Molecular structure inference from molecular spectra

with transformers

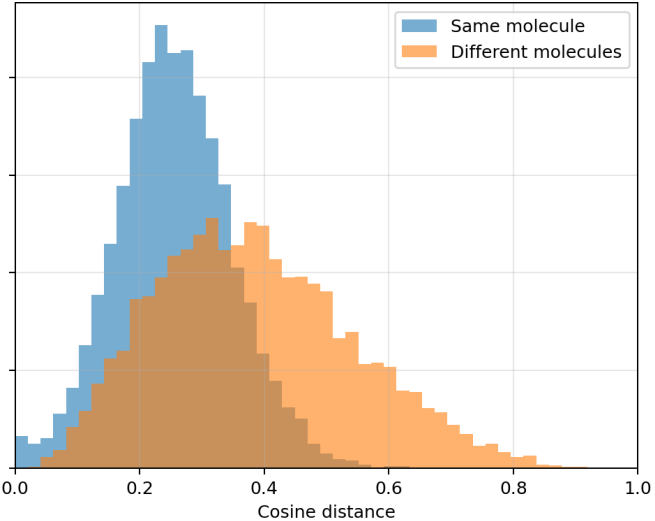
ChemBERTa

ChemBERTa 2020, Chithrananda et al. [2]

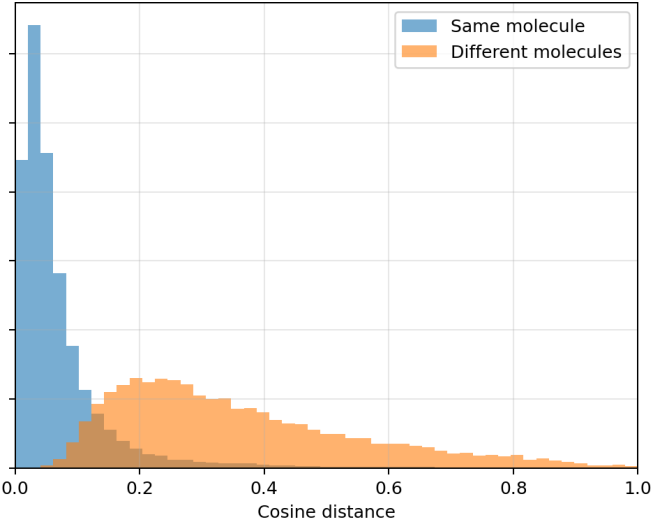
- Model: 6-layer / 12-head RoBERTa [1] variant from HuggingFace
- Pretraining: **masked language modeling** on SMILES (15% masked)
- Subsets of 100K–10M of 77M canonicalized SMILES from PubChem
- Tokenizers: compared BPE vs regex-based SMILES tokenizer
- Pretraining: max length 512, vocab size 52K, up to 10 epochs on V100
- Downstream tasks: MoleculeNet (BBBP, ClinTox, HIV, Tox21); scaffold splits; linear head finetuning
- HuggingFace model seyonec/ChemBERTa-zinc-base-v1 has a "**pooler**"

ChemBERTa-2 2022, Ahmad et al. [3]

- Model: RoBERTa-style [1], 5M–46M parameters
- Pretraining: masked language modeling and multi-task regression
- Data: 77M from PubChem; canonicalized & shuffled; validation: 100K
- Vocabulary: 591 tokens; max sequence length 512
- MTR labels: 200 RDKit-computed properties per compound
- Finetuning: MoleculeNet (8 tasks, both regression and classification); scaffold split; up to 100 epochs
- Results: MTR outperforms MLM; 6 of 8 tasks beat Chemprop D-MPNN
- "pooler" weights missing from model checkpoints



ChemBERTa 2020 [2], “ZINC base” 100k (?), [CLS]-pooling (?)



ChemBERTa-2 2022 [3], 77M-MTR, average pooling

Functional groups in “Alberts” vs “PubChem”

FG from Jung et al. [4] (used in Alberts et al. [5]/[6])

