Nuclear magnetic resonance spectroscopy with language transformers

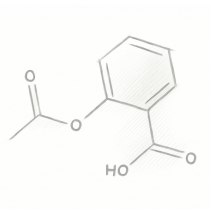
R. Andreev

Nuclear magnetic resonance spectroscopy with language transformers

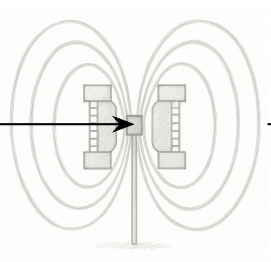
R. Andreev

Thanks to

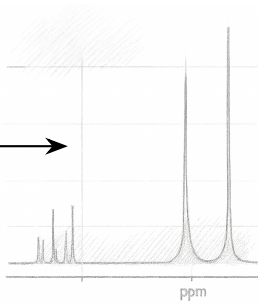
- E. Konukoglu Computational Vision and Learning (D-ITET, ETHZ)
- N. Schmid Applied Mathematics and Physics (ZHAW)
- M.-O. Ebert & S. Papalo Organic Chemistry (D-CHAB, ETHZ)
- A. Hove Lambda GPU Cloud, who provided most of the compute



analyte

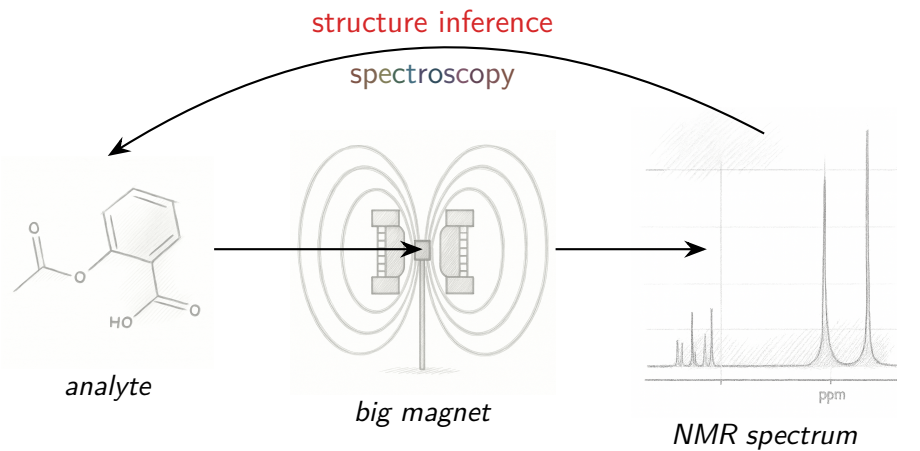


big magnet

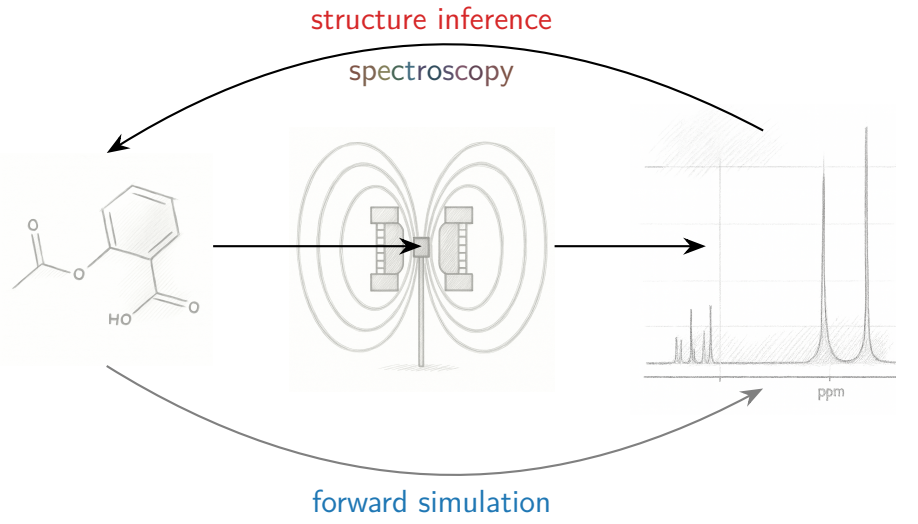


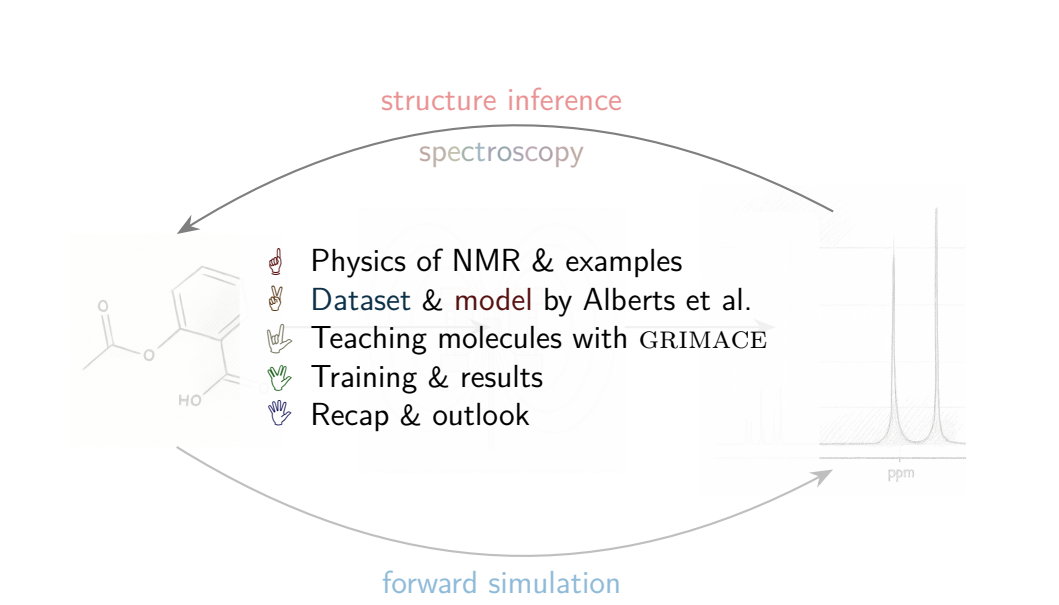
NMR spectrum

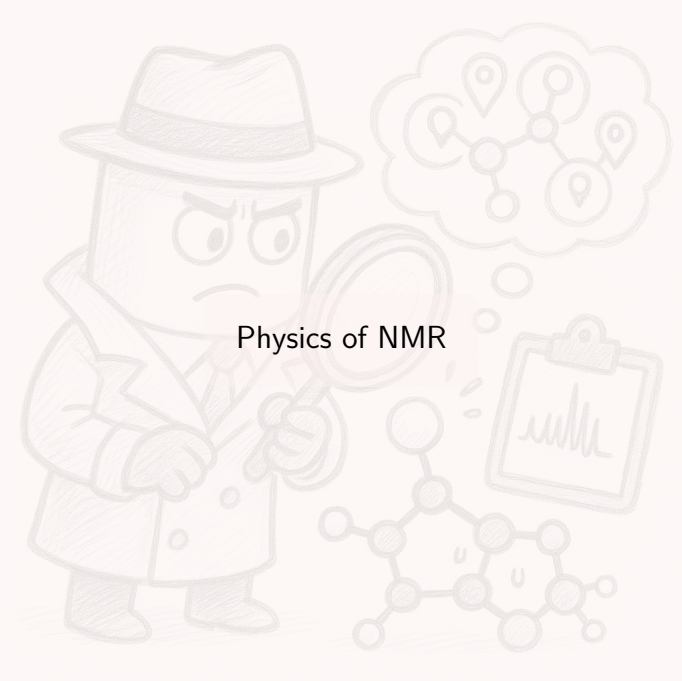
Nuclear magnetic resonance



Nuclear magnetic resonance




structure inference
spectroscopy
Physics of NMR & examples
Dataset & model by Alberts et al.
Teaching molecules with GRIMACE
Training & results
Recap & outlook
O
O
HO
ppm
forward simulation



Physics of NMR

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$. Its observables are combinations of $\mathbb{I}$ and the Pauli matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t \psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t \hat{P} \psi^{\mathrm{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$\langle \hat{M} \rangle(t) = \psi(t)^\dagger \hat{M} \psi(t) = (\psi^{\mathrm{up}})^\dagger (\hat{P}^\dagger \hat{E}_t^\dagger \hat{M} \hat{E}_t \hat{P}) \psi^{\mathrm{up}} = e^{-i\omega t}$$

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$. Its observables are combinations of $\mathbb{I}$ and the Pauli matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \qquad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors

$$\psi^{\mathsf{up}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \psi_{\mathsf{down}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t \psi = -\frac{i}{\hbar}\widehat{H}\psi$ gives the spin evolution operator
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t \psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t \hat{P} \psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$m(t) = \psi(t)^\dagger \hat{M} \psi(t) = (\psi^{\mathsf{up}})^\dagger (\hat{P}^\dagger \hat{E}_t^\dagger \hat{M} \hat{E}_t \hat{P}) \psi^{\mathsf{up}} = e^{-i\omega t}$$

- Package the superposition of *all spins* in the *free induction decay*
- Fourier-transform, then shift and scale w.r.t. a reference compound

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is

$$\hat{H} = \hbar\,\omega\,I_z \quad + \quad \ldots$$

with Larmor frequency

$$\omega \approx \quad -\gamma_{\text{isotope}}\,B_0\,(1 - \text{shielding}) \quad + \quad 2\pi \sum_{s'} J_{ss'} \times (\pm\tfrac{1}{2})_{s'}$$

- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t \psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t \hat{P} \psi^{\text{up}}$
- The macroscopic xy magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is

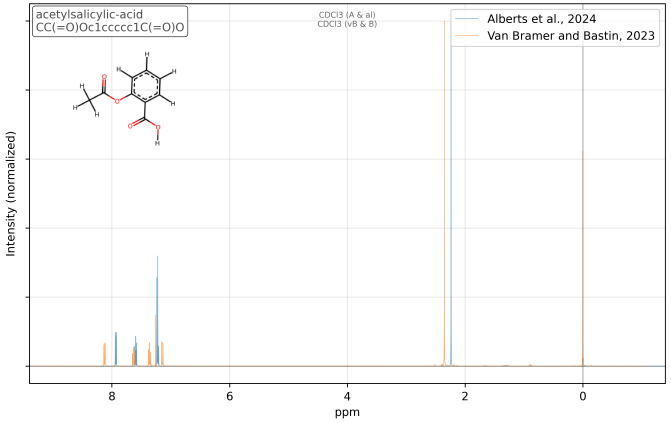
$$\hat{H} = \hbar\,\omega\,I_z \quad + \quad \ldots$$

with Larmor frequency

$$\omega \approx \quad -\gamma_{\text{isotope}}\,B_0\,(1 - \text{shielding}) \quad + \quad 2\pi \sum_{s'} J_{ss'} \times (\pm\tfrac{1}{2})_{s'}$$

- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t \psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t \hat{P} \psi^{\text{up}}$
- The macroscopic xy magnetization $\hat{M} = \sigma_- + i\sigma_-$ of s is observed as

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling $\rightarrow$



- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator

$$\hat{E}_t = \begin{pmatrix} e^{+\frac{1}{2}i\omega t} & 0 \\ 0 & e^{-\frac{1}{2}i\omega t} \end{pmatrix}$$

assuming $\hat{H} = \hbar\omega I_z$ (notably, ignoring relaxation)

- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathrm{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator $\hat{E}_t$
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$m(t) = \psi(t)^\dagger \hat{M}\psi(t) = (\psi^{\mathsf{up}})^\dagger(\hat{P}^\dagger\hat{E}_t^\dagger\hat{M}\hat{E}_t\hat{P})\psi^{\mathsf{up}} = e^{-i\omega t}$$

- Package the superposition of *all spins* in the *free induction decay*
- Fourier-transform, then shift and scale w.r.t. a reference compound

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator $\hat{E}_t$
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis

$$\psi^{\mathsf{up}} \to \hat{P}\psi^{\mathsf{up}} \quad \text{where} \quad \hat{P} = \exp(-i\tfrac{\pi}{4}\sigma_y)$$

- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$m(t) = \psi(t)^\dagger\hat{M}\psi(t) = (\psi^{\mathsf{up}})^\dagger(\hat{P}^\dagger\hat{E}_t^\dagger\hat{M}\hat{E}_t\hat{P})\psi^{\mathsf{up}} = e^{-i\omega t}$$

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator $\hat{E}_t$
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$m(t) = \psi(t)^\dagger \hat{M}\psi(t) = (\psi^{\mathsf{up}})^\dagger(\hat{P}^\dagger\hat{E}_t^\dagger\hat{M}\hat{E}_t\hat{P})\psi^{\mathsf{up}} = e^{-i\omega t}$$

- Package the superposition of *all spins* in the *free induction decay*
- Fourier-transform, then shift and scale w.r.t. a reference compound

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator $\hat{E}_t$
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$m(t) = \psi(t)^\dagger\hat{M}\psi(t) = (\psi^{\mathsf{up}})^\dagger(\hat{P}^\dagger\hat{E}_t^\dagger\hat{M}\hat{E}_t\hat{P})\psi^{\mathsf{up}} = e^{-i\omega t}$$

- Package the superposition of *all spins* in the *free induction decay*
- Fourier-transform, then shift and scale w.r.t. a reference compound

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator $\hat{E}_t$
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

$$m(t) = \psi(t)^\dagger\hat{M}\psi(t) = (\psi^{\mathsf{up}})^\dagger(\hat{P}^\dagger\hat{E}_t^\dagger\hat{M}\hat{E}_t\hat{P})\psi^{\mathsf{up}} = e^{-i\omega t}$$

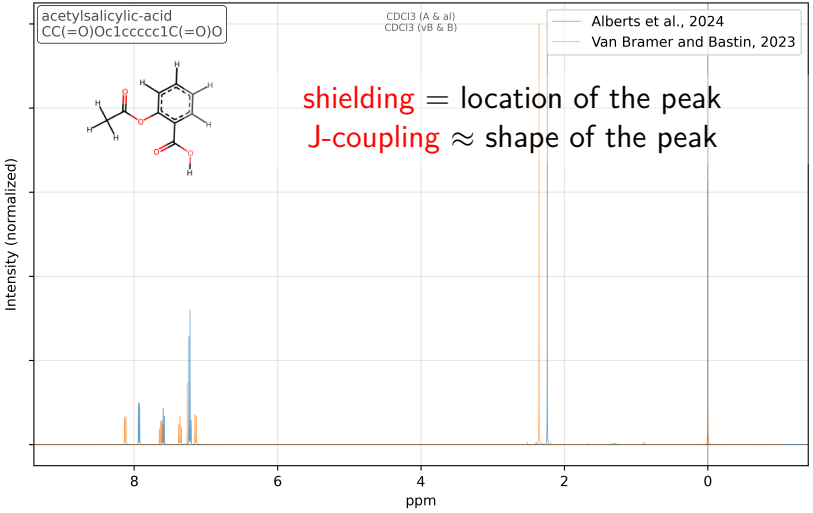
- Package the superposition of *all spins* in the *free induction decay*
- Fourier-transform, then shift and scale w.r.t. a reference compound

- The spin of a $\frac{1}{2}$-spin particle ($^1\mathrm{H}$, $^{13}\mathrm{C}$) is a normed vector $\psi \in \mathbb{C}^2$
- The operator $I_z = \frac{1}{2}\sigma_z$ has eigenvalues $\pm\frac{1}{2}$ and eigenvectors *up*/*down*
- The Hamiltonian of a $\frac{1}{2}$-spin s in a z-magnetic field $\vec{B}_0$ is $\hat{H} \approx \hbar\omega I_z$
- In NMR, we measure ω, and infer/interpret shielding and J-coupling
- Schrödinger eqn. $\partial_t\psi = -\frac{i}{\hbar}\hat{H}\psi$ gives the spin evolution operator $\hat{E}_t$
- At thermal equilibrium, $\vec{B}_0$ causes a small excess of *up* spins ψ^{up}
- A radiofrequency pulse knocks those spins onto the x-axis
- Ignoring the relaxation, the spin state is $\psi(t) = \hat{E}_t\hat{P}\psi^{\mathsf{up}}$
- The macroscopic xy-magnetization $\hat{M} = \sigma_x + i\sigma_y$ of s is observed as

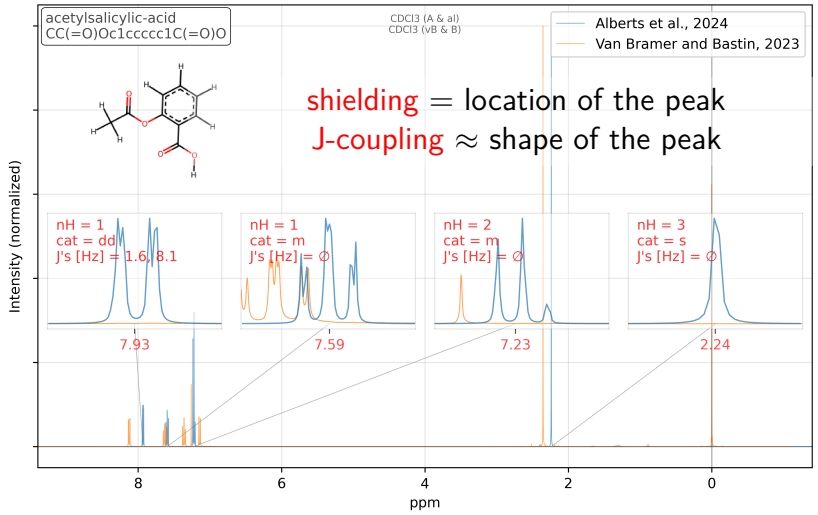
$$m(t) = \psi(t)^\dagger\hat{M}\psi(t) = (\psi^{\mathsf{up}})^\dagger(\hat{P}^\dagger\hat{E}_t^\dagger\hat{M}\hat{E}_t\hat{P})\psi^{\mathsf{up}} = e^{-i\omega t}$$

