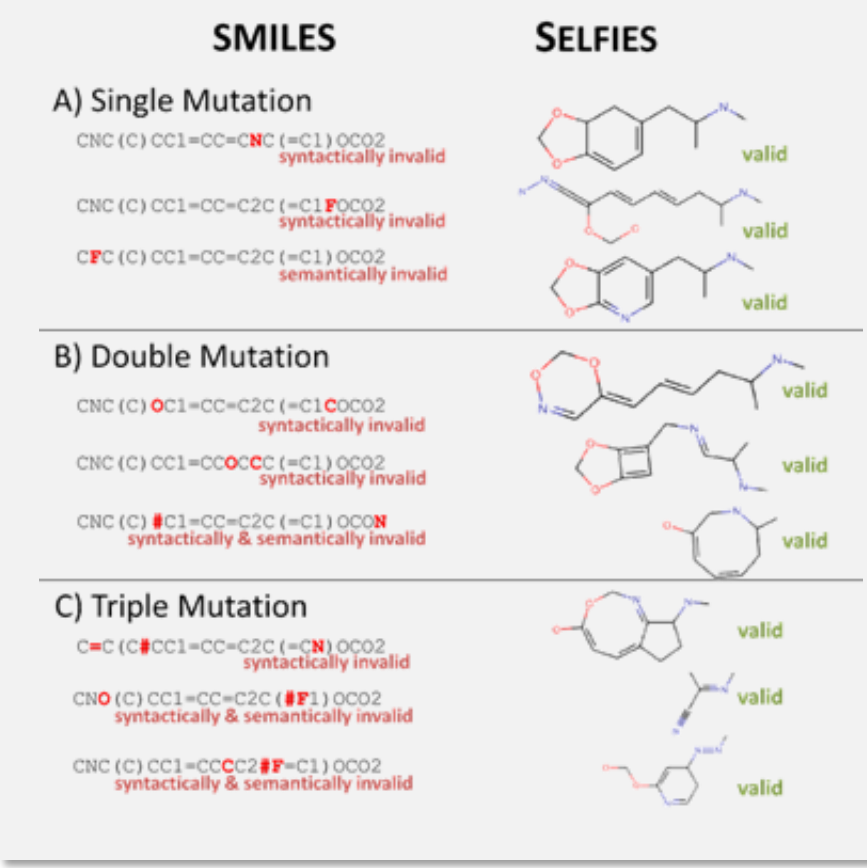
Modeling opportunities and challenges

Beyond generative models

Superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules



But: **Edit distances** in SELFIES do not relate to **structural similarity** in molecules!



Nigam, AkshatKumar, et al. "Beyond generative models: superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules using SELFIES." *Chemical science* 12.20 (2021): 7079-7090.

Beyond generative models

Superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules

Table 1 Number and percentage of unique molecules obtained within different fingerprint-based similarity thresholds (δ) of the starting structures. The molecules in each experiment were generated from 250 000 random string mutations of the starting structures. Additionally, for celecoxib, we also formed the local chemical space with a scaffold constraint

		Number of molecules (and percentage)		
Starting structure (method)	Fingerprint	$\delta > 0.75$	$\delta > 0.60$	$\delta > 0.40$
Aripirazole (SELFIES, random)	ECFP4	513 (0.25%)	4206 (2.15%)	34 416 (17.66%)
Albuterol (SELFIES, random)	FCFP4	587 (0.32%)	4156 (2.33%)	16 977 (9.35%)
Mestranol (SELFIES, random)	AP	478 (0.22%)	4079 (1.90%)	45 594 (21.66%)
Celecoxib (SELFIES, random)	ECFP4	198 (0.10%)	1925 (1.00%)	18 045 (9.44%)
Celecoxib (SELFIES, terminal 10%)	ECFP4	864 (2.02%)	9407 (21.99%)	34 187 (79.91%)
Celecoxib (SELFIES, central 10%)	ECFP4	111 (0.08%)	1767 (1.32%)	15 348 (11.45%)
Celecoxib (SELFIES, initial 10%)	ECFP4	368 (0.53%)	7345 (10.53%)	34 702 (49.74%)
Celecoxib (SMILES, random)	ECFP4	122 (18.43%)	515 (77.49%)	662 (100.00%)
Celecoxib (SMILES, terminal 10%)	ECFP4	90 (20.79%)	368 (84.99%)	433 (100.00%)
Celecoxib (SMILES, central 10%)	ECFP4	114 (22.18%)	419 (81.52%)	514 (100.00%)
Celecoxib (SMILES, initial 10%)	ECFP4	122 (19.71%)	490 (79.16%)	619 (100.00%)
Celecoxib (DeepSMILES, random)	ECFP4	132 (4.43%)	953 (31.99%)	2793 (93.76%)
Celecoxib (DeepSMILES, terminal 10%)	ECFP4	106 (9.73%)	513 (47.11%)	1083 (99.45%)
Celecoxib (DeepSMILES, central 10%)	ECFP4	53 (6.54%)	162 (19.98%)	658 (81.13%)
Celecoxib (DeepSMILES, initial 10%)	ECFP4	105 (9.28%)	609 (53.80%)	1106 (97.70%)
Celecoxib (SELFIES, scaffold constraint)	ECFP4	354 (0.44%)	6311 (7.79%)	53 479 (66.07%)
Celecoxib (CReM, ChEMBL: SCScore ≤ 2.5)	ECFP4	239 (0.58%)	5547 (13.47%)	14 887 (36.14%)