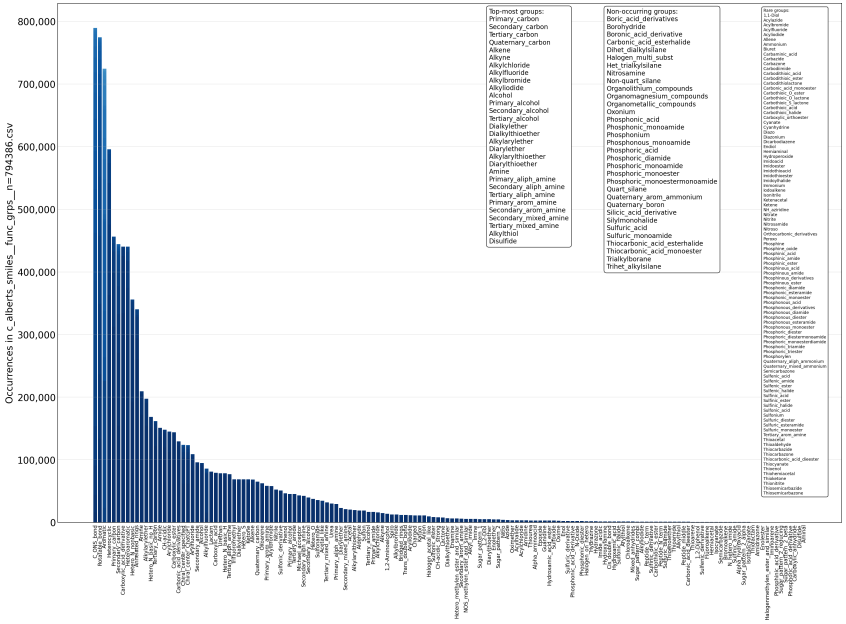
Group	SMARTS	Group	SMARTS
Acid anhydride	`[CX3](=[OX1])[OX2][CX3](=[OX1])`	Haloalkane	`[#6][F,Cl,Br,I]`
Acyl halide	`[CX3](=[OX1])[F,Cl,Br,I]`	Hydrazine	`[NX3][NX3]`
Alcohol	`[#6][OX2H]`	Hydrazone	`[NX3][NX2]=[#6]`
Aldehyde	`[CX3H1](=O)[#6,H]`	Isocyanate	`[NX2]=[C]=[O]`
Alkane	`[CX4;H3,H2]`	Isothiocyanate	`[NX2]=[C]=[S]`
Alkene	`[CX3]=[CX3]`	Ketone	`[#6][CX3](=O)[#6]`
Alkyne	`[CX2]#[CX2]`	Nitrile	`[NX1]#[CX2]`
Amide	`[NX3][CX3](=O)[#6]`	Phenol	`[OX2H][cX3]:[c]`
Amine	`[NX3;H2,H1,H0;!$(NC=O)]`	Phosphine	`[PX3]`
Arene	`[cX3]1[cX3][cX3][cX3][cX3][cX3]1`	Sulfonamide	`[#16X4]([NX3])(=[OX1])(=[OX1])[#6]`
Azo	`[#6][NX2]=[NX2][#6]`	Sulfonate	`[#16X4](=[OX1])(=[OX1])([#6])[OX2H0]`
Carbamate	`[NX3][CX3](=[OX1])[OX2H0]`	Sulfide	`[#16X2H0]`
Carboxylic acid	`[CX3](=O)[OX2H]`	Sulfone	`[#16X4](=[OX1])(=[OX1])([#6])[#6]`
Enamine	`[NX3][CX3]=[CX3]`	Sulfonic acid	`[#16X4](=[OX1])(=[OX1])([#6])[OX2H]`
Enol	`[OX2H][#6X3]=[#6]`	Sulfoxide	`[#16X3]=[OX1]`
Ether	`[OD2]([#6])[#6]`	Thial	`[CX3H1](=O)[#6,H]`
Ester	`[#6][CX3](=O)[OX2H0][#6]`	Thioamide	`[NX3][CX3]=[SX1]`
Imide	`[CX3](=O)[NX3][CX3](=O)`	Thiol	`[#16X2H]`
Imine	`$([CX3]([##6])[##6]),$([CX3H][##6])]=$([NX2][##6]),$([NX2H])`		