- Package the superposition of *all spins* in the *free induction decay*
- Fourier-transform, then shift and scale w.r.t. a reference compound

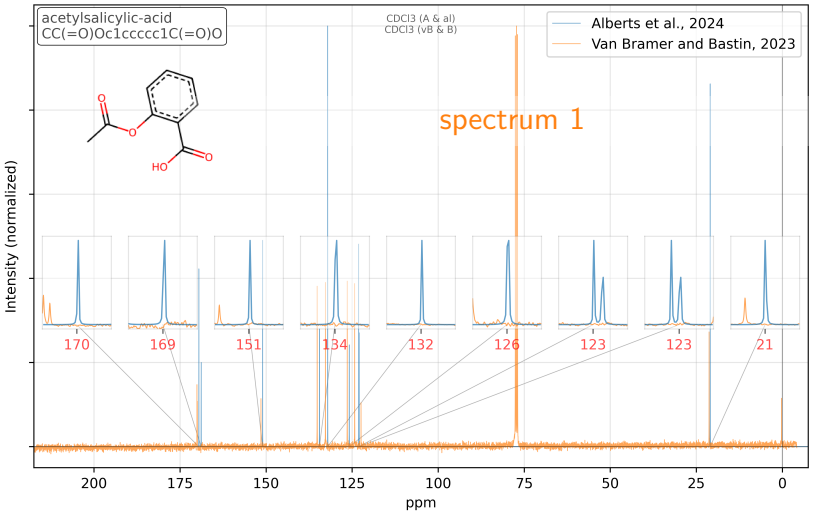
^{1}H–NMR spectrum of acetylsalicylic acid (*aspirin*)



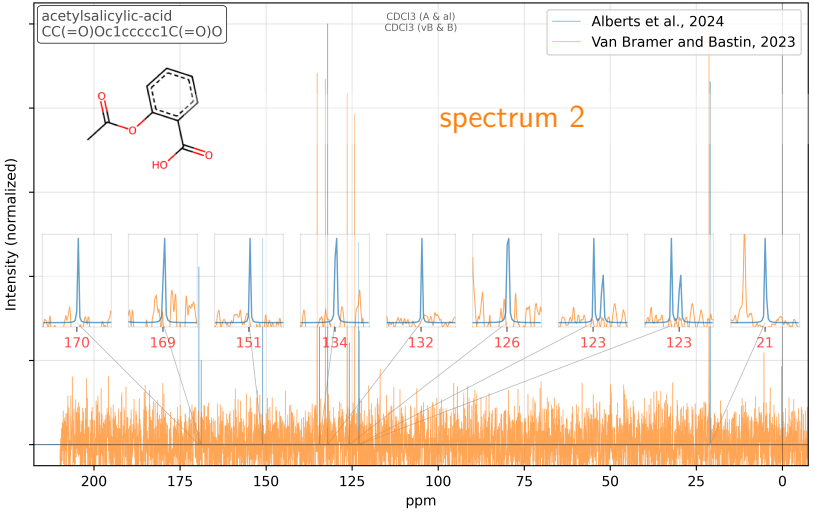
^{1}H–NMR spectrum of acetylsalicylic acid (*aspirin*)



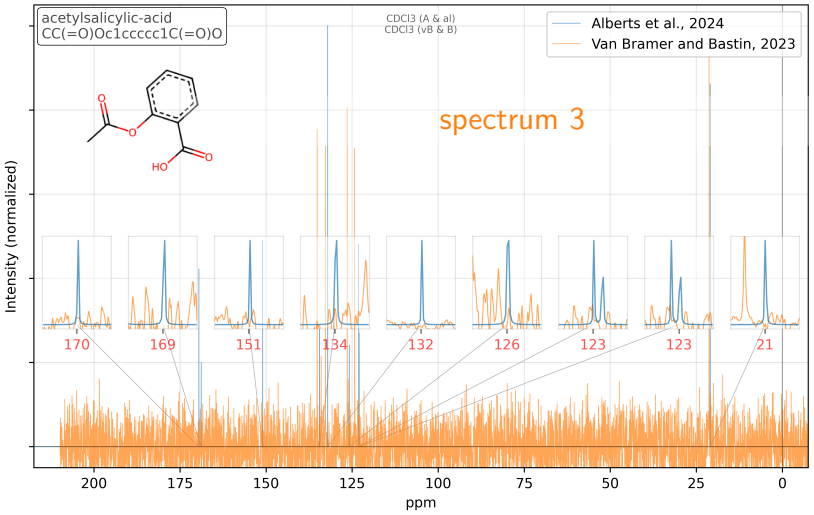
^{13}C–NMR spectrum of acetylsalicylic acid (*aspirin*)



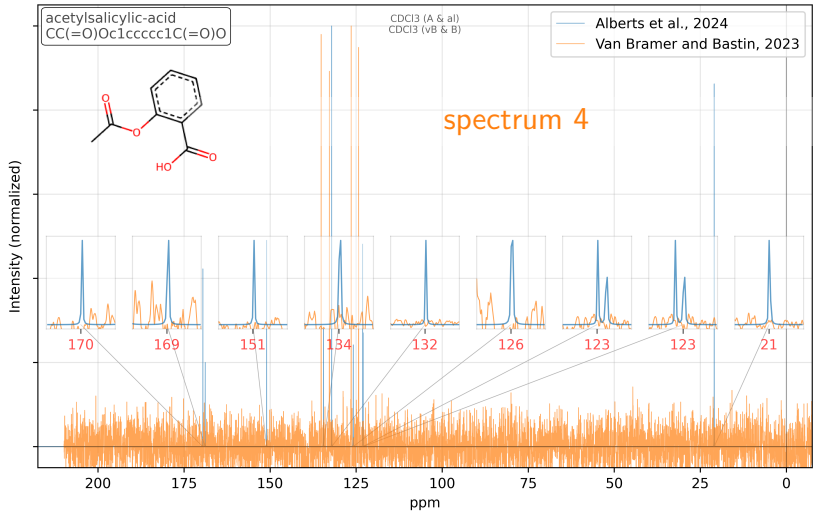
^{13}C–NMR spectrum of acetylsalicylic acid (*aspirin*)



^{13}C–NMR spectrum of acetylsalicylic acid (*aspirin*)



^{13}C–NMR spectrum of acetylsalicylic acid (*aspirin*)



Strychnine

- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine is chiral, with fused rings and quite rigid
- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]
- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

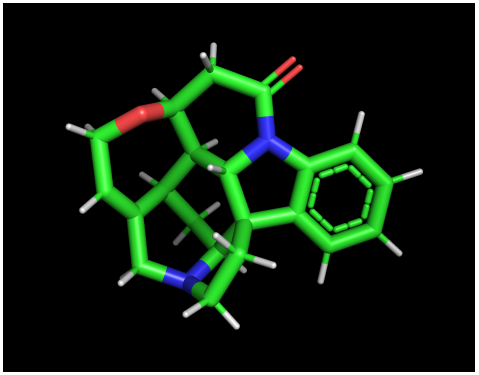
Many thanks to S. Papalo and M.-O. Ebert of LOC, ETH Zurich!

MestReNova v15.1.0-38027 with NMRPredict v1.11

OpenMM† 8.4.0 + ORCA† 6.1.0 / GNN predictor
† – responsible vibe coding involved

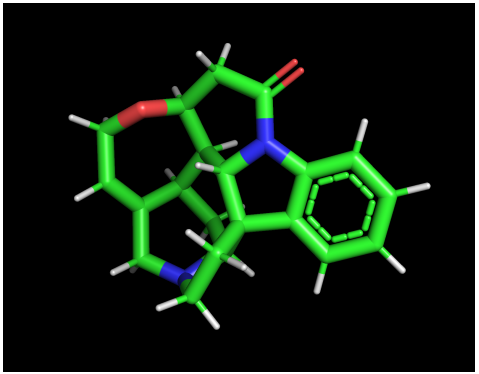
Strychnine

- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine is chiral, with fused rings and quite rigid



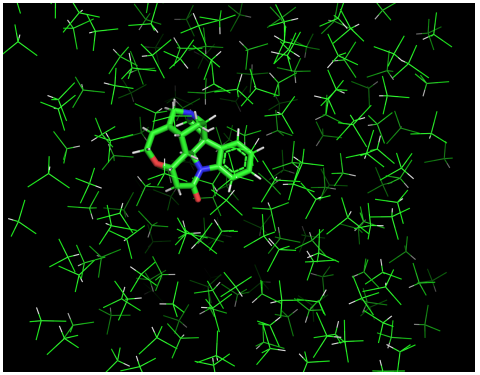
- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]
- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine is chiral, with fused rings and quite rigid



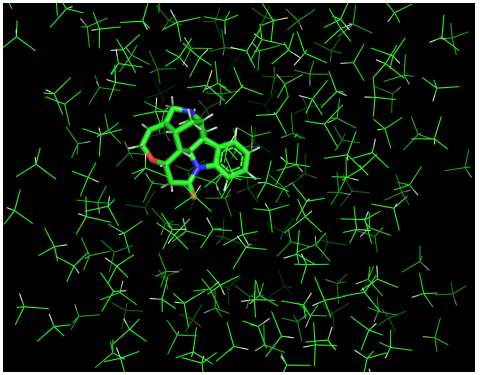
- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]
- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine in chloroform-d, computed with OpenMM



- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]
- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine in chloroform-d, computed with OpenMM



- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]
- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine in chloroform-d, computed with OpenMM
- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]:

  Strychnine is a natural product that, through isolation, structural elucidation and synthetic efforts, shaped the field of organic chemistry.

  Currently, strychnine is used as a pesticide to control rodents because of its potent neurotoxicity.

  The polycyclic architecture of strychnine has inspired chemists [..].

  Here we report the biosynthetic pathway of strychnine, along with the related molecules brucine and diaboline.

- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

Many thanks to S. Papalo and M.-O. Ebert of LOC, ETH Zurich!

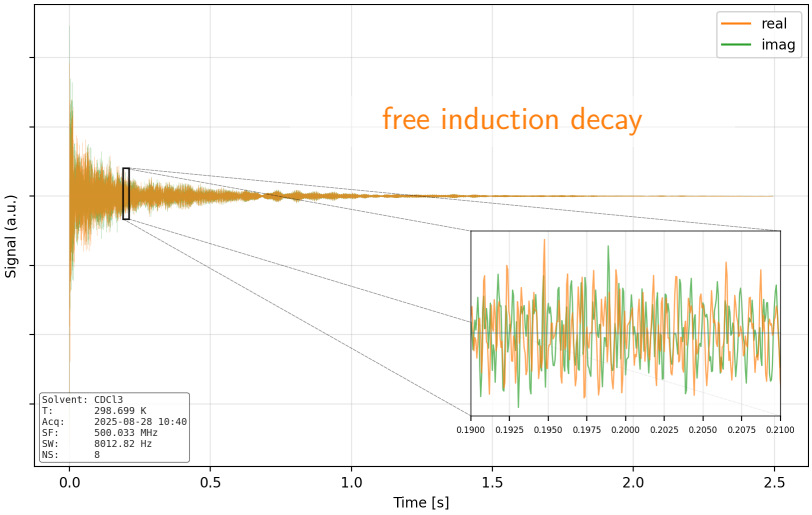
- /ˈstrɪkniːn, -nɪn/; US chiefly: /-naɪn/ [wiki]
- Strychnine in chloroform-d, computed with OpenMM
- Hong et al., "Biosynthesis of strychnine", Nature (2022) [2]
- Let's see the ^{1}H-NMR spectrum, both measured and simulated...

Many thanks to S. Papalo and M.-O. Ebert of LOC, ETH Zurich!

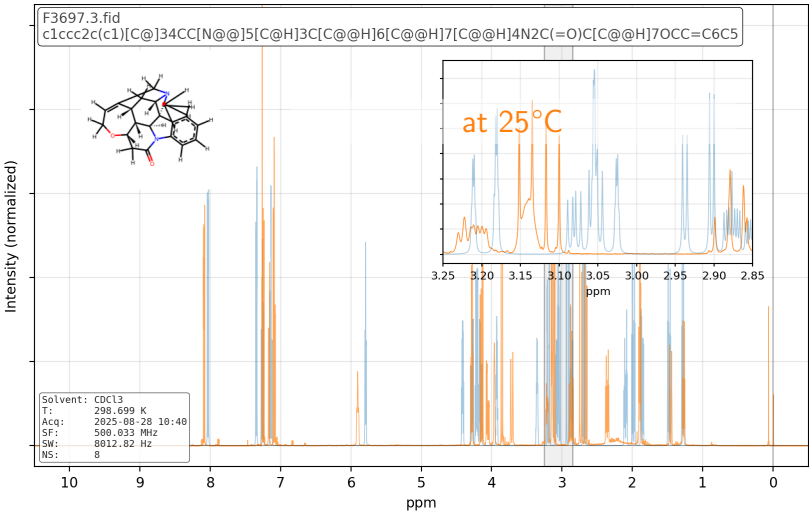
MestReNova v15.1.0-38027 with NMRPredict v1.11