SELFIES

SMILES

DeepSMILES

While SELFIES generates the most valid molecules, only a tiny fraction of them are structurally similar

Nigam, AkshatKumar, et al. "Beyond generative models: superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules using SELFIES." *Chemical science* 12.20 (2021): 7079-7090.

Beyond generative models

Superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules

SELFIES can still be used to *explore chemical spaces* when augmented with an appropriate *measure for structural similarity*

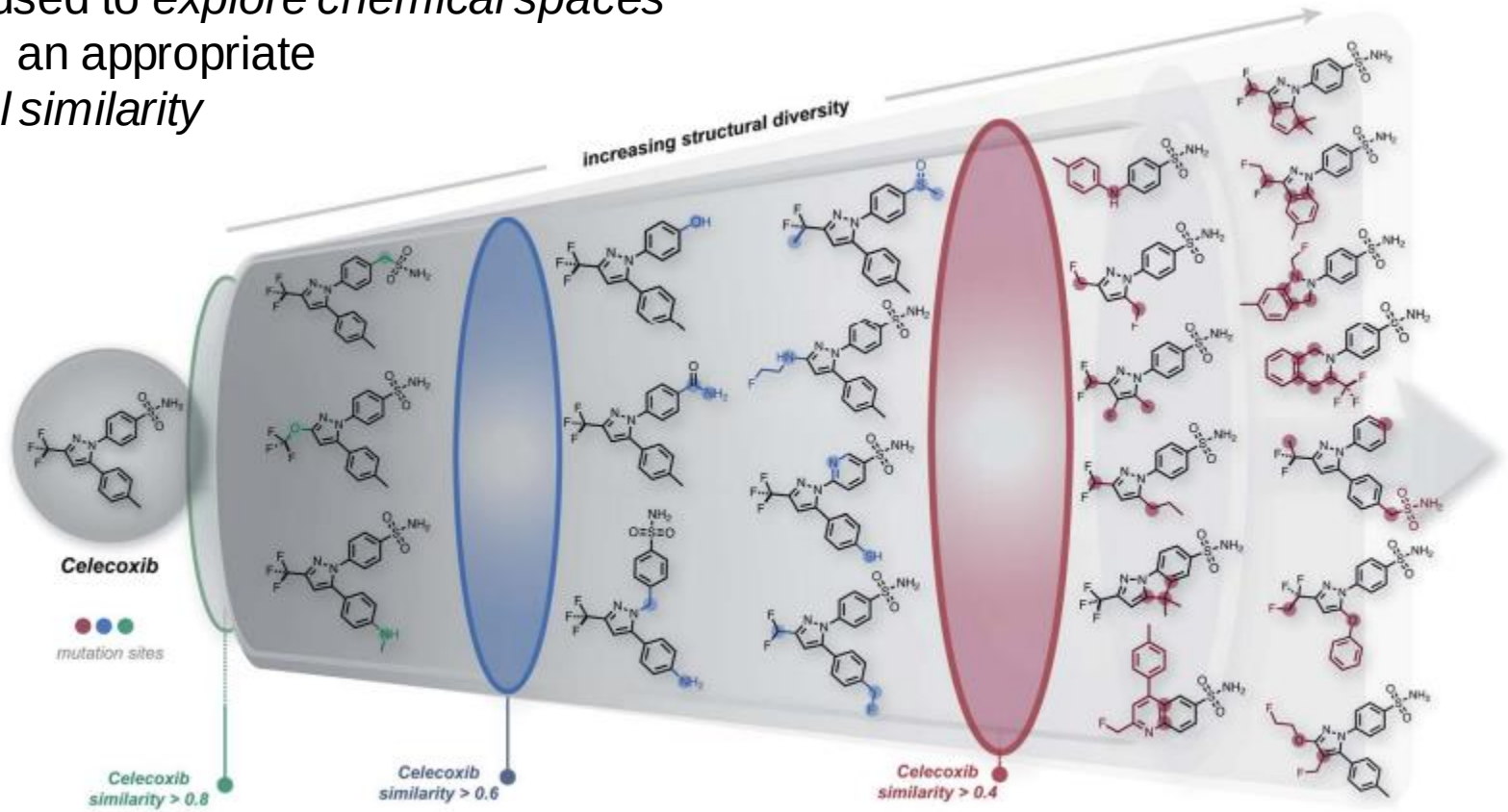
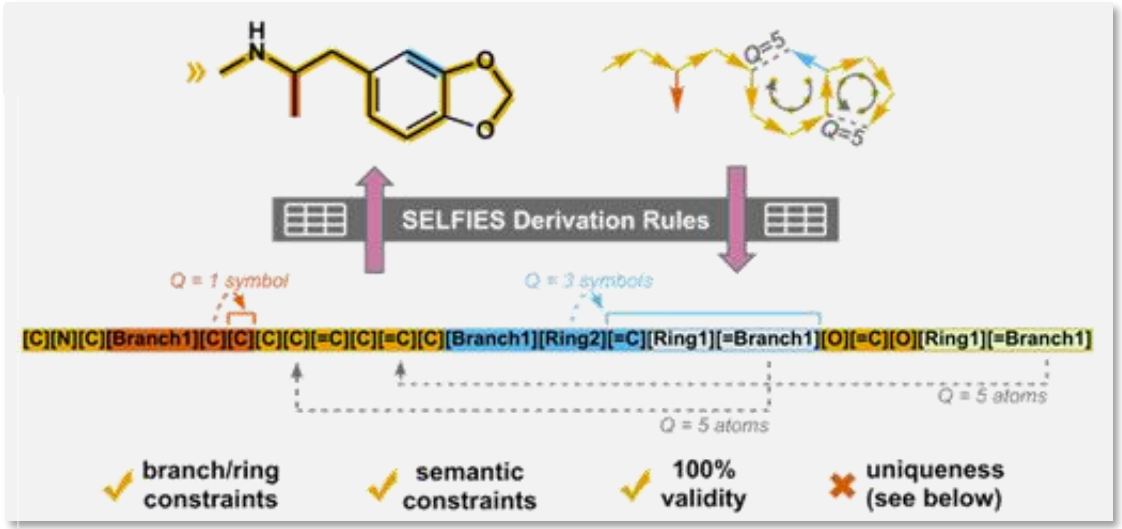