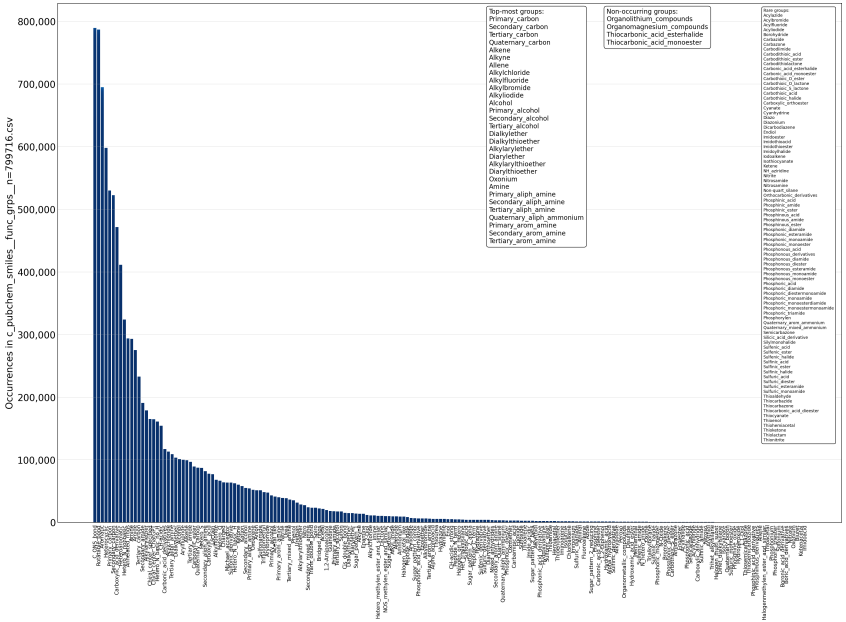
Group (Excluded)	SMARTS
Allene	`[CX3]=[CX2]=[CX3]`
Azide	`$(*-[NX2-]-[NX2+]##[NX1]),$(*-[NX2]=[NX2+]=[NX1-])`
Carbamic acid	`[NX3][CX3](=[OX1])[OX2H]`
Cyanate	`[OX2][CX2]#N`
Nitrate	`$([NX3](=[OX1])(=[OX1])O),$([NX3+]([OX1-])(=[OX1])O)`
Nitrite	`[OX2][NX2]=O`
Sulfinic acid	`[#16X3](=[OX1])[OX2H]`
Thiocyanate	`[SX2][CX2]#N`
Thioketone	`[#6][CX3](=S)[#6]`

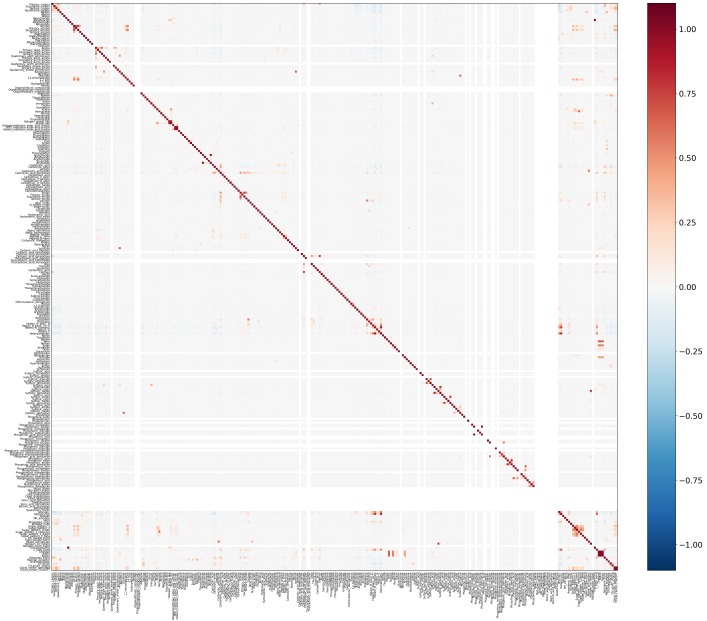
Functional groups: Open Babel SMARTS

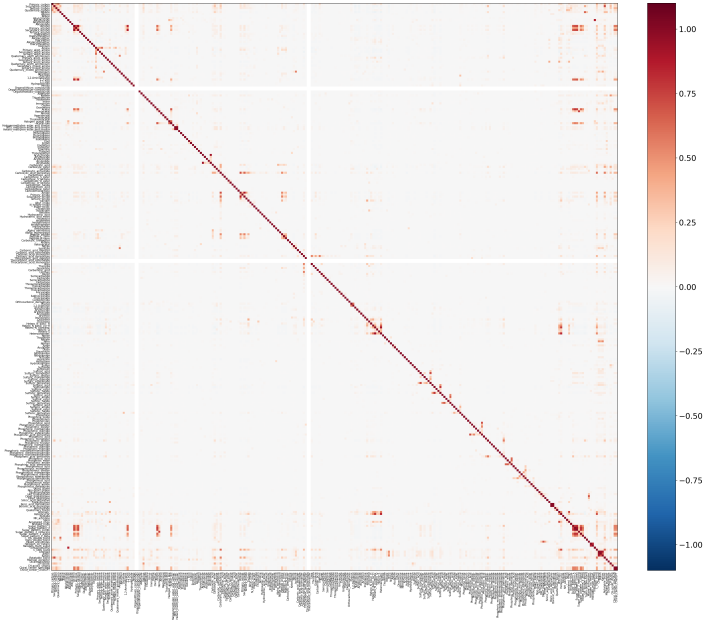
- Open Babel is an open-source chemical toolbox for interconverting chemical formats and analyzing molecular data
- Includes a curated set of SMARTS patterns for functional groups [url], authored by Christian Laggner, Inte:Ligand Software-Entwicklungs und Consulting GmbH (2005)
- Originally under the LGPL license[1]
- 288 SMARTS that `rdkit` could parse, and 19 not [url]
- Added another one for "chiral center" by ChatGPT