OpenMM† 8.4.0 + ORCA† 6.1.0 / GNN predictor
† – responsible vibe coding involved

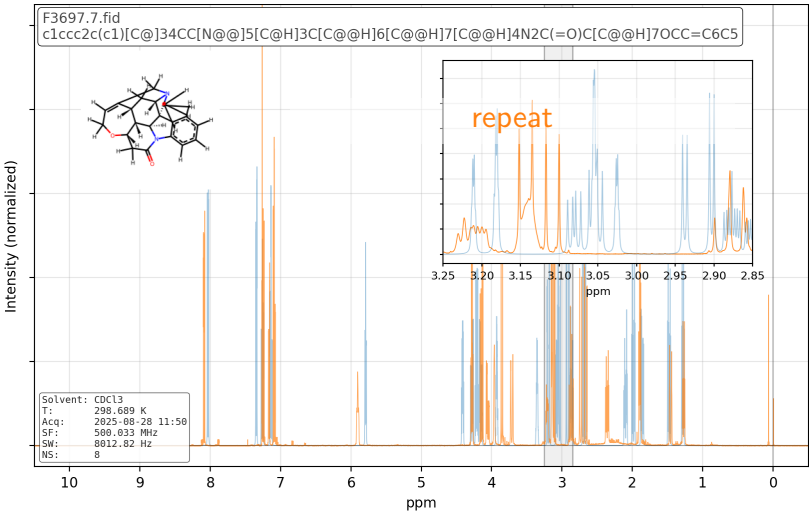
^{1}H–NMR spectrum of strychnine



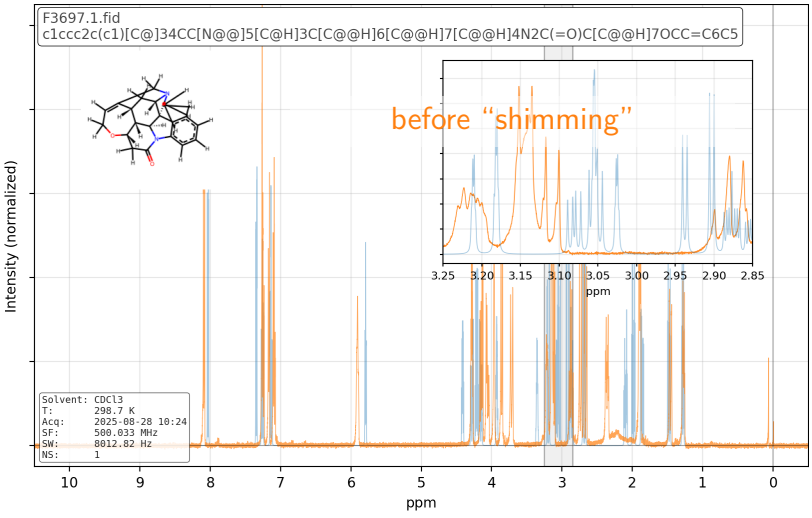
^{1}H–NMR spectrum of strychnine



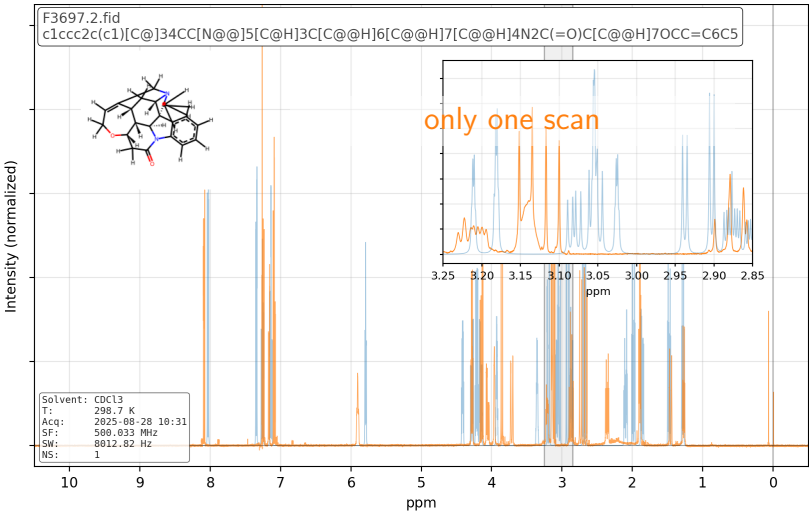
^{1}H–NMR spectrum of strychnine



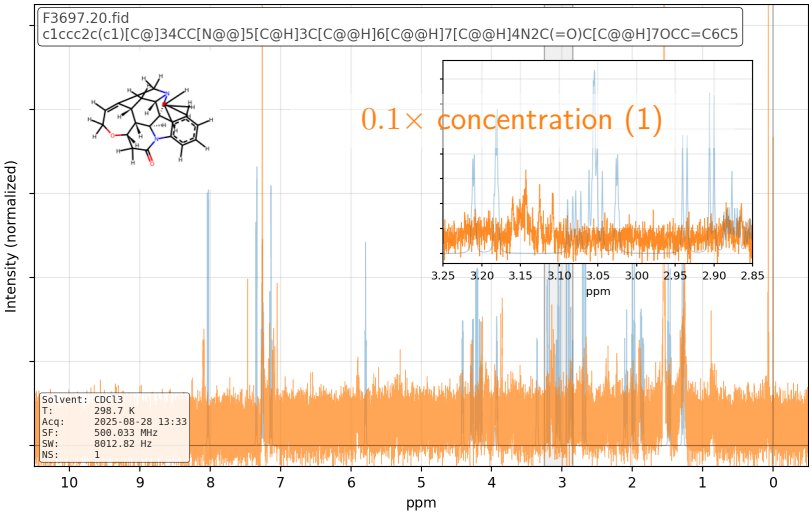
^{1}H–NMR spectrum of strychnine



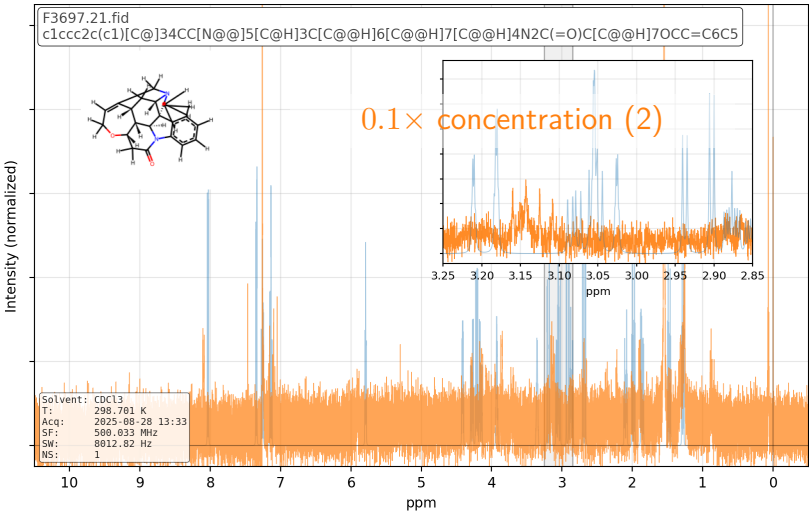
^{1}H–NMR spectrum of strychnine



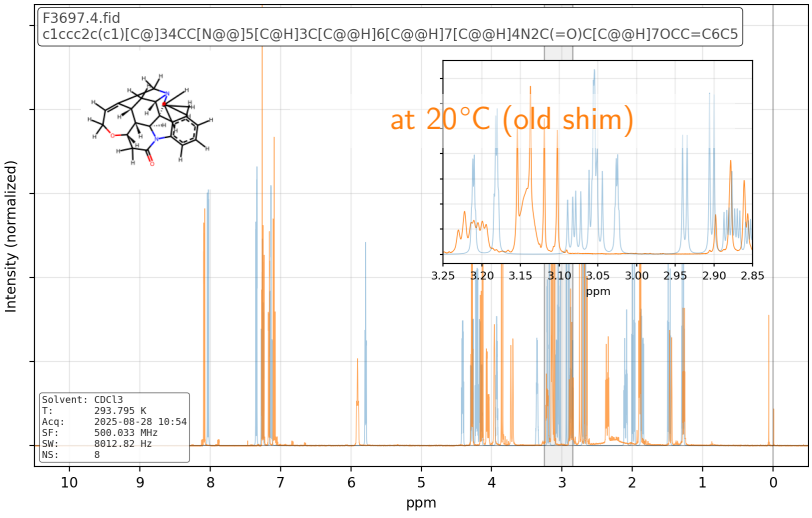
^{1}H–NMR spectrum of strychnine



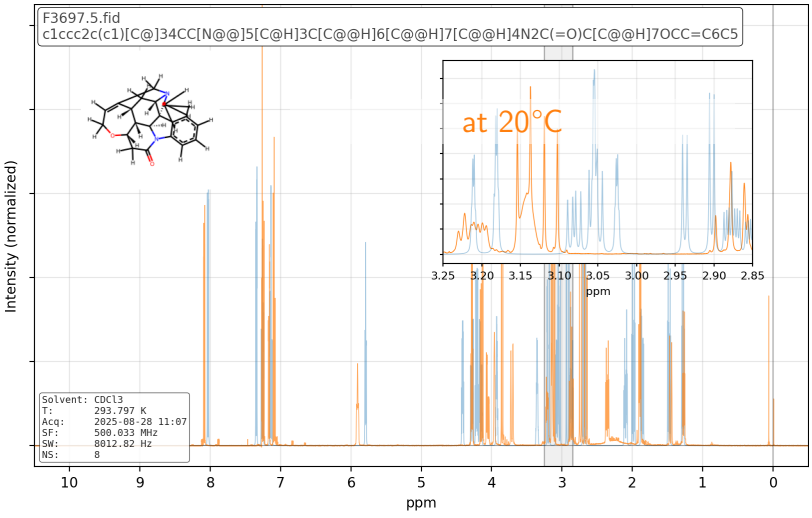
^{1}H–NMR spectrum of strychnine



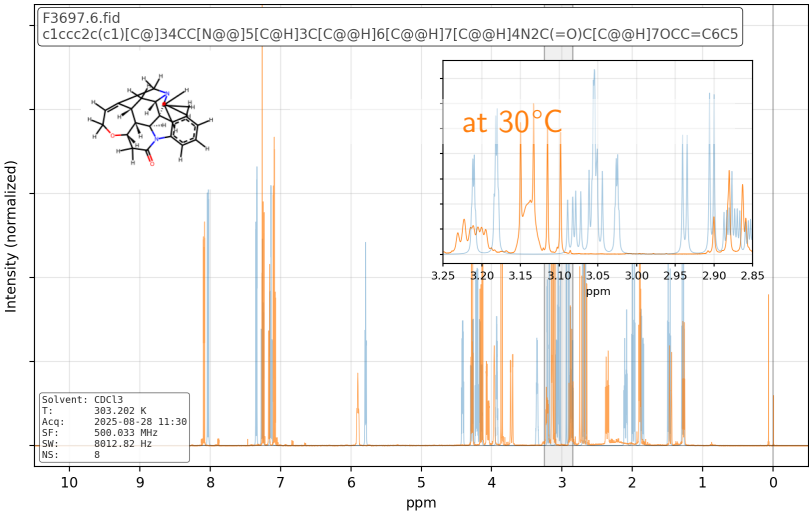
^{1}H–NMR spectrum of strychnine



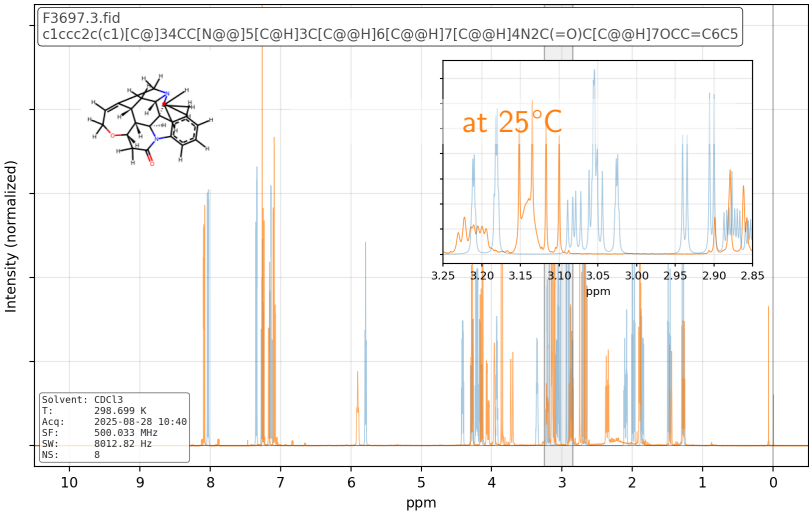
^{1}H–NMR spectrum of strychnine



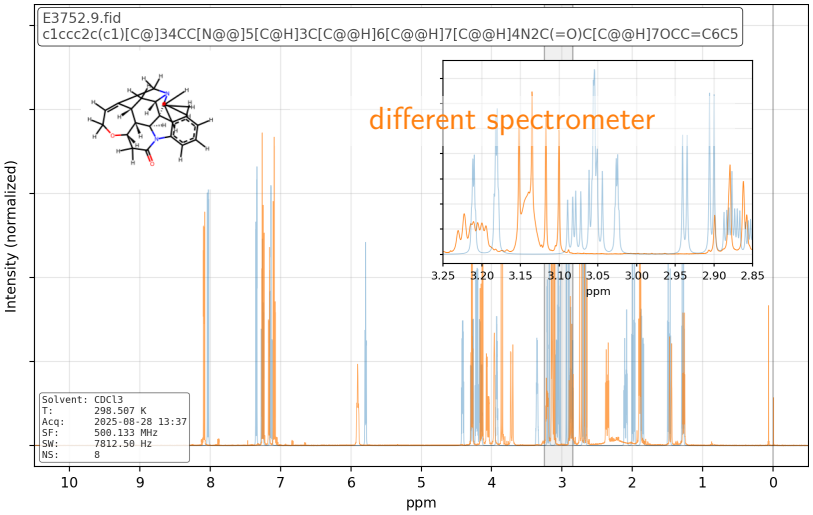
^{1}H–NMR spectrum of strychnine



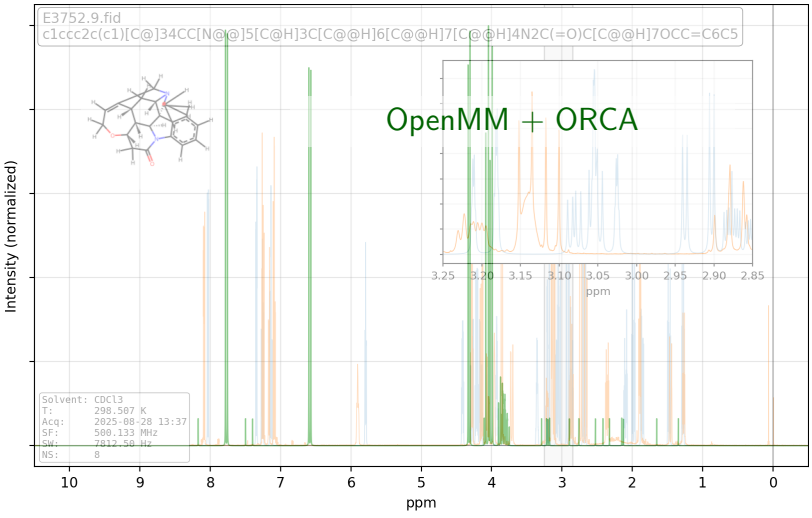
^{1}H–NMR spectrum of strychnine



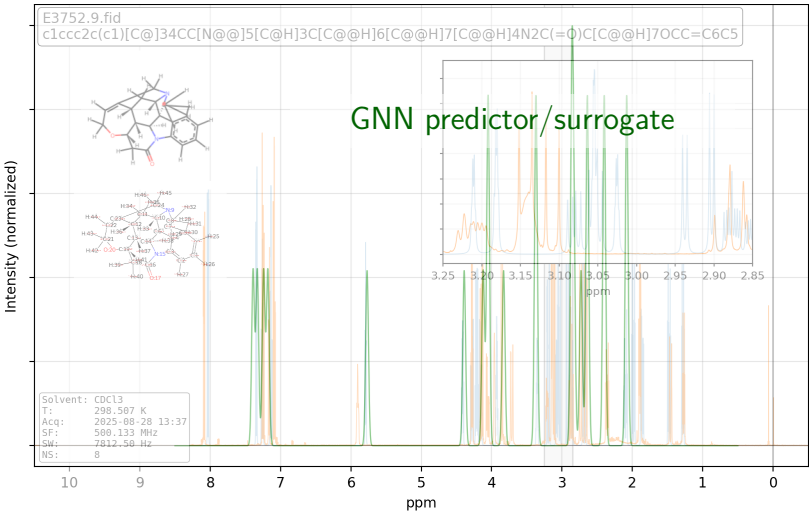
^{1}H–NMR spectrum of strychnine



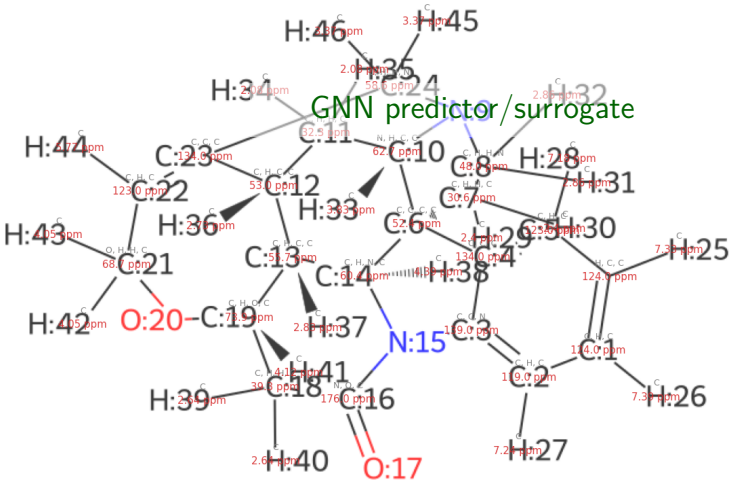
^{1}H–NMR spectrum of strychnine



^{1}H–NMR spectrum of strychnine



^{1}H–NMR spectrum of strychnine



Annotations by MestReNova (via Alberts et al. [1]):

^{1}H peaks:

```
...
{
  "category": "ddt",
  "centroid": 6.992,
  "delta": 6.993,
  "j_values": "0.88_2.02_8.79_",
  "nH": 1,
  "rangeMax": 7.021,
  "rangeMin": 6.965
},
...
```

^{13}C peaks:

```
...
{
  "delta (ppm)": 130.27,
  "integral": 0.00096,
  "intensity": 0.0512,
  "width (ppm)": 0.0119
},
...
```

HSQC peaks:

```
...
{
  "13C_centroid": 56.18,
  "13C_max": 56.91,
  "13C_min": 55.45,
  "1H_centroid": 3.813,
  "1H_max": 3.866,
  "1H_min": 3.759,
  "nH": 3.0  # !!!
},
...
```

Repackage as NMR text data for language transformers:

```
F|C16H23BrO4
Q|6.0:1.11-1.25@23|3.0:3.76-3.87@56.2|3.0:1.01-1.12@14.2|···|1.0:3.03-3.14@37.6
C|3.1@172|3.1@157|5.4@130|3.2@130|3.1@129|5.3@128|5.3@112|···|9.7@14.2
H|1:7.09-7.13:dp:0.86,1.8|1:6.96-7.02:ddt:0.88,2,8.8|···|3:1.04-1.09:t:6.4
```

– we use this as the de facto NMR spectrum

Problem statement:

$$\mathrm{F} + \mathrm{Q} + \mathrm{C} + \mathrm{H} \rightarrow \mathrm{MOLECULE}$$

Dataset & model by Alberts et al.

- No large-scale open experimental dataset
- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]
- Alberts et al. [4, 5] trained transformers on simulated spectra

[4] M Alberts et al. "Learning the language of NMR: Structure elucidation from NMR spectra using transformer models". *ChemRxiv* (2023)

[5] M Alberts et al. "Automated structure elucidation at human-level accuracy via a multimodal multitask language model". *ChemRxiv* (2025)

- No large-scale open experimental dataset
- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]

[1] M Alberts et al. "Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry". *NeurIPS 2024 Datasets and Benchmarks Track*. 2024

[3] D Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017

using MestReNova, similarly to

[6] F Hu et al. "Accurate and efficient structure elucidation from routine one-dimensional NMR spectra using multitask machine learning". *ACS Central Science* 10.11 (2024), pp. 2162–2170

Is it "representative"?

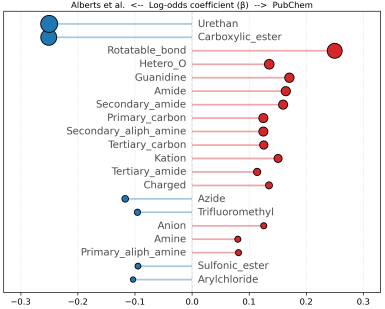
- Alberts et al. [4, 5] trained transformers on simulated spectra

[4] M Alberts et al. "Learning the language of NMR: Structure elucidation from NMR spectra using transformer models". *ChemRxiv* (2023)

- No large-scale open experimental dataset
- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]

[1] M Alberts et al. "Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry". *NeurIPS 2024 Datasets and Benchmarks Track*. 2024

[3] D Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017



Molecule provenance,

Alberts et al. [1] vs PubChem [7]

is detected by logistic regression based on functional groups with AUC-ROC ~75%

Circle area = permutation importance $\lesssim 2\%$

- Alberts et al. [4, 5] trained transformers on simulated spectra

- No large-scale open experimental dataset
- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]

[1] M Alberts et al. "Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry". *NeurIPS 2024 Datasets and Benchmarks Track*. 2024

[3] D Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017

This is the dataset we use.