Fig. 2 Systematic local chemical space exploration of celecoxib using mutations of different SELFIES representations. The similarity is calculated using the Tanimoto distance of the ECFP4 fingerprint between celecoxib and the generated structures.

Nigam, AkshatKumar, et al. "Beyond generative models: superfast traversal, optimization, novelty, exploration and discovery (STONED) algorithm for molecules using SELFIES." *Chemical science* 12.20 (2021): 7079-7090.

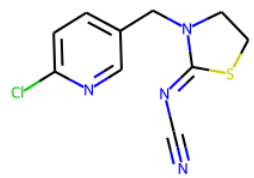
Concluding remarks

Self-referencing embedded strings as robust representation of molecules




```
In [2]: import selfies as sf
        sf.__version__
Out[2]: '2.0.0'

In [3]: # Thiacloprid - an insecticide of the neonicotinoid class
        mol_smiles = "Clc1ccc(CN2CCS/C/2=N\\C#N)cn1"
        Chem.MolFromSmiles(mol_smiles)
Out[3]:
```



```
In [4]: # Convert smiles to selfies
        mol_selfies = sf.encoder(mol_smiles)
        mol_selfies
Out[4]: '[Cl][C][=C][C][=C][Branch1][N][C][N][C][C][S][/C][-/Ring1][Branch1][=N][\\C][#N][C][=N][Ring1][#C]'
```

SELFIES team

Major contributors:

Alston Lo

Seyone Chithrananda

Chemistry advisor:

Robert Pollice

If you want to try SELFIES, go to

https://github.com/aspuru-guzik-group/selfies

or download it with

pip install selfies

We're hiring!

Contact: florian.haese@bayer.com

Science for a better life