[1] Use, modification, and distribution of the code, including in proprietary software, as long as improvements to the LGPL-covered components remain open-source


Occurrences in c_alberts_smiles__func_grps__n=794386.csv
800,000
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
Top-most groups:
Primary_carbon
Secondary_carbon
Tertiary_carbon
Quaternary_carbon
Alkene
Alkyne
Alkylchloride
Alkylfluoride
Alkylbromide
Alkyliodide
Alcohol
Primary_alcohol
Secondary_alcohol
Tertiary_alcohol
Dialkylether
Dialkylthioether
Alkylarylether
Diarylether
Alkylarylthioether
Diarylthioether
Amine
Primary_aliph_amine
Secondary_aliph_amine
Tertiary_aliph_amine
Primary_arom_amine
Secondary_arom_amine
Secondary_mixed_amine
Tertiary_mixed_amine
Alkylthiol
Disulfide
Non-occurring groups:
Boric_acid_derivatives
Borohydride
Boronic_acid_derivative
Carbonic_acid_esterhalide
Dihet_dialkylsilane
Halogen_multi_subst
Het_trialkylsilane
Nitrosamine
Non-quart_silane
Organolithium_compounds
Organomagnesium_compounds
Organometallic_compounds
Oxonium
Phosphonic_acid
Phosphonic_monoamide
Phosphonium
Phosphonous_monoamide
Phosphoric_acid
Phosphoric_diamide
Phosphoric_monoamide
Phosphoric_monoester
Phosphoric_monoestermonoamide
Quart_silane
Quaternary_arom_ammonium
Quaternary_boron
Silicic_acid_derivative
Silylmonohalide
Sulfuric_acid
Sulfuric_monoamide
Thiocarbonic_acid_esterhalide
Thiocarbonic_acid_monoester
Trialkylborane
Trihet_alkylsilane
Rare groups
1,1-Diol
Acylazide
Acylhydrazide
Acylhydrazide
Acylsulfonamide
Ammonium
Biuret
Carbamimic_acid
Carbazide
Carbazone
Carbodiimide
Carbodithioic_acid
Carbodithioic_ester
Carbodithioic_thioester
Carbonic_acid_monoester
Carbonic_acid_diester
Carbothioic_O_lactone
Carbothioic_S_lactone
Carbothioic_acid
Carbothioic_halide
Carboxylic_orthoester
Cyanate
Cyanhydrine
Diazo
Diazonium
Dicarbodiazene
Endiol
Enhydrazinol
Enylhydroxylamide
Imidoacid
Imidoester
Imidothioacid
Imidothioester
Imidylthiolate
Imnoketam
Isocyanate
Isoimiline
Ketenacetal
Ketene
Mf_selrefine
Nitrate
Nitro
Nitrosamide
Nitroso
Nitrosocarbonic_derivatives
Peroxo
Phosphine
Phosphinic_amide
Phosphinic_acid
Phosphinic_amide
Phosphinic_ester
Phosphinous_acid
Phosphinous_amide
Phosphinous_derivatives
Phosphinous_ester
Phosphonic_diamide
Phosphonic_esteramide
Phosphonic_imiamide
Phosphonic_monoamide
Phosphonic_monoester
Phosphonic_diester
Phosphonous_derivatives
Phosphonous_diamide
Phosphonous_monoamide
Phosphonous_esteramide
Phosphonous_monoester
Phosphonous_diester
Phosphoric_diesteramonoamide
Phosphoric_monoesterdiamide
Phosphoric_triamide
Phosphoric_triester
Phosphorylyl
Quaternary_aliph_ammonium
Quaternary_mixed_ammonium
Semicarbazone
Sulfenic_acid
Sulfenic_amide
Sulfenic_ester
Sulfenic_halide
Sulfinic_acid
Sulfinic_amide
Sulfinic_halide
Sulfonic_acid
Sulfonamide
Sulfonic_diester
Sulfonic_esteramide
Sulfonic_monoester
Tertiary_arom_amine
Thioacetal
Thioacetal
Thioacetaldehyde
Thiocarbazone
Thiocarbonic_acid_diester
Thiocyanate
Thioenol
Thioenolate
Thioketene
Thioketone
Thioketoximate
Thiosemicarbazide