- Alberts et al. [4, 5] trained transformers on simulated spectra

[4] M Alberts et al. "Learning the language of NMR: Structure elucidation from NMR spectra using transformer models". *ChemRxiv* (2023)

[5] M Alberts et al. "Automated structure elucidation at human-level accuracy via a multimodal multitask language model". *ChemRxiv* (2025)

- No large-scale open experimental dataset
- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]

[1] M Alberts et al. "Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry". *NeurIPS 2024 Datasets and Benchmarks Track*. 2024

[3] D Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017

- Alberts et al. [4, 5] trained transformers on simulated spectra

[4] M Alberts et al. "Learning the language of NMR: Structure elucidation from NMR spectra using transformer models". *ChemRxiv* (2023)

[5] M Alberts et al. "Automated structure elucidation at human-level accuracy via a multimodal multitask language model". *ChemRxiv* (2025)

- Alberts et al. [4] trained a ~30M° transformer to read a string like

```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

- A SMILES is a (non-unique) serialization of the molecular graph
- At inference: beam search heuristic to generate the most likely SMILES
- Accuracy: Is the correct molecule among the top–N predictions?
- They reported accuracy of

~76% top–10 on $\mathrm{F} + \mathrm{H} \longrightarrow \mathrm{SMILES}$
~85% top–10 on $\mathrm{F} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

- Alberts et al. [4] trained a ~30M° transformer to read a string like

```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

– like in Hu et al. [6]

- A SMILES is a (non-unique) serialization of the molecular graph
- At inference: beam search heuristic to generate the most likely SMILES
- Accuracy: Is the correct molecule among the top–N predictions?
- They reported accuracy of

~76% top–10 on $\mathrm{F} + \mathrm{H} \longrightarrow \mathrm{SMILES}$
~85% top–10 on $\mathrm{F} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

- Alberts et al. [4] trained a ~30M° transformer to read a string like

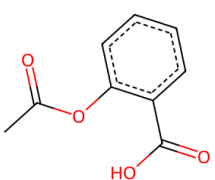
```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

- A SMILES is a (non-unique) serialization of the molecular graph

```
c1ccc(C(=O)O)c(c1)OC(C)=O
CC(=O)Oc1c(cccc1)C(O)=O
CC(=O)Oc1c(C(O)=O)cccc1
c1cc(C(=O)O)c(OC(=O)C)cc1
...
```



The hydrogens H are implied by chemical valency!

- Alberts et al. [4] trained a ~30M° transformer to read a string like

```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

- A SMILES is a (non-unique) serialization of the molecular graph
- At inference: beam search heuristic to generate the most likely SMILES ...called
  - candidates
  - hypotheses
  - predictions
- Accuracy: Is the correct molecule among the top–N predictions?
- They reported accuracy of

$$\sim 76\% \text{ top–}10 \quad \text{on} \quad \mathrm{F} + \mathrm{H} \longrightarrow \mathrm{SMILES}$$
$$\sim 85\% \text{ top–}10 \quad \text{on} \quad \mathrm{F} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$$

- Alberts et al. [4] trained a ~30M° transformer to read a string like

```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

- A SMILES is a (non-unique) serialization of the molecular graph
- At inference: beam search heuristic to generate the most likely SMILES
- Accuracy: Is the correct molecule among the top–N predictions?
- They reported accuracy of

~76% top–10 on $\mathrm{F} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

~85% top–10 on $\mathrm{F} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

- Alberts et al. [4] trained a ~30M° transformer to read a string like

```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

- A SMILES is a (non-unique) serialization of the molecular graph
- At inference: beam search heuristic to generate the most likely SMILES
- Accuracy: Is the correct molecule among the top–N predictions?
- They reported accuracy of

~76% top–10 on $\mathrm{F} + \mathrm{H} \longrightarrow \mathrm{SMILES}$
~85% top–10 on $\mathrm{F} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

- Alberts et al. [4] trained a ~30M° transformer to read a string like

```
C 9 H 8 O 4 1HNMR | 7.97 7.90 dd 1H J 1.60 8.10 | 7.64
7.55 m 1H | 7.26 7.18 m 2H | 2.26 2.22 s 3H
```

- ...and infer the molecule as a *canonical* SMILES like

```
CC(=O)Oc1ccccc1C(=O)O
```

- A SMILES is a (non-unique) serialization of the molecular graph
- At inference: beam search heuristic to generate the most likely SMILES
- Accuracy: Is the correct molecule among the top–N predictions?
- They reported accuracy of

~76% top–10 on $\mathrm{F} + \mathrm{H} \longrightarrow \mathrm{SMILES}$
~85% top–10 on $\mathrm{F} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

Similarly:

Similarly:

- We trained a ~83M° autoregressive transformer DistilGPT2
- ...to predict the *canonical* SMILES with accuracy

$$\sim 84\% \text{ top–10} \quad \text{on} \quad \mathrm{F} + \mathrm{Q} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$$

Similarly:

- We trained a ~83M° autoregressive transformer DistilGPT2
- ...to predict the *canonical* SMILES with accuracy

~84% top–10 on $\mathrm{F} + \mathrm{Q} + \mathrm{C} + \mathrm{H} \longrightarrow \mathrm{SMILES}$

Similarly:

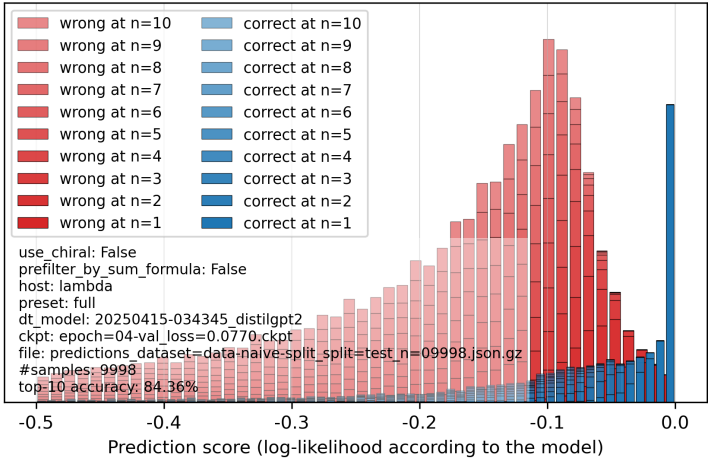
- We trained a ~83M° autoregressive transformer DistilGPT2
- ...to predict the *canonical* SMILES with accuracy

  ~84% top–10 on F + Q + C + H → SMILES

Let's have a look at the structure of the predictions:

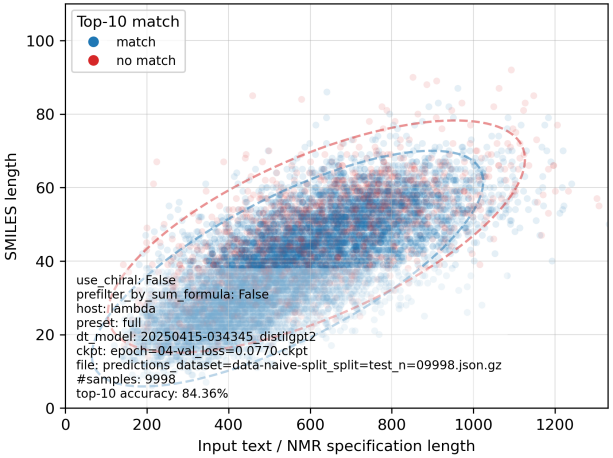
- Prediction confidence (log-likelihood of the sequence)
- Accuracy depending on SMILES length
- Diversity of predicted SMILES that serialize the correct structure
- Confidence mid-SMILES during inference
- Saliency maps: "attention" to SMILES syntax *vs.* molecular structure
- Dataset bias: in-patent SMILES similarity

Prediction confidence (log-likelihood of the sequence)



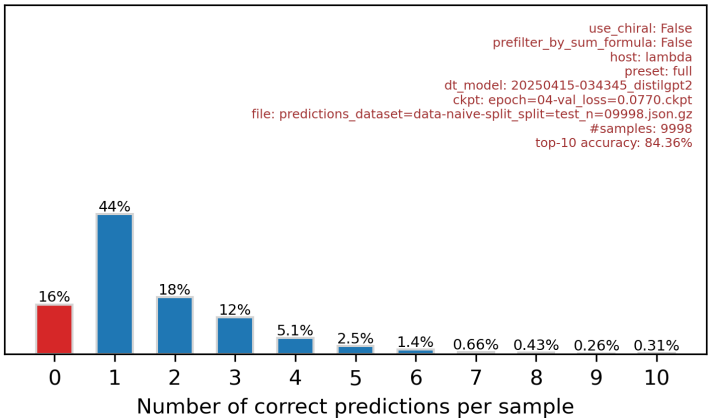
The model is more confident on correct predictions

Accuracy depending on SMILES length



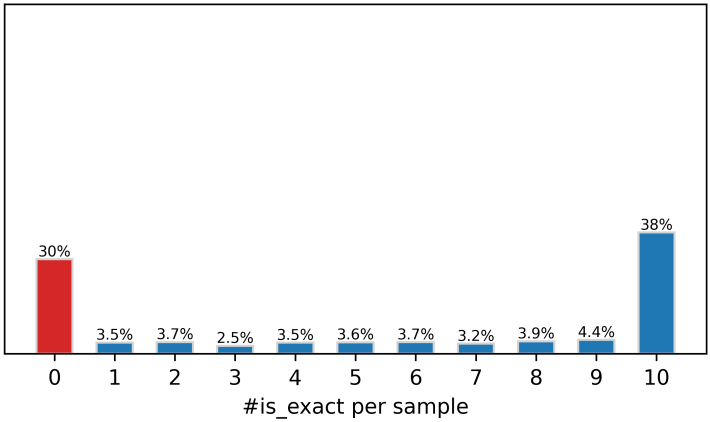
SMILES without a top–10 match are on average ∼10 characters longer

Diversity of correctly predicted SMILES



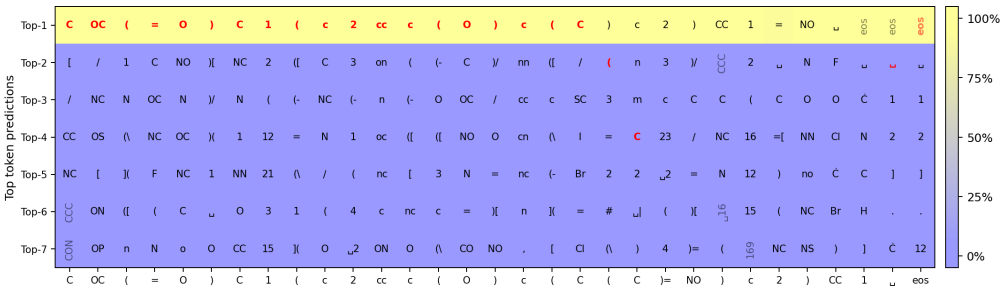
The model is unable to serialize molecules in alternate ways

Diversity of correctly predicted SMILES



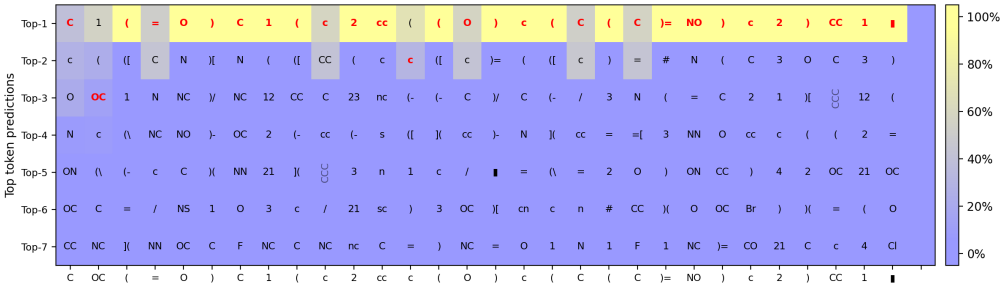
upcoming model...

Confidence mid-SMILES during inference



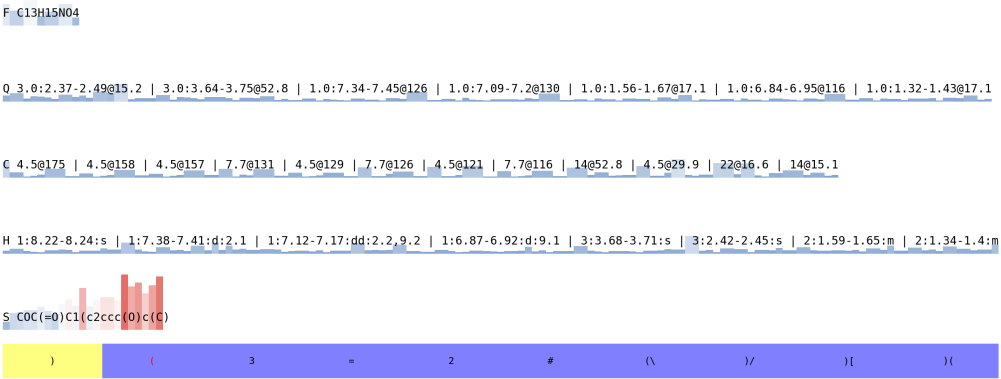
The model is confidently wrong — not a good prior for search

Confidence mid-SMILES during inference



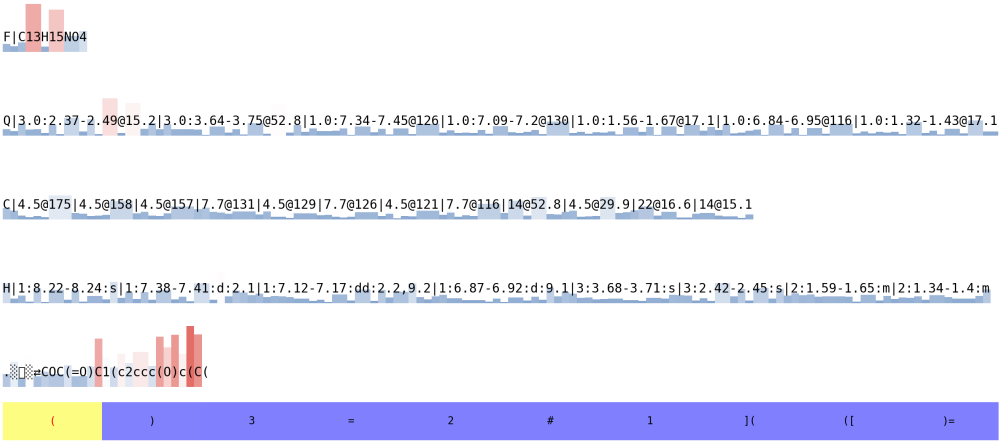
upcoming model...

Saliency: SMILES syntax *vs.* molecular structure



The model may be focused on syntax rather than molecular structure

Saliency: SMILES syntax *vs.* molecular structure



upcoming model...

In-patent SMILES similarity

- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]

[1] M Alberts et al. "Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry". *NeurIPS 2024 Datasets and Benchmarks Track*. 2024

[3] D Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017

- US3956269 from 1976 has 26 compounds (~80% quantile), including
- The dataset is difficult to disentangle into a train/validation/test split, because of *similarity within* and *overlap between* patents
- In particular:
  - short SMILES are intrinsically easier to infer, while
  - long SMILES are made easier by in-patent association

In-patent SMILES similarity

- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]

for example:

- US3956269 from 1976 has 26 compounds (~80% quantile), including
- The dataset is difficult to disentangle into a train/validation/test split, because of *similarity within* and *overlap between* patents
- In particular:
  - short SMILES are intrinsically easier to infer, while
  - long SMILES are made easier by in-patent association

In-patent SMILES similarity

- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]
- US3956269 from 1976 has 26 compounds (~80% quantile), including
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(=O)CCCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(C)(C)C)cc1`
  - `Cc1ccc(OC(=O)CCC(=O)OOC(=O)CCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)CCC(=O)OOC(C)(C)C)cc1`
- The dataset is difficult to disentangle into a train/validation/test split, because of *similarity within* and *overlap between* patents
- In particular:
  - short SMILES are intrinsically easier to infer, while
  - long SMILES are made easier by in-patent association

In-patent SMILES similarity

- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]
- US3956269 from 1976 has 26 compounds (~80% quantile), including
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(=O)CCCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(·····C···)·····(C)··C)cc1`
  - `Cc1ccc(OC(=O)·CCC(=O)OOC(=O)·CCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)·CCC(=O)OOC(·····C···)·····(C)··C)cc1`
- The dataset is difficult to disentangle into a train/validation/test split, because of *similarity within* and *overlap between* patents
- In particular:
  - short SMILES are intrinsically easier to infer, while
  - long SMILES are made easier by in-patent association

In-patent SMILES similarity

- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]
- US3956269 from 1976 has 26 compounds (~80% quantile), including
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(=O)CCCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(·····C···)·····(C)··C)cc1`
  - `Cc1ccc(OC(=O)·CCC(=O)OOC(=O)·CCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)·CCC(=O)OOC(·····C···)·····(C)··C)cc1`
- The dataset is difficult to disentangle into a train/validation/test split, because of *similarity within* and *overlap between* patents
  - They didn't, as far as we can tell
  - We embedded SMILES and used k-means to split ($\rightarrow$ appendix)
- In particular:
  - short SMILES are intrinsically easier to infer, while
  - long SMILES are made easier by in-patent association

In-patent SMILES similarity

- Alberts et al. [1] simulated spectra for ~795k molecules from USPTO [3]
- US3956269 from 1976 has 26 compounds (~80% quantile), including
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(=O)CCCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)CCCC(=O)OOC(·····C···)·····(C)··C)cc1`
  - `Cc1ccc(OC(=O)·CCC(=O)OOC(=O)·CCC(=O)Oc2ccc(C)cc2)cc1`
  - `Cc1ccc(OC(=O)·CCC(=O)OOC(·····C···)·····(C)··C)cc1`
- The dataset is difficult to disentangle into a train/validation/test split, because of *similarity within* and *overlap between* patents
- In particular:
  - short SMILES are intrinsically easier to infer, while
  - long SMILES are made easier by in-patent association

⚠ The ~85% top–10 accuracy is very scenario-specific

⚠ The model learns chunks of SMILES – not molecular structure

Graph representation integrating multiple alternate chemical equivalents

GRIMACE

c1ccc(C(=O)O)c(c1)OC(C)=O

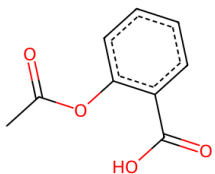
CC(=O)Oc1c(cccc1)C(O)=O

CC(=O)Oc1c(C(O)=O)cccc1

c1cc(C(=O)O)c(OC(=O)C)cc1

...

how many in total?



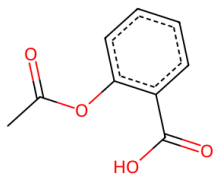
c1ccc(C(=O)O)c(c1)OC(C)=O

CC(=O)Oc1c(cccc1)C(O)=O

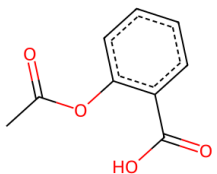
CC(=O)Oc1c(C(O)=O)cccc1

c1cc(C(=O)O)c(OC(=O)C)cc1

...