Occurrences in c_pubchem_smiles__func_grps__n=799716.csv
800,000
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
Top-most groups:
Primary_carbon
Secondary_carbon
Tertiary_carbon
Quaternary_carbon
Alkene
Alkyne
Allene
Alkylchloride
Alkylfluoride
Alkylbromide
Alkyloiodide
Alcohol
Primary_alcohol
Secondary_alcohol
Tertiary_alcohol
Dialkylether
Dialkylthioether
Alkylarylether
Diarylether
Alkylarylthioether
Diarylthioether
Oxonium
Amine
Primary_aliph_amine
Secondary_aliph_amine
Tertiary_aliph_amine
Quaternary_aliph_ammonium
Primary_arom_amine
Secondary_arom_amine
Tertiary_arom_amine
Non-occurring groups:
Organolithium_compounds
Organomagnesium_compounds
Thiocarbonic_acid_esterhalide
Thiocarbonic_acid_monoester
Rem groups
Acylazide
Acylbromide
Acylchloride
Acylfluoride
Benzylfluoride
Benzylhalide
Carbazone
Carbodiimide
Carbodithioic_acid
Carbodithioic_ester
Carbonic_acid_ester
Carbonic_acid_esterhalide
Carbonic_acid_monoester
Carbonic_acid_ester
Carbothioic_O_lactone
Carbothioic_S_lactone
Carbothioic_acid
Carbothioic_halide
Carbonylic_orthoester
Cyanate
Cyanhydrine
Diazo
Diazonium
Dicarpholsulfone
Endiol
Enolester
Imidethioacid
Imidoiodender
Imidobromide
Isobutylene
Isothiocyanate
Nitrene
N1_phrdine
Nitrate
Nitrosamide
Nitrosamine
Nitroso_N_oxide
Orthocarbolic_derivatives
Phosphinic_acid
Phosphinic_ester
Phosphinic_ester
Phosphinous_acid
Phosphinous_ester
Phosphinous_ester
Phosphonic_diamide
Phosphonic_monoamide
Phosphonic_monoamide
Phosphonic_monoester
Phosphonic_acid
Phosphonous_derivatives
Phosphonous_diamide
Phosphonous_diester
Phosphonous_monoamide
Phosphonous_monoamide
Phosphonous_monoester
Phosphonic_acid
Phosphonic_diester
Phosphonic_diesteramonoamide
Phosphonic_monoamide
Phosphonic_monoamide
Phosphonic_monoesteramonoamide
Phosphonic_monoesteramonoamide
Phosphonic_triamide
Phosphonium
Quaternary_arom_ammonium
Quaternary_mixed_ammonium
Selenocyanate
Silicic_acid_derivative
Silylmonohalide
Sulfenic_acid
Sulfenic_ester
Sulfenic_halide
Sulfinic_acid
Sulfinic_ester
Sulfinic_halide
Sulfonic_acid
Sulfonic_diester
Sulfonic_monoamide
Sulfonic_monoamide
Thioaldehyde
Thioketone
Thiocarbonyl
Thiocarbonic_acid_diester
Thiocyanate
Thiourea
Thiohemiacetal
Thiolactone
Thiolactam
Thiandrite


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

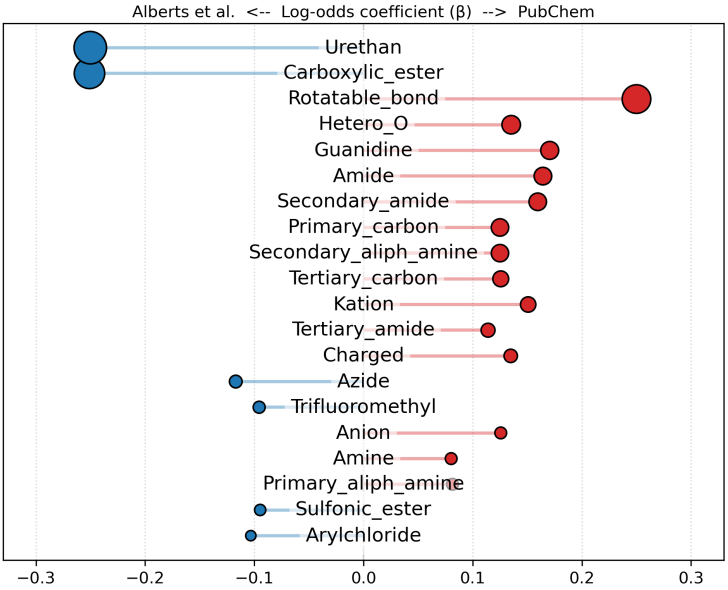

1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

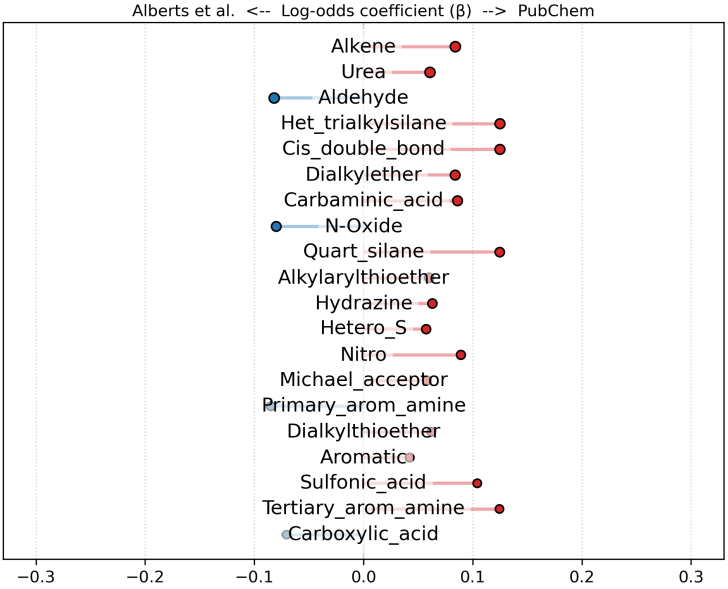
Elastic-net logistic regression: Alberts vs PubChem

- Merge FG counts from Alberts ($y = 0$) and PubChem ($y = 1$)
- Model: $\log \frac{P(y=1|x)}{P(y=0|x)} = \beta_0 + \sum_j \beta_j x_j$, with x_j scaled by their *s.d.*
- Minimize over β the objective $\alpha := 10\%$

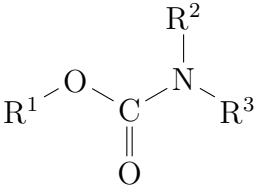
$$\frac{1}{n}\sum_{i=1}^{n}\left(-y_i \log \hat{p}_i - (1-y_i)\log(1-\hat{p}_i)\right) + \tfrac{1}{C}\left(\alpha\|\beta\|_1 + \tfrac{1}{2}(1-\alpha)\|\beta\|_2^2\right)$$

  where $\hat{p}_i := \frac{1}{1+\exp(-x^{(i)}\cdot\beta)}$ for sample $\#i$ out of $n \approx 2 \times 800\text{k}$ samples
- Grid-search C to max ROC-AUC (3xCV mean); use growing subsets
- $\exp \beta_j$ scales PubChem:Alberts odds per 1 *s.d.* increase in FG j
- AUC = chance model scores random PubChem > random Alberts
- Permutation importance = AUC drop when a FG is shuffled ($\lesssim 2\%$)


Alberts et al. <-- Log-odds coefficient (β) --> PubChem
Urethan
Carboxylic_ester
Rotatable_bond
Hetero_O
Guanidine
Amide
Secondary_amide
Primary_carbon
Secondary_aliph_amine
Tertiary_carbon
Kation
Tertiary_amide
Charged
Azide
Trifluoromethyl
Anion
Amine
Primary_aliph_amine
Sulfonic_ester
Arylchloride
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3


Alberts et al. <-- Log-odds coefficient (β) --> PubChem
Alkene
Urea
Aldehyde
Het_trialkylsilane
Cis_double_bond
Dialkylether
Carbaminic_acid
N-Oxide
Quart_silane
Alkylarylthioether
Hydrazine
Hetero_S
Nitro
Michael_acceptor
Primary_arom_amine
Dialkylthioether
Aromatic
Sulfonic_acid
Tertiary_arom_amine
Carboxylic_acid
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3