`c1ccc(C(=O)O)c(c1)OC(C)=O`
`CC(=O)Oc1c(cccc1)C(O)=O`
`CC(=O)Oc1c(C(O)=O)cccc1`
`c1cc(C(=O)O)c(OC(=O)C)cc1`
...
304 distinct serializations



"restricted" by `rdkit` to sensible variants

- It makes sense to train

  random serialization $\rightarrow$ molecular properties

  as data augmentation and regularization [8]
- Does it make sense to train

  NMR data $\rightarrow$ random serialization – ?
- Consider two serializations, shown to the model at random:

  `c1(ccccc1OC(=O)C)C(O)=O` and `c1(ccccc1OC(C)=O)C(O)=O`
- The model will eventually learn the next-token probabilities

  $\mathbb{P}(\texttt{=} \mid \text{prefix}) \approx 50\%$ and $\mathbb{P}(\texttt{C} \mid \text{prefix}) \approx 50\%$
- This training signal is weak, inconsistent and incomplete
- Key idea:

- It makes sense to train

  random serialization $\rightarrow$ molecular properties

- Does it make sense to train

  NMR data $\rightarrow$ random serialization

  where the serialization changes from epoch to epoch?

- Consider two serializations, shown to the model at random:

  `c1(ccccc1OC(=O)C)C(O)=O` and `c1(ccccc1OC(C)=O)C(O)=O`

- The model will eventually learn the next-token probabilities

  $\mathbb{P}(\texttt{=} \mid \text{prefix}) \approx 50\%$ and $\mathbb{P}(\texttt{C} \mid \text{prefix}) \approx 50\%$

- This training signal is weak, inconsistent and incomplete
- Key idea:

- It makes sense to train

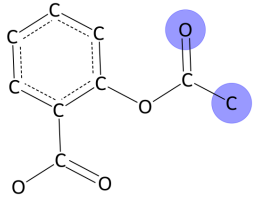
  random serialization $\longrightarrow$ molecular properties

- Does it make sense to train

  NMR data $\longrightarrow$ random serialization – ?

- Consider two serializations, shown to the model at random:

`c1(ccccc1OC(=O)C)C(O)=O` and `c1(ccccc1OC(C)=O)C(O)=O`



- The model will eventually learn the next-token probabilities

$\mathbb{P}(\boxed{=} \mid \text{prefix}) \approx 50\%$ and $\mathbb{P}(\boxed{C} \mid \text{prefix}) \approx 50\%$

- It makes sense to train

$$\text{random serialization} \rightarrow \text{molecular properties}$$

- Does it make sense to train

$$\text{NMR data} \rightarrow \text{random serialization} \; - \; ?$$

- Consider two serializations, shown to the model at random:

`c1(ccccc1OC(=O)C)C(O)=O` and `c1(ccccc1OC(C)=O)C(O)=O`

- The model will eventually learn the next-token probabilities

$$\mathbb{P}(\boxed{=} \mid \underline{\text{prefix}}) \approx 50\% \quad \text{and} \quad \mathbb{P}(\boxed{\texttt{C}} \mid \underline{\text{prefix}}) \approx 50\%$$

but:

- This training signal is weak, inconsistent and incomplete
- Key idea:

- It makes sense to train

  random serialization $\rightarrow$ molecular properties

- Does it make sense to train

  NMR data $\rightarrow$ random serialization – ?

- Consider two serializations, shown to the model at random:

  `c1(ccccc1OC(=O)C)C(O)=O` and `c1(ccccc1OC(C)=O)C(O)=O`

- The model will eventually learn the next-token probabilities

  $\mathbb{P}(\boxed{=} \mid \text{prefix}) \approx 50\%$ and $\mathbb{P}(\boxed{\texttt{C}} \mid \text{prefix}) \approx 50\%$

- This training signal is weak, inconsistent and incomplete:
  - pulls one token at a time,
  - we can't cycle through many variants.
- Key idea:

- It makes sense to train

$$\text{random serialization} \rightarrow \text{molecular properties}$$

- Does it make sense to train

$$\text{NMR data} \rightarrow \text{random serialization} \; - \; ?$$

- Consider two serializations, shown to the model at random:

`c1(ccccc1OC(=O)C)C(O)=O` and `c1(ccccc1OC(C)=O)C(O)=O`

- The model will eventually learn the next-token probabilities

$$\mathbb{P}(\texttt{=} \mid \text{prefix}) \approx 50\% \quad \text{and} \quad \mathbb{P}(\texttt{C} \mid \text{prefix}) \approx 50\%$$

- This training signal is weak, inconsistent and incomplete
- Key idea: Let's supervise on the next-token distribution directly

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them
- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1

- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O

- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O

- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	C	)=	O	)	C	(	O	)=	O	

- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	C	)=	O	)	C	(	O	)=	O	

...

- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	C	)=	O	)	C	(	O	)=	O	

...

- ...until a stopping criterion is met ($\rightarrow$ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	C	)=	O	)	C	(	O	)=	O	

...

- ...until a stopping criterion is met ($\rightarrow$ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O
c	1	(	cc	cc	c	1	OC	(	C	)=	O	)	C	(	O	)=	O	

...

- ...until a stopping criterion is met ($\rightarrow$ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

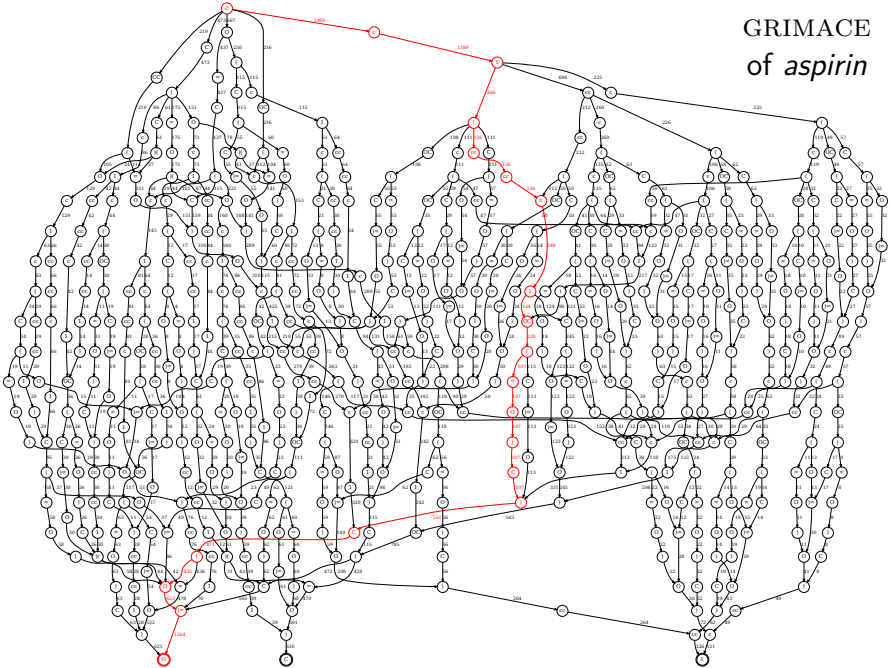
- Key idea: Let's supervise on the next-token distribution directly
- Sample serializations of the same molecule and tokenize them

	CC	(	=	O	)	O	c	1	c	(	C	(	O	)=	O	)	cc	cc	1
	CC	(	=	O	)	O	c	1	c	(	cc	cc	1	)	C	(	O	)=	O
→	c	1	(	cc	cc	c	1	OC	(	=	O	)	C	)	C	(	O	)=	O
	c	1	(	cc	cc	c	1	OC	(	C	)=	O	)	C	(	O	)=	O	

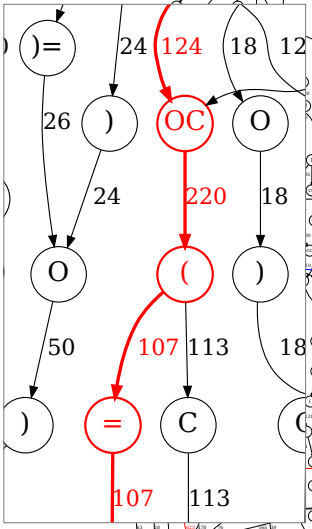
...

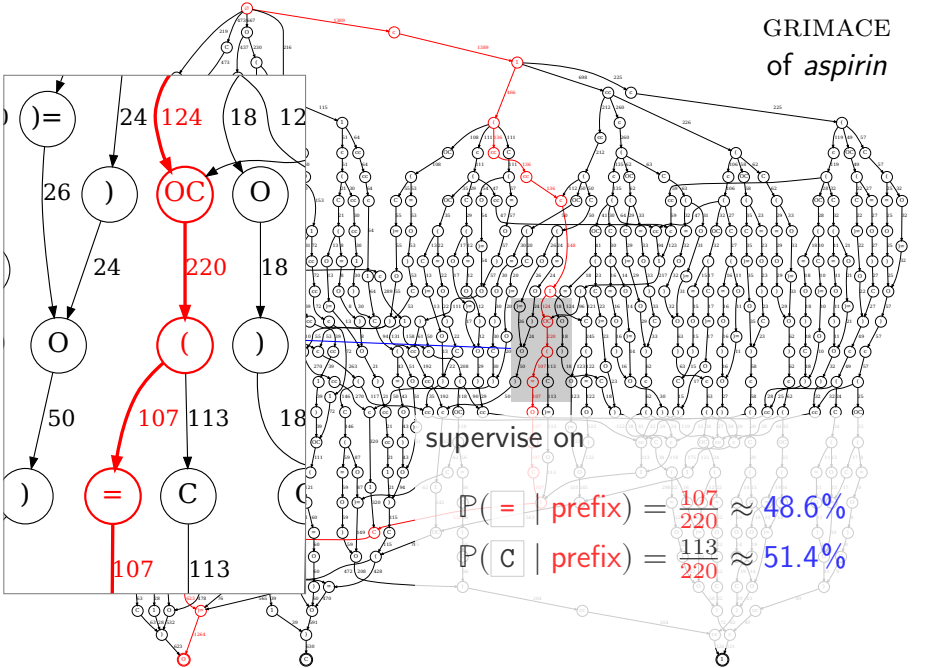
- ...until a stopping criterion is met (→ appendix)
- Construct the word tree (merge common prefixes and suffixes)
- Count samples passing each out-edge and normalize to probabilities
- To train, pick one path per epoch, supervise on those probabilities

GRIMACE
of *aspirin*



GRIMACE
of *aspirin*



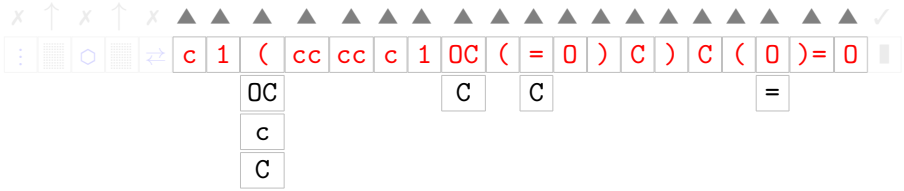


supervise on

$$\mathbb{P}(\boxed{=} \mid \text{prefix}) = \frac{107}{220} \approx 48.6\%$$

$$\mathbb{P}(\boxed{\texttt{C}} \mid \text{prefix}) = \frac{113}{220} \approx 51.4\%$$

We supervise the output with “teacher forcing” on sequences like



where

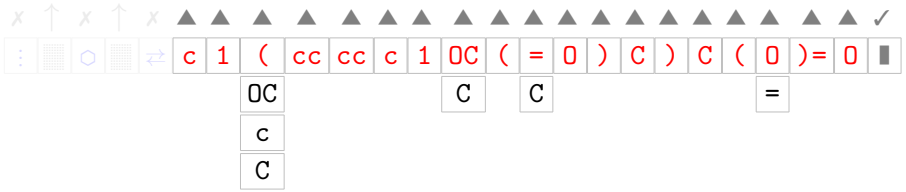
trigger tokens:

- ⋮ – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with "teacher forcing" on sequences like



where

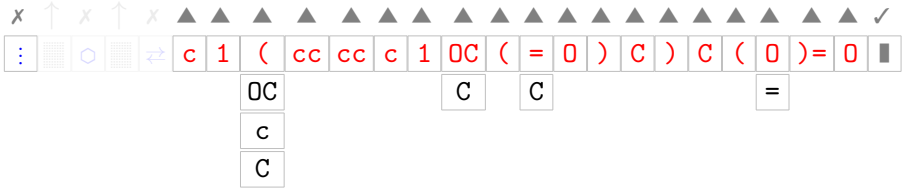
trigger tokens:

- ⋮ – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with “teacher forcing” on sequences like



where

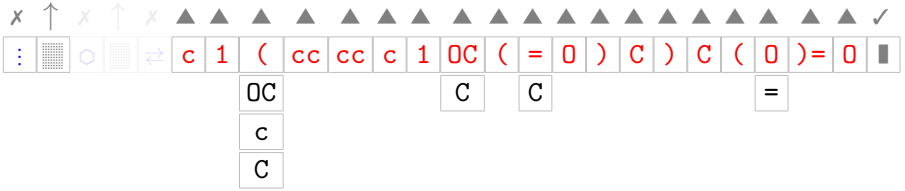
trigger tokens:

- ⋮ – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with “teacher forcing” on sequences like



where

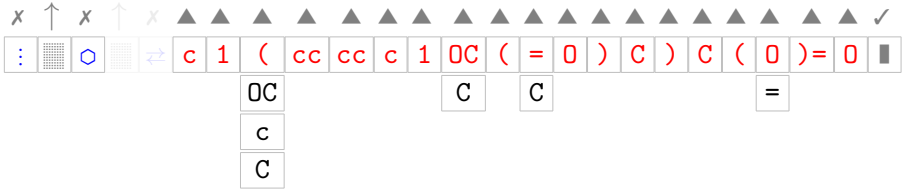
trigger tokens:

- – vector embedding [9]
- – functional group counts
- – GRIMACE
- – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with “teacher forcing” on sequences like



where

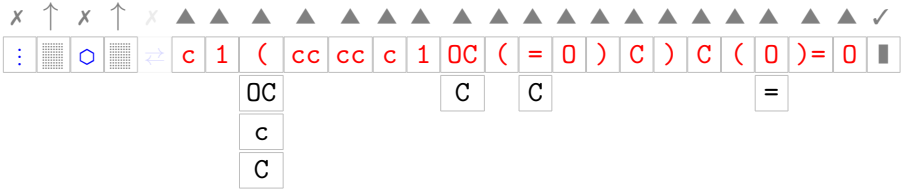
trigger tokens:

- ⋮ – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with “teacher forcing” on sequences like



where

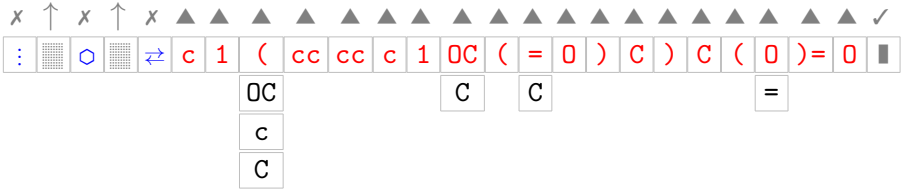
trigger tokens:

- ⋮ – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with "teacher forcing" on sequences like



where

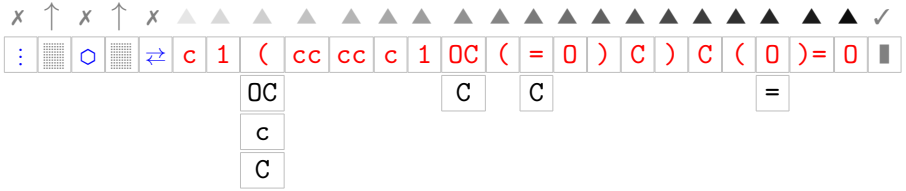
trigger tokens:

- : – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

We supervise the output with “teacher forcing” on sequences like



where

trigger tokens:

- ⋮ – vector embedding [9]
- ⬡ – functional group counts
- ⇄ – GRIMACE
- ▒ – not-a-token

supervision:

- ▲ – KL divergence $\times(1+q)^n$
- ✓ – cross-entropy
- ✗ – none
- ↑ – auxiliary head

Training & results

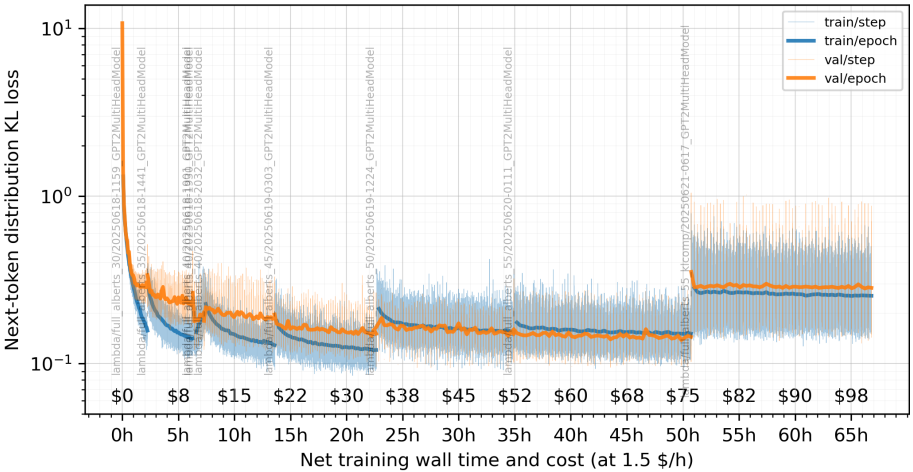
We trained

- the ~83M° DistilGPT2 decoder-only transformer
- on the Alberts et al. [1] dataset – up to ~560k training samples
- on an NVIDIA GH200 – ARM64 + H100: 96 GB VRAM, 64 vCPUs, 432 GiB RAM
- to infer

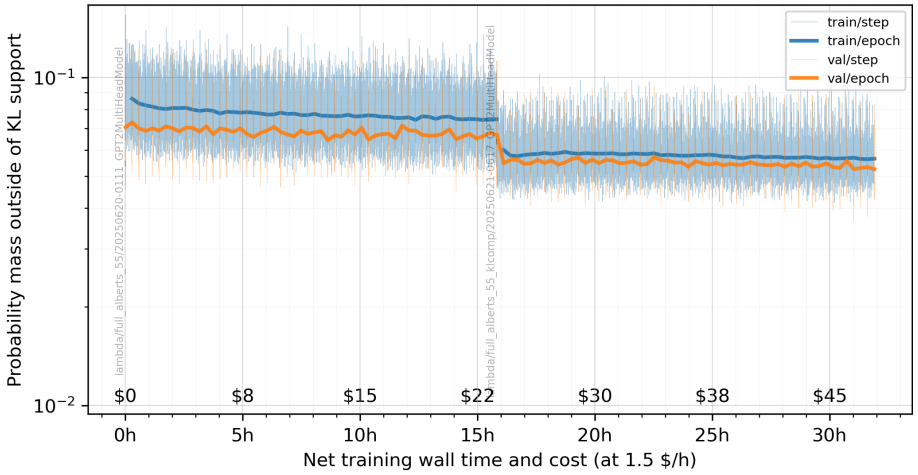
$$\mathrm{F} + \mathrm{Q} + \mathrm{C} + \mathrm{H} \rightarrow$$  GRIMACE

- through a curriculum of increasing SMILES length:
  1. 30 epochs on $|\mathrm{SMILES}| \leq 30$ from the pre-trained state
  2. 35 epochs on $|\mathrm{SMILES}| \leq 35$
  3. 40 epochs on $|\mathrm{SMILES}| \leq 40$
  4. 45 epochs on $|\mathrm{SMILES}| \leq 45$
  5. 50 epochs on $|\mathrm{SMILES}| \leq 50$
  6. 55 epochs on $|\mathrm{SMILES}| \leq 55$
  7. 55 epochs on $|\mathrm{SMILES}| \leq 55$ with KL geometric weight 5%

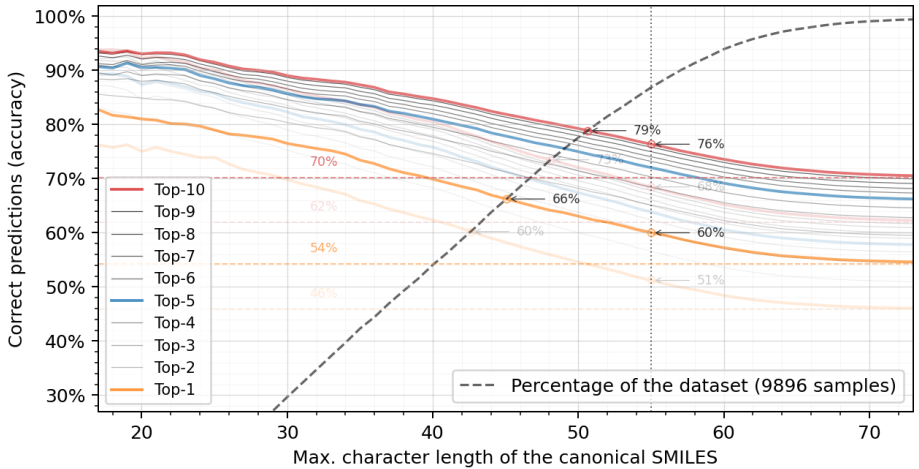
KL loss during training runs 1 to 6 (no KL weight) and 7 (with KL weight)



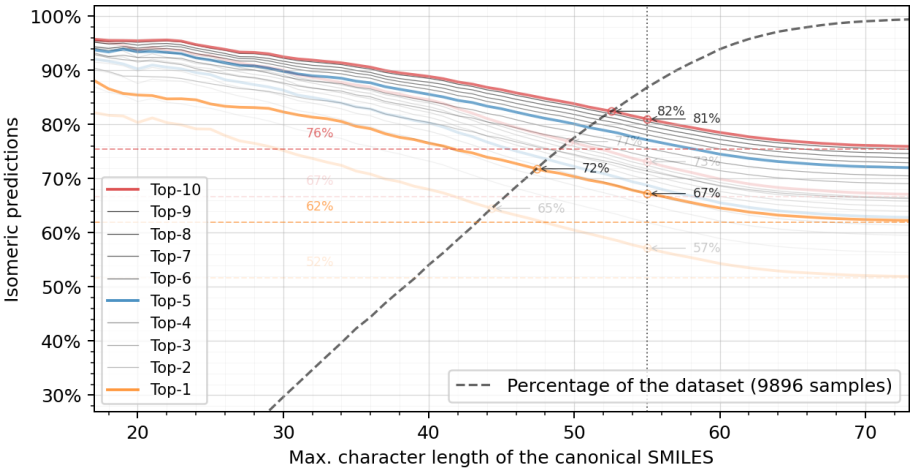
Off-support loss during training (last two runs: effect of KL weight)



Accuracy on the test set (exact match)

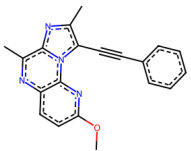


Accuracy on the test set (stereo-isomer match – ignoring stereochemistry)

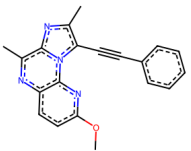


Example: the top–10 predictions for a test sample

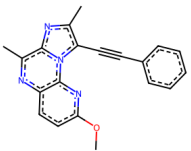
c1(C)c2n(c(C#Cc3ccccc3)c(C)n2)c2c(ccc(n2)OC)n1



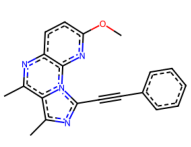
c1(C)c2n(c(C#Cc3ccccc3)c(C)n2)c2c(ccc(OC)n2)n1



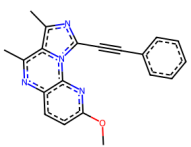
c1(C)c2n(c(C#Cc3ccccc3)c(n2)C)c2c(ccc(n2)OC)n1



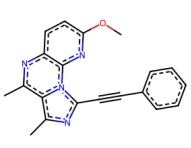
C(#Cc1ccccc1)c1n2c(c(C)nc3c2nc(cc3)OC)c(C)n1



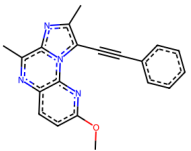
C(#Cc1ccccc1)c1n2c3c(ccc(n3)OC)nc(c2c(C)n1)C



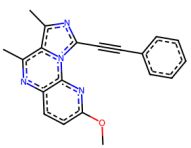
c1ccccc1C#Cc1n2c(c(C)nc3c2nc(cc3)OC)c(C)n1



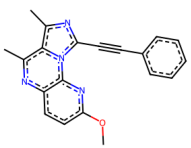
c1(C)c2n(c(C#Cc3ccccc3)c(n2)C)c2c(ccc(OC)n2)n1



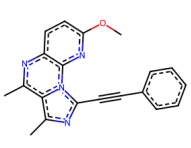
C(#Cc1ccccc1)c1n2c(c(C)nc3ccc(nc32)OC)c(C)n1



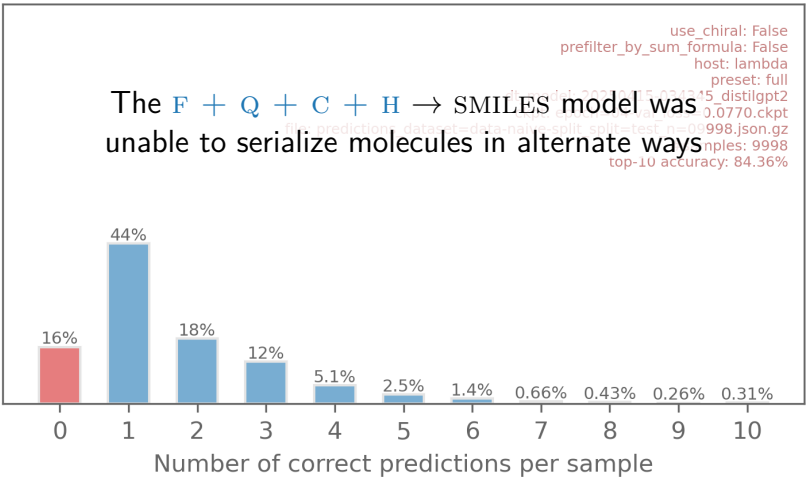
C(#Cc1ccccc1)c1n2c(c(C)nc3ccc(nc23)OC)c(C)n1



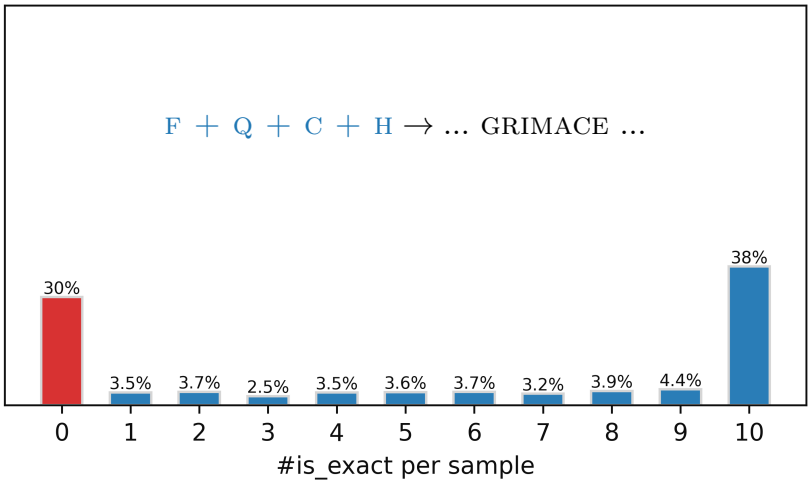
c1cc(cccl)C#Cc1n2c(c(C)nc3c2nc(cc3)OC)c(C)n1



Diversity of predictions

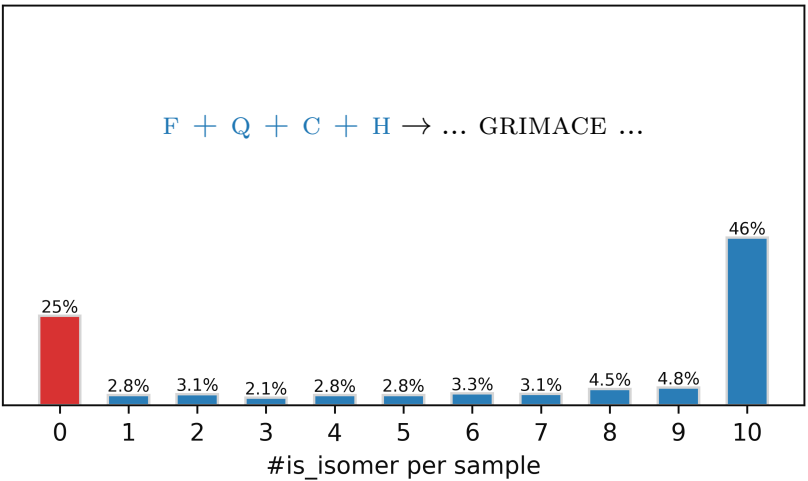


Diversity of predictions



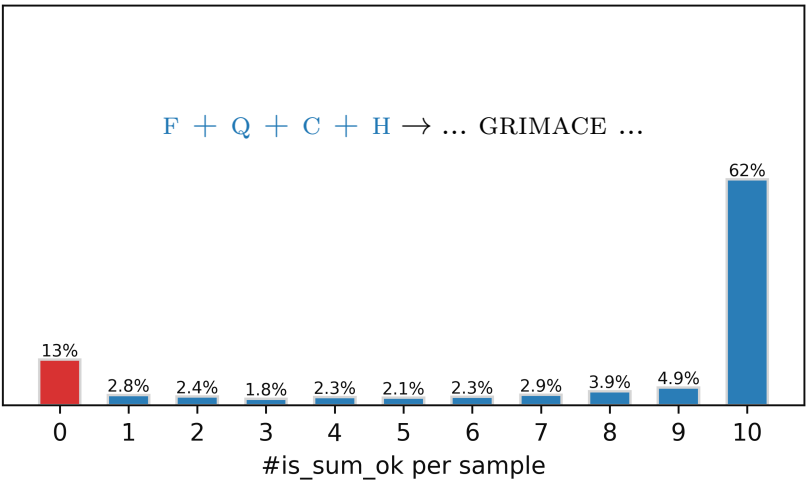
In 38% of test samples, all 10 predictions are exact

Diversity of predictions



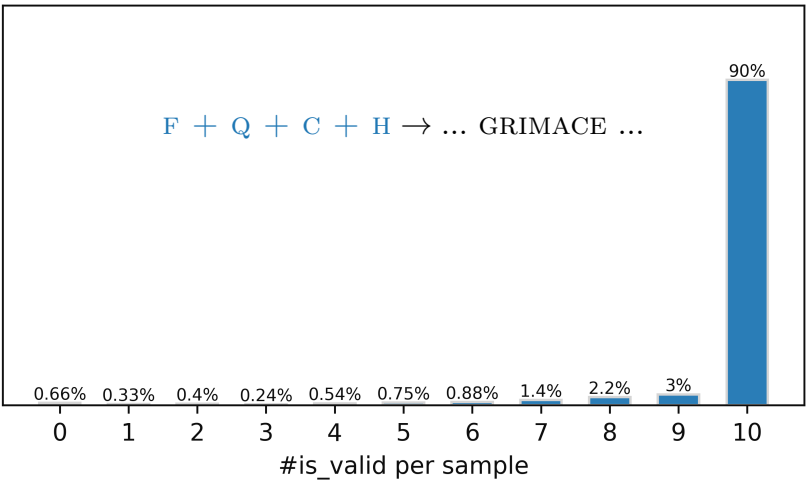
Stereo-isomer = equal connectivity (modulo 3d flags)

Diversity of predictions



How often is the chemical sum formula correct?

Diversity of predictions



How often is the generated SMILES valid?

Cf. Fig. 9 from the SI to Hu et al. [6]:

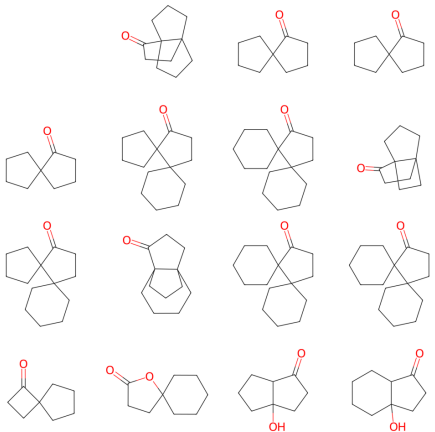


Figure 9: An example of one of the molecules correctly predicted by the substructure-to-structure transformer model (top left, black box) and all 15 structures generated by the model.

Cf. Fig. 9 from the SI to Hu et al. [6]:

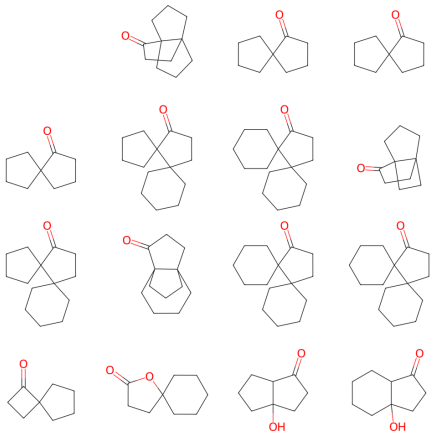


Figure 9: An example of one of the molecules correctly predicted by the substructure-to-structure transformer model (top left, black box) and all 15 structures generated by the model.

Cf. Fig. 9 from the SI to Hu et al. [6]:

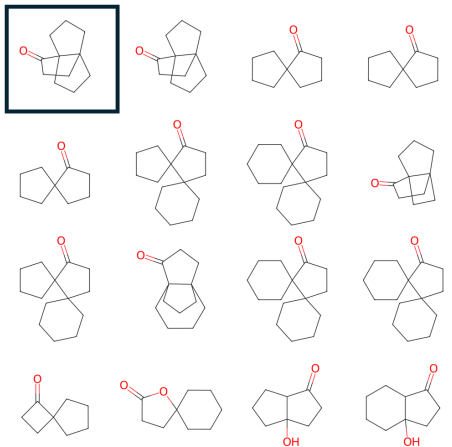
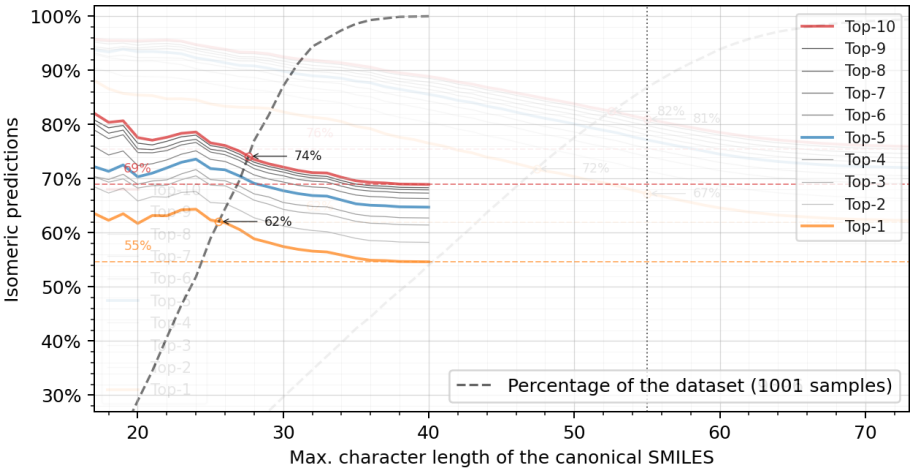
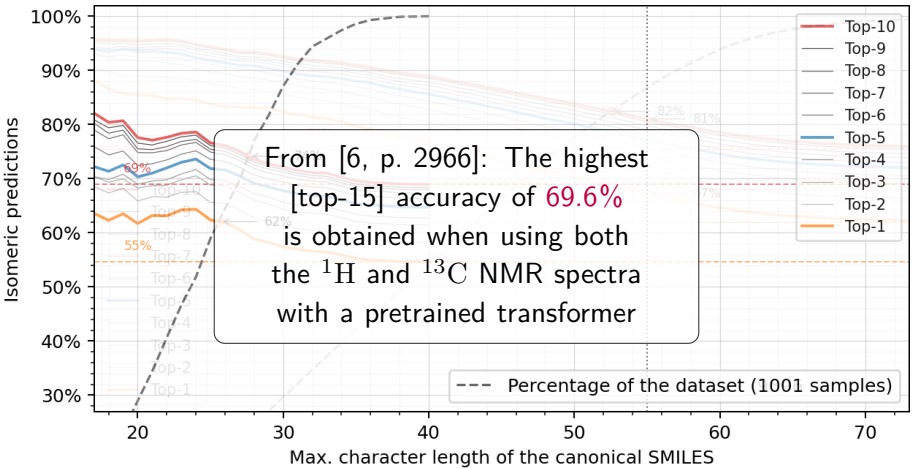


Figure 9: An example of one of the molecules correctly predicted by the substructure-to-structure transformer model (top left, black box) and all 15 structures generated by the model.