Functional group: urethane / carbamate



- Combines an ester and an amide in one motif. Found in:
- protecting groups (e.g., Boc, Cbz[2]), polyurethanes, drugs/pesticides
- SMARTS: `[#7X3][#6](=[OX1])[#8X2][#6]`
  - `[#7X3]` = trivalent nitrogen (NR)
  - `[#6](=[OX1])` = carbonyl carbon (C=O)
  - `[#8X2][#6]` = ether oxygen and its attached carbon

[2] Boc = tert-butoxycarbonyl R−NH−COO−tBu / Cbz = benzyloxycarbonyl R−NH−COO−CH_2Ph

Functional group: carboxylic ester

$$R^1-C(=O)-O-R^2$$

- Formed by condensation of a carboxylic acid and an alcohol. Found in:
  - fragrances and flavors (e.g., ethyl acetate, isoamyl acetate)
  - fats and oils (triglycerides)
  - prodrugs and polymers (e.g., PLA, PMMA)
- `[CX3;$([R0][#6]),$([H1R0])](=[OX1])[OX2][#6;!$(C=[O,N,S])]`
  - `[CX3]` = sp^2 carbon
  - `$([R0][#6])` = non-ring carbon bonded to carbon
  - `$([H1R0])` = or hydrogen
  - `=[OX1]` = carbonyl oxygen
  - `[OX2][#6]` = ether oxygen and attached carbon
  - `[#6;!$(C=[O,N,S])]` = carbon that's not =O, =N, or =S

$$\text{H} + \text{C} + \text{HSQC NMR} \rightarrow \text{structure}$$

based on the ∼800'000 molecule synthetic dataset

[5] Marvin Alberts et al. "Unraveling Molecular Structure: A Multimodal Spectroscopic Dataset for Chemistry". In: *NeurIPS 2024 Datasets and Benchmarks Track*. 2024. [url]

in turn based on

[7] Daniel Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017. [url]

and MestReNova software (NMR prediction plug-in).

GPT-2 series

GPT-2 is a text-to-text decoder-only autoregressive transformer [8].

Original training:

- Unsupervised next-token prediction on the WebText corpus
- Causal language modeling: $\mathbb{P}(x_t \mid x_1, \ldots, x_{t-1})$

Architecture:

- Byte-Level BPE tokenizer (iteratively merge frequent pairs, 50k)
- Stack of masked self-attention blocks (12–48 layers)
- Learned token and positional embeddings

Key models:

- GPT-2: Up to 1.5B parameters (XL), strong zero-shot performance
- **DistilGPT-2:** "distilled" to 82M (cf. [9]), 1024 token context

GPT-2 reference implementation[3]

Input: $x \in \mathbb{N}^{B \times T}$ (token indices)

Token embedding: $\mathbf{E}_{\text{tok}} = \text{Embedding}(x) \in \mathbb{R}^{B \times T \times d}$

Position embedding: $\mathbf{E}_{\text{pos}} = \text{Embedding}[1 : T]$

Combined: $\mathbf{H}^{(0)} = \text{Dropout}(\mathbf{E}_{\text{tok}} + \mathbf{E}_{\text{pos}})$

For each block $\ell = 1 \ldots L$:

$\mathbf{X} = \mathbf{H}^{(\ell-1)}$

$\mathbf{X} \leftarrow \mathbf{X} + \text{CausalSelfAttn}(\text{LayerNorm}(\mathbf{X}))$

$\mathbf{X} \leftarrow \mathbf{X} + \text{MLP}(\text{LayerNorm}(\mathbf{X}))$

$\mathbf{H}^{(\ell)} = \mathbf{X}$

Final: $\mathbf{H}_{\text{final}} = \text{LayerNorm}(\mathbf{H}^{(L)})$

Logits: $\hat{y} = \mathbf{H}_{\text{final}} \cdot \mathbf{E}_{\text{tok}}^{\top}$ (weight tying)

[3] Based on https://github.com/karpathy/nanoGPT/blob/master/model.py

Roadmap

- More careful data split (correlations due to USPTO origin)
- Pre-train on SMILES, for example:
  - bias a hidden state towards ChemBERTa,
  - hijack a hidden state to predict functional groups,
  - augment with non-canonical SMILES.
- Correctness self-assessment flag – after SMILES add ✓/✗
- Backtracking / inference-time reasoning – provide/teach *undo*
- Train on the one-atom-one-peak quasi-NMR

SMILES augementation

SMILES augmentation I

- Given SMILES, predict a variant: learn the next-token distribution
- Goal: teach molecular structure, not the syntax / canonicalization
- Issue: Number of alternative SMILES is $\sim$exponential in length
- Expectation: starts with easy parts and isn't confidently wrong