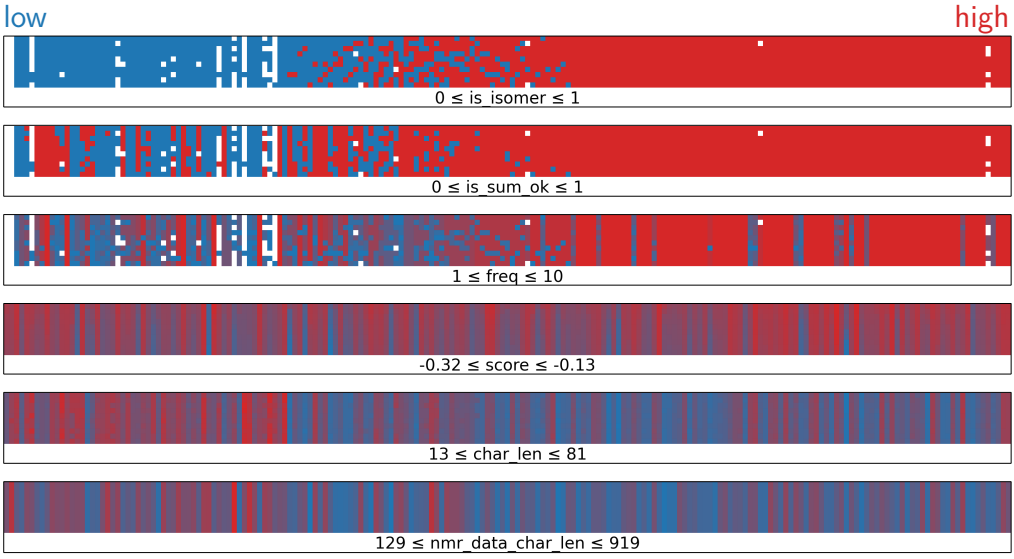
Evaluation on SpectraBase (simulated; as in Hu et al. [6])

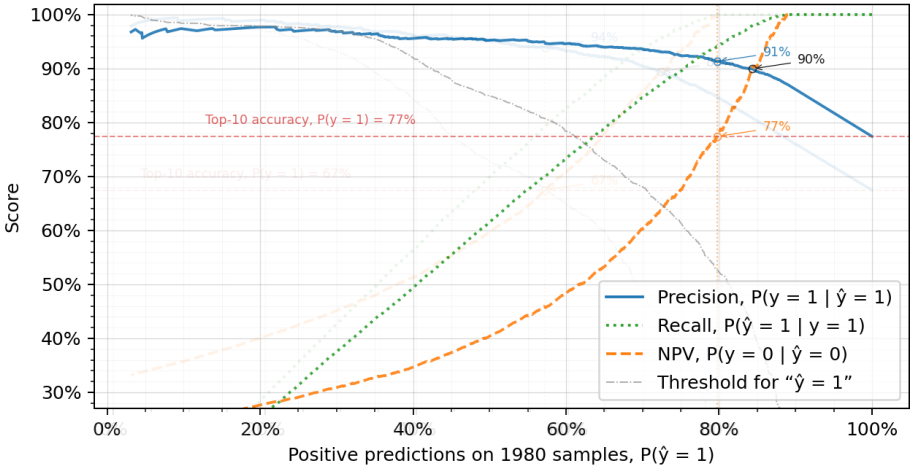


Evaluation on SpectraBase (simulated; as in Hu et al. [6])



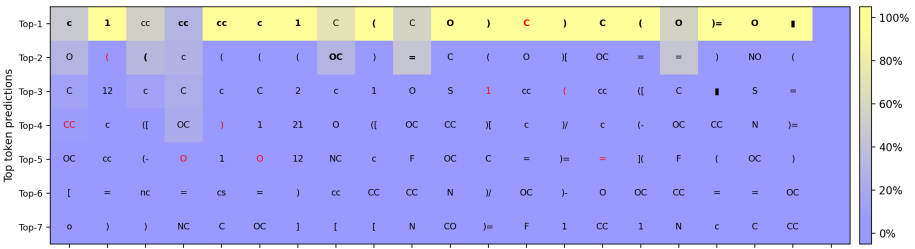
“Flag plot” of 10 hypotheses × 200 test samples



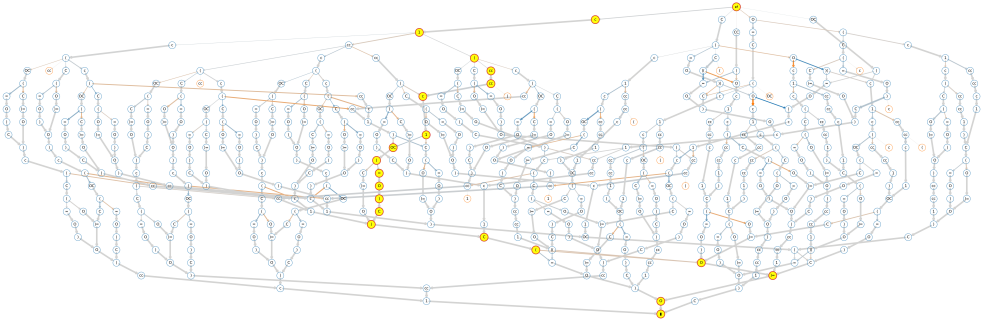


Meta-classifier for $y =$ "Is there a top–10 stereo-isomer match?" has $\mathbb{P}(\hat{y} = y) \approx 90\%$

Aspirin


Top token predictions
Top-1 c 1 cc cc cc c 1 C (C O) C) C (O)= O ▮
Top-2 O ((c (((OC) = C (O)[OC = =) NO (
Top-3 C 12 c C c C 2 c 1 O S 1 cc (cc ([C ▮ S =
Top-4 CC c ([OC) 1 21 O ([OC CC][c)/ c (- OC CC N)=
Top-5 OC cc (- O 1 O 12 NC c F OC C =)= =][F (OC)
Top-6 [= nc = cs =) cc CC CC N)/ OC)- O OC CC = = OC
Top-7 o)) NC C OC] [[N CO)= F 1 CC 1 N c C CC
CC (= O) O c 1 cc cc c 1 C (= O) O ▮
100%
80%
60%
40%
20%
0%

Aspirin



Recap & outlook

Graph representation integrating multiple alternate chemical equivalents is a novel* way of teaching chemical structures to language transformers

It allows the model to express uncertainty

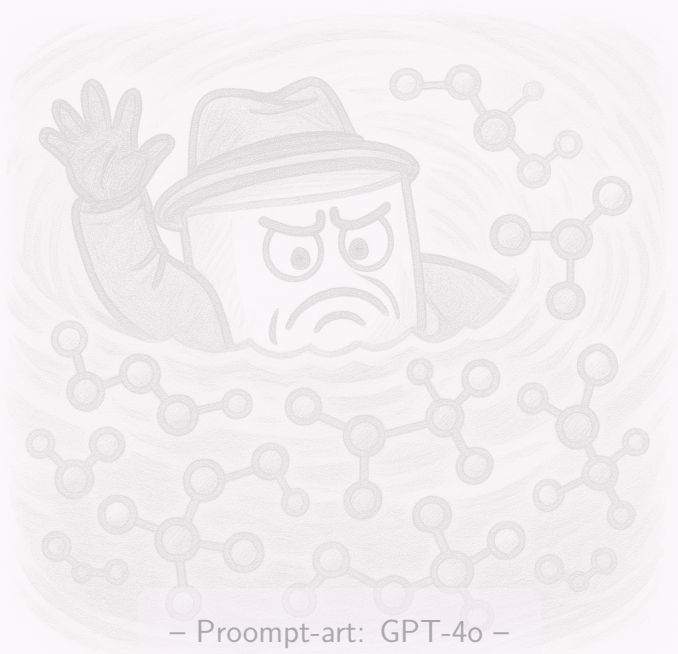
The multimodal NMR annotation → GRIMACE multi-task transformer achieves ~80% top–10 accuracy on ~80% of the synthetic dataset (and we can tell with ~90% confidence)

Spectroscopy:

- A data-driven model of variability in experimental NMR spectra
- Cycle-consistent search with "NMR $\rightarrow$ GRIMACE" as a prior

GRIMACE:

- Optical hand-written molecule recognition
- A spectrum-aware auto-encoder for molecules
- An efficient way to compute the reference GRIMACE



– Proompt-art: GPT-4o –

References I

[1] M Alberts et al. "Unraveling molecular structure: A multimodal spectroscopic dataset for chemistry". *NeurIPS 2024 Datasets and Benchmarks Track*. 2024.

[2] B Hong et al. "Biosynthesis of strychnine". *Nature* 607 (2022), pp. 617–622.

[3] D Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017.

[4] M Alberts, F Zipoli, and AC Vaucher. "Learning the language of NMR: Structure elucidation from NMR spectra using transformer models". *ChemRxiv* (2023).

[5] M Alberts, N Hartrampf, and T Laino. "Automated structure elucidation at human-level accuracy via a multimodal multitask language model". *ChemRxiv* (2025).

[6] F Hu et al. "Accurate and efficient structure elucidation from routine one-dimensional NMR spectra using multitask machine learning". *ACS Central Science* 10.11 (2024), pp. 2162–2170.

[7] S Kim et al. "PubChem 2025 update". *Nucleic Acids Research* 53.D1 (2025), pp. D1516–D1525.

[8] EJ Bjerrum. "SMILES enumeration as data augmentation for neural network modeling of molecules". *arXiv* 1703.07076 (2017).

[9] W Ahmad et al. "ChemBERTa-2: Towards chemical foundation models". *arXiv* 2209.01712v1 (2022).

[10] E Jonas. "Deep imitation learning for molecular inverse problems". *Advances in Neural Information Processing Systems*. Vol. 32. 2019.

Appendix

The multi-task objective

$$\text{minimize} \quad \sum_{\text{task}} \left(\sigma_{\text{task}}^{-2} \operatorname{softplus}(\text{loss}_{\text{task}}) + \log \sigma_{\text{task}}^2\right)$$

with trainable σ_{task}, consists of the task-specific losses:

- for the "chembedding" head

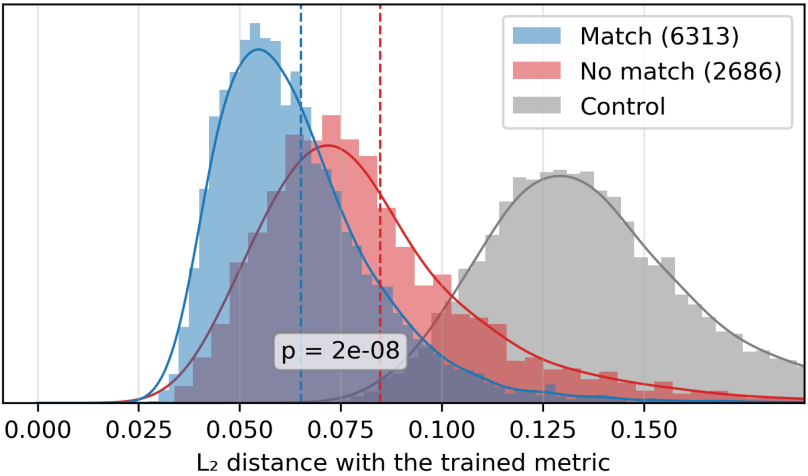
$$\mathbf{e} \quad \mapsto \quad \tfrac{1}{n}\left(\tfrac{1}{2}\|T\mathbf{e}\|_2^2 - \log\det T + \tfrac{n}{2}\log 2\pi\right), \quad n = 384,$$

- for the "$\log(1 + \text{functional group count})$" head

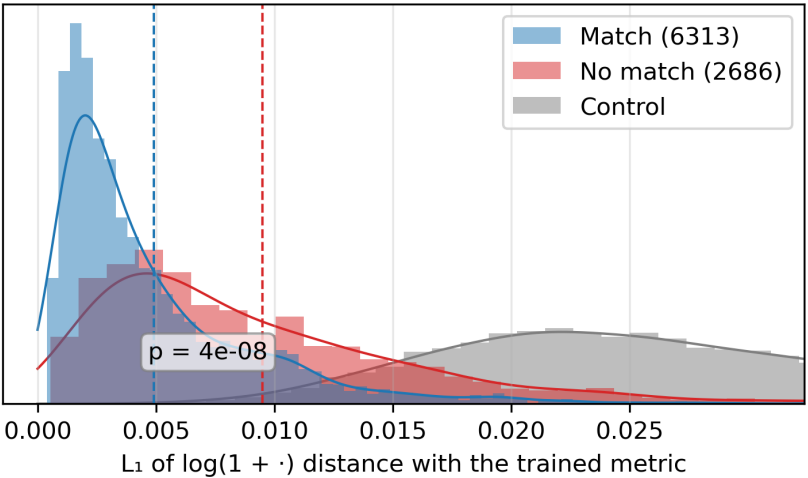
$$\mathbf{f} \quad \mapsto \quad \tfrac{1}{m}\left(\|S\mathbf{f}\|_1 - \log\det S + m\log 2\right), \quad m = 289,$$

- average KL loss for the GRIMACE,
- cross-entropy loss for the end-of-sequence token,

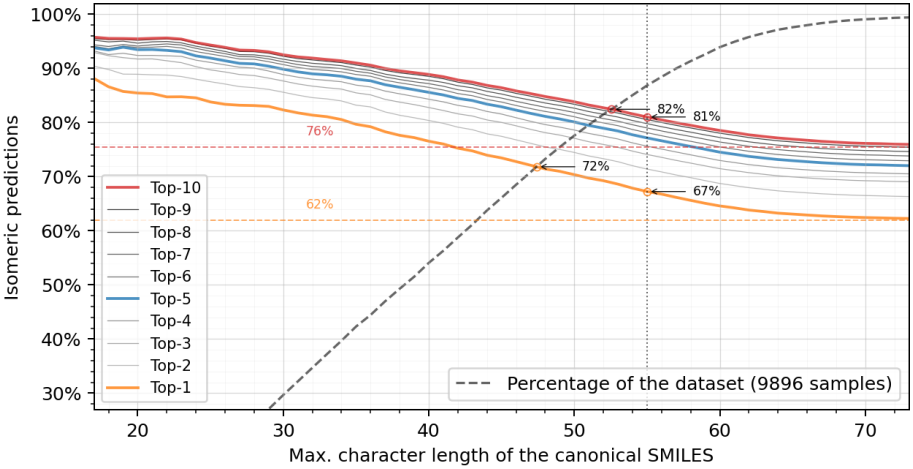
where T and S are trainable matrices.



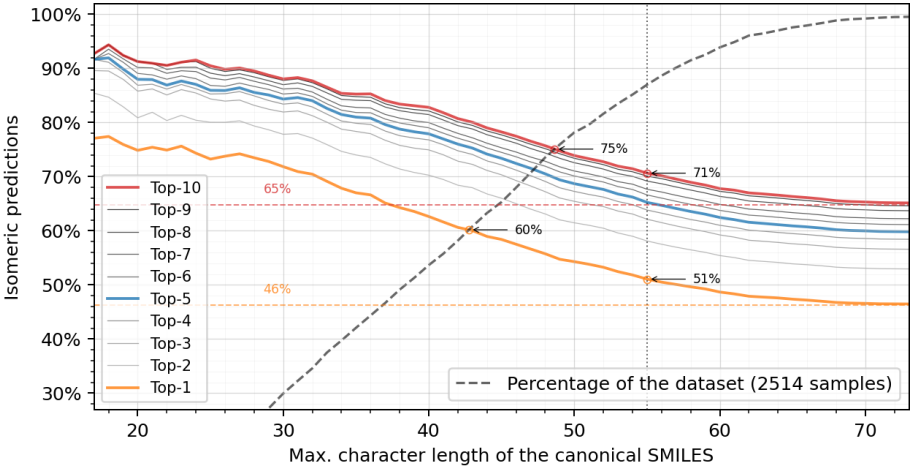
$\frac{1}{\sqrt{n}}\|T \cdot\|_2$-norm, stratified by whether there is a top–10 match



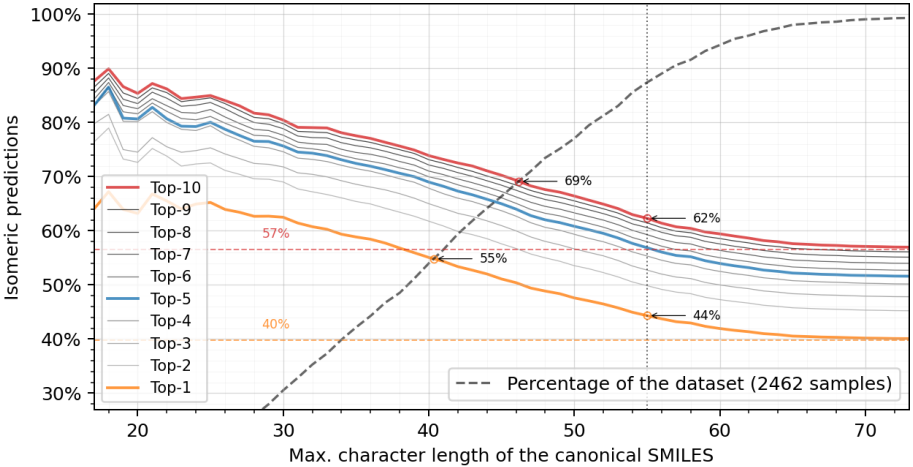
$\frac{1}{m}\|S \cdot\|_1$-norm, stratified by whether there is a top–10 match



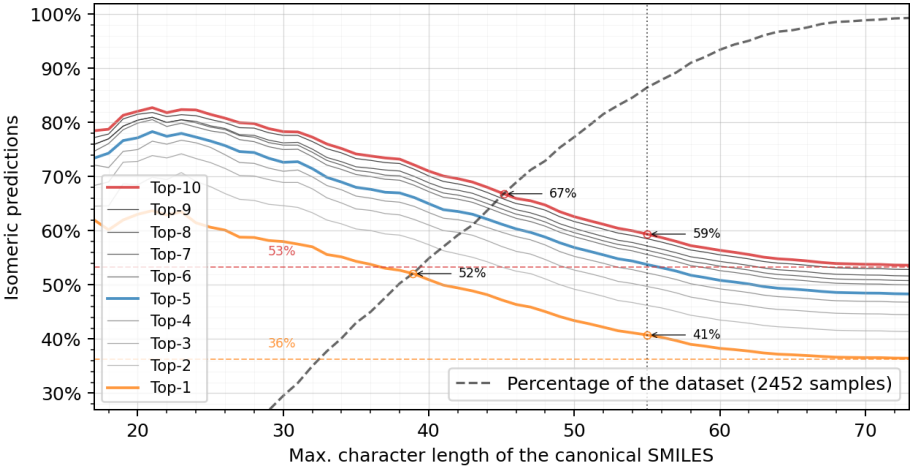
Original training



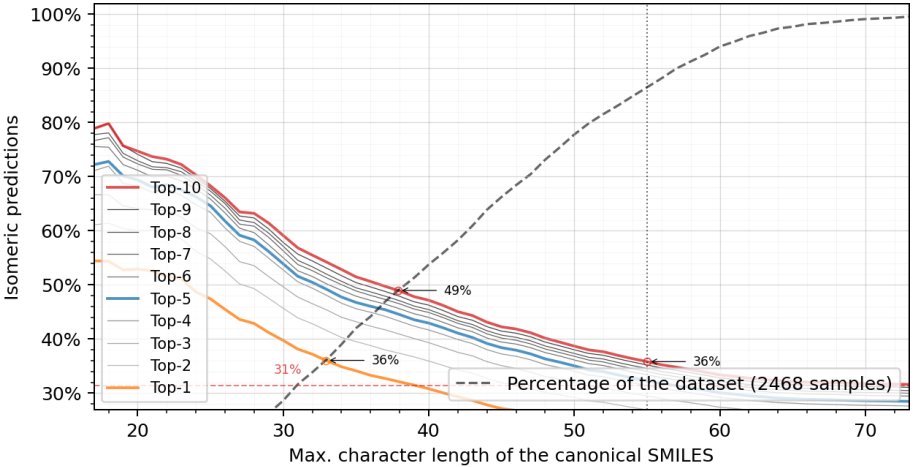
After fine-tuning on the validation set with HSQC / C-NMR dropout



After fine-tuning, evaluated on F + Q + ⊠ + H test set



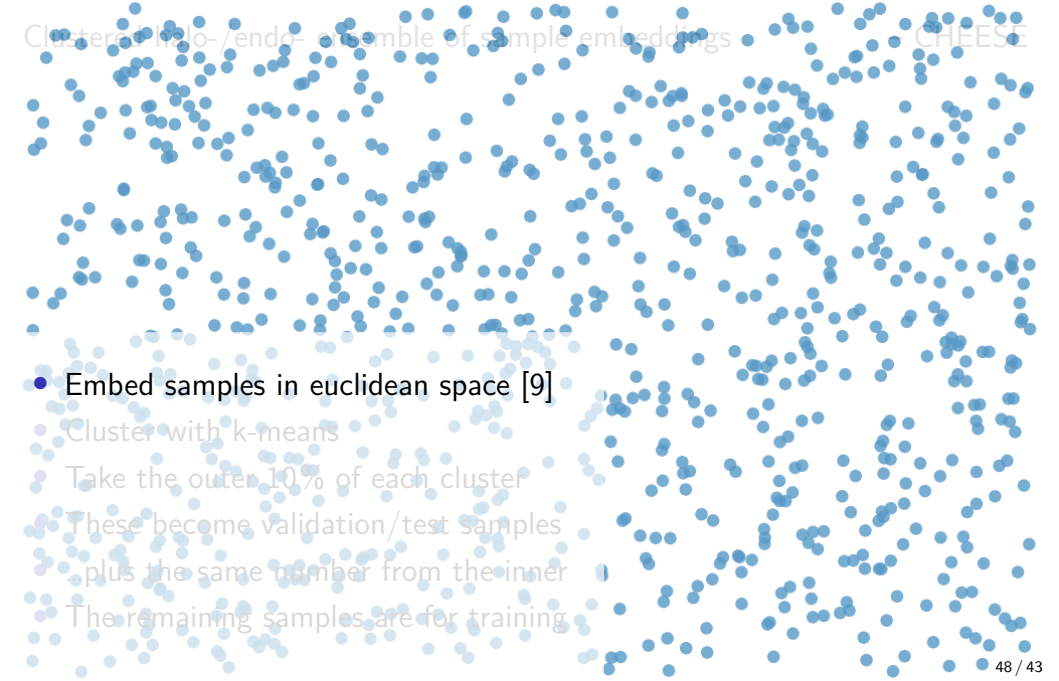
After fine-tuning, evaluated on $\mathrm{F} + \cancel{\mathrm{O}} + \mathrm{C} + \mathrm{H}$ test set



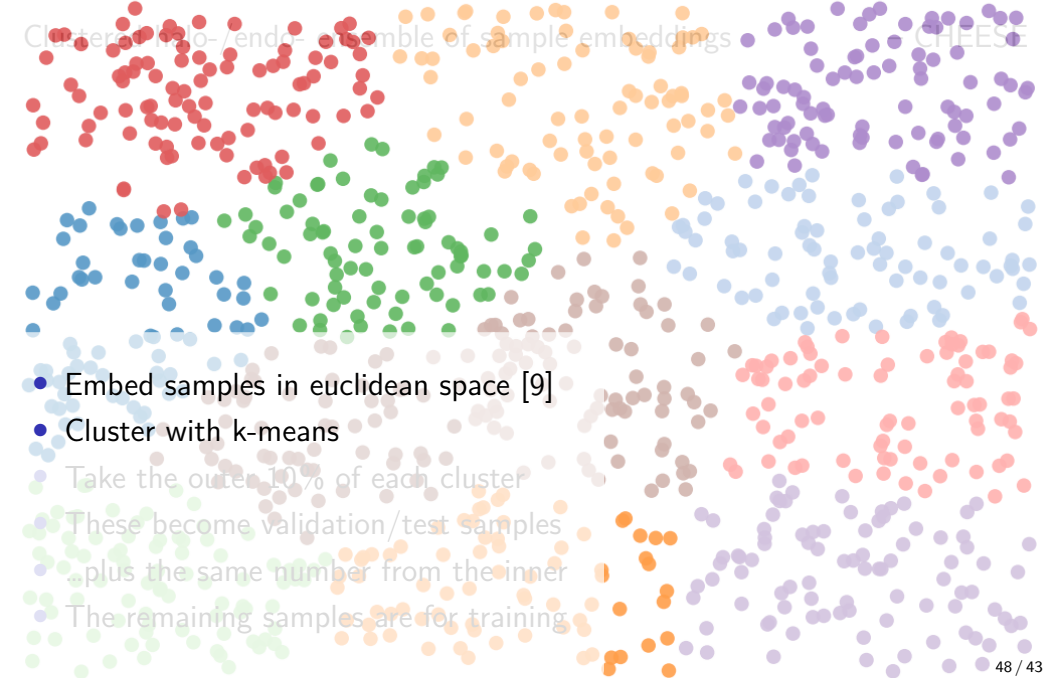
After fine-tuning, evaluated on F + ~~Cl~~ + ~~Br~~ + H test set

Clustered halo-/endo- ensemble of sample embeddings – CHEESE

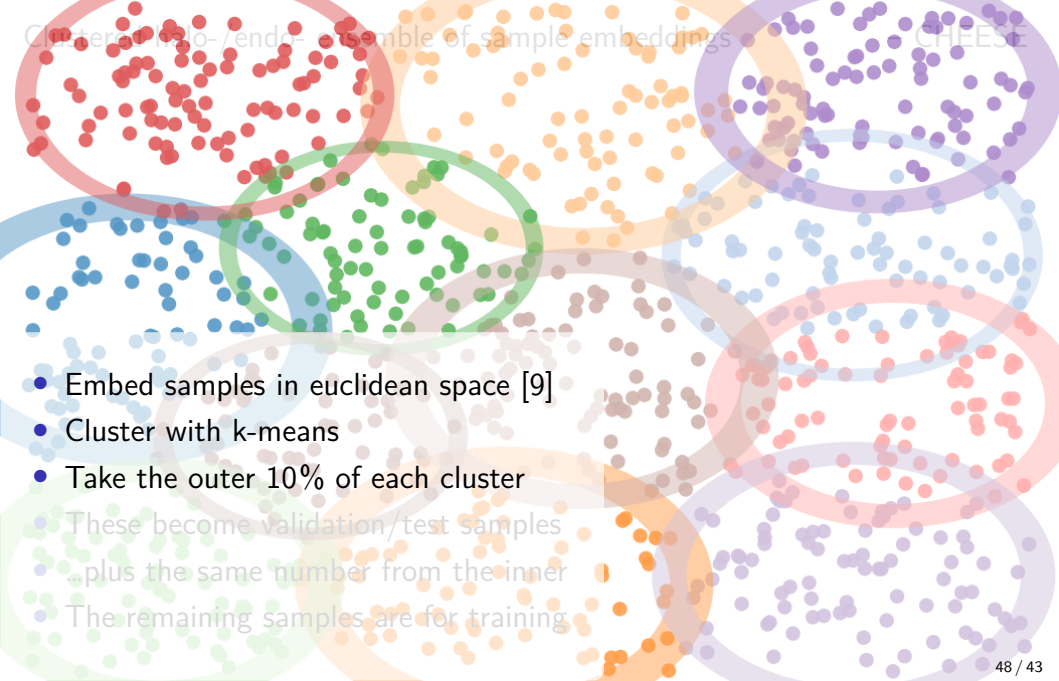
- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- ...plus the same number from the inner
- The remaining samples are for training



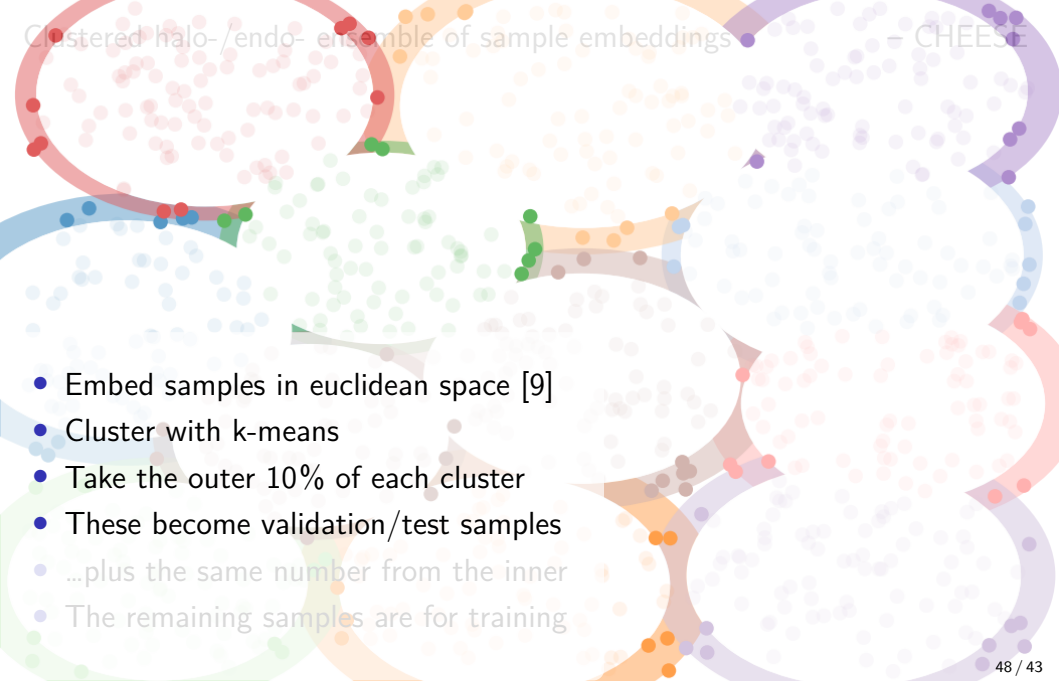
- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- plus the same number from the inner
- The remaining samples are for training



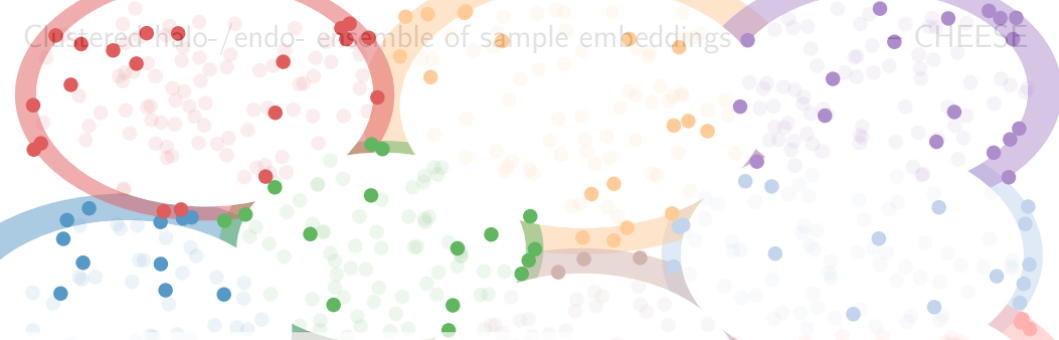
- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- ...plus the same number from the inner
- The remaining samples are for training



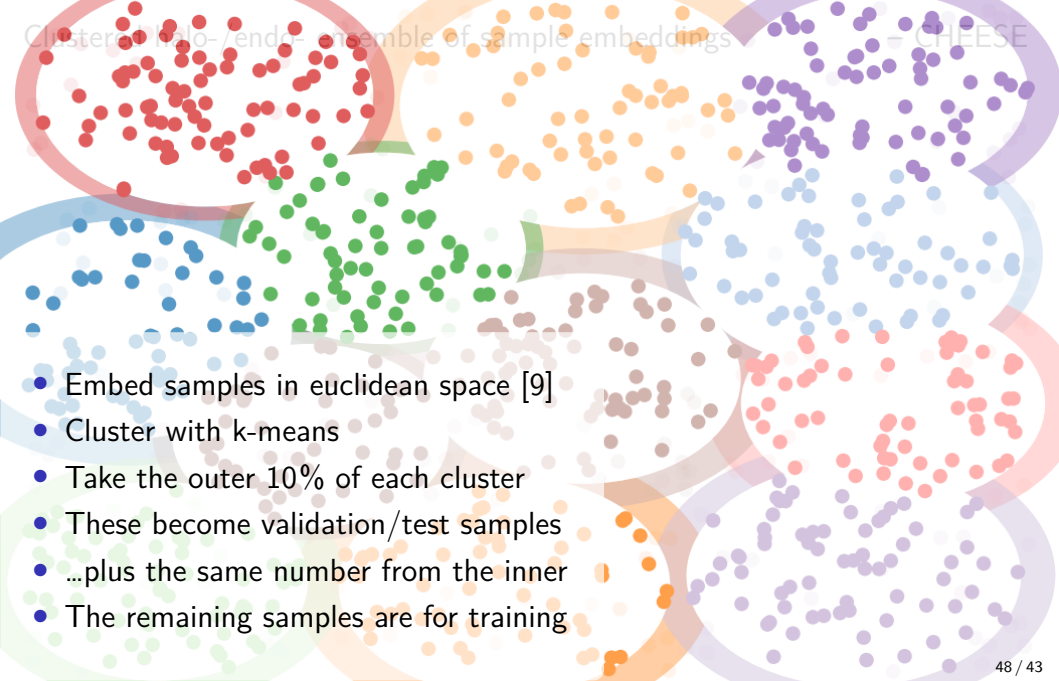
- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- ..plus the same number from the inner
- The remaining samples are for training



- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- ...plus the same number from the inner
- The remaining samples are for training

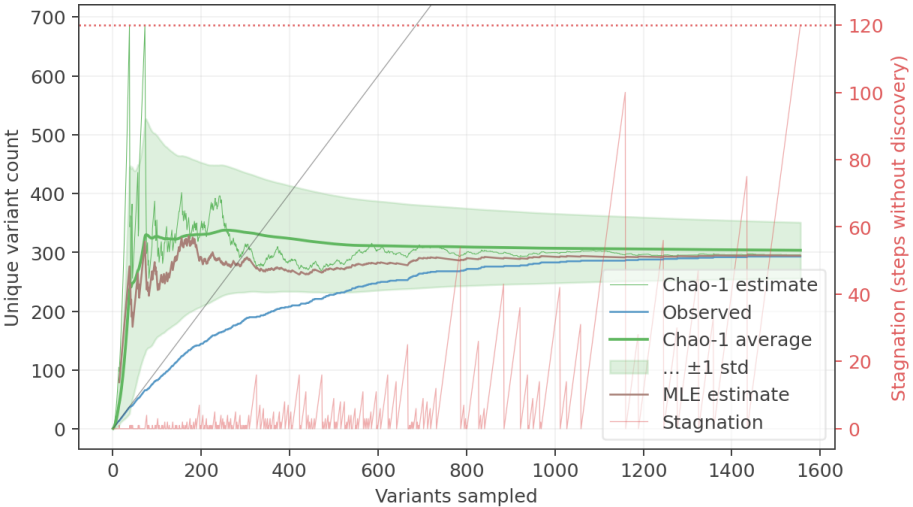


- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- ...plus the same number from the inner
- The remaining samples are for training



- Embed samples in euclidean space [9]
- Cluster with k-means
- Take the outer 10% of each cluster
- These become validation/test samples
- ...plus the same number from the inner
- The remaining samples are for training

Collecting random serializations



this slide is intentionally left blank