SMILES augmentation II

Examples:

- "SMILES enumeration as data augmentation for neural network modeling of molecules", Bjerrum, 2017 [10]
- "Randomized SMILES strings improve the quality of molecular generative models", Arús-Pous et al., 2019 [11]
- "Pushing the boundaries of molecular property prediction for drug discovery with multitask learning BERT enhanced by SMILES enumeration", Zhang et al., 2022 [12] – much like ChemBERTa-v2 [3]
- "Invalid SMILES are beneficial rather than detrimental to chemical language models", Skinnider, 2024 [13]

SMILES augmentation: Bjerrum, 2017 [10]

- Predict "Quantitative Structure–Activity Relationship"
- Dataset: 756 DHFR[4] inhibitors with $\log IC_{50}$ values (from [14])
- Method:
  - ∼130 randomized SMILES per molecule via RDKit
  - LSTM models trained on canonical or enumerated SMILES
- Results:
  - R^2: 0.56 (canonical) → 0.66 (enumerated)
  - RMS error: 0.62 → 0.52 (with SMILES averaging)
  - "Canonical model" failed on alternate SMILES

[4] dihydrofolate reductase

SMILES augmentation: Arús-Pous et al., 2019 [11]

- Improve RNN-based molecular generation using SMILES augmentation
- Data: GDB-13 (1M, 10K, 1K) and ChEMBL (1.48M); 300 epochs
- Method:
  - One randomized SMILES per molecule per epoch (pre-generated)
  - Model learns to predict the next token in that string
  - Loss applied to token-level predictions, not to variant distributions
- Results:
  - GDB-13 coverage: 83% (randomized) vs 73% (canonical)
  - Stronger gains in low-data settings (e.g., 1K $\rightarrow$ 34% vs 14%)
  - On ChEMBL: 64% unique valid SMILES vs 35%

SMILES augmentation: Skinnider, 2024 [13]

- Evaluate the role of invalid SMILES in chemical language models
- Models: LSTM (3-layer, 1024 hidden), GPT (8-layer, 256-dim, 8 heads)
- Data: ChEMBL and GDB-13, with 10×–30× SMILES enumeration
- Findings:
  - Invalid SMILES correlate with low-likelihood samples
  - SMILES generalize better than SELFIES on GDB-13 coverage
  - SELFIES tend to overproduce aliphatic, non-aromatic structures
  - Removing SELFIES valency constraints improves output
- Invalid SMILES are not harmful noise but useful signals of poor generation; filtering them is simple and effective

Loss on functional group counts

Hijacking hidden states

O=C([O-])[O-].[Ca+2][:][_][○][_][EOS]

A given SMILES structure

Hijacking hidden states

O=C([O-])[O-].[Ca+2][:][_][∘][_][EOS]

Unsupervised trigger token (molecular embedding)

Hijacking hidden states

O=C([O-])[O-].[Ca+2][:][_][∘][_][EOS]

Supervised hidden state (molecular embedding)

Hijacking hidden states

O=C([O-])[O-].[Ca+2][:][_][∘][_][EOS]

Unsupervised trigger token (functional groups)

Hijacking hidden states

O=C([O-])[O-].[Ca+2][:][_][∘][_][EOS]

Supervised hidden state (functional groups)

Hijacking hidden states

O=C([O-])[O-].[Ca+2][:][_][∘][_][EOS]

End of sequence (supervised)

Whitened-ℓ_1 loss for correlated targets

Let $\mathbf{y}$ denote the functional group counts and $\hat{\mathbf{y}}$ our estimate thereof, where $\mathbf{y}_i$ is the count of functional group i in a given molecule. Issues:

- correlations between counts;
- large variation in counts (rare vs abundant); presence more important;
- ideally should weight the groups by usefulness to the main task.

Proposal: Set $\mathbf{z} := \log(1 + \mathbf{y})$, and write $\mathbf{e} := \mathbf{z} - \hat{\mathbf{z}}$. Learn a global matrix T with $\det T > 0$, along with the $\hat{\mathbf{z}}$, to

$$\text{minimize} \quad \|T\mathbf{e}\|_1 - \log\det T.$$

Motivation:

- $\|T\mathbf{e}\|_1$ promotes sparsity, hence decorrelates ("whitens") the errors;
- prevent collapse by regularizing T with $-\log\det T$;
- negative log-likelihood of $|\det T|\exp(-\|T\mathbf{e}\|_1)$, cf. Kendall et al. [15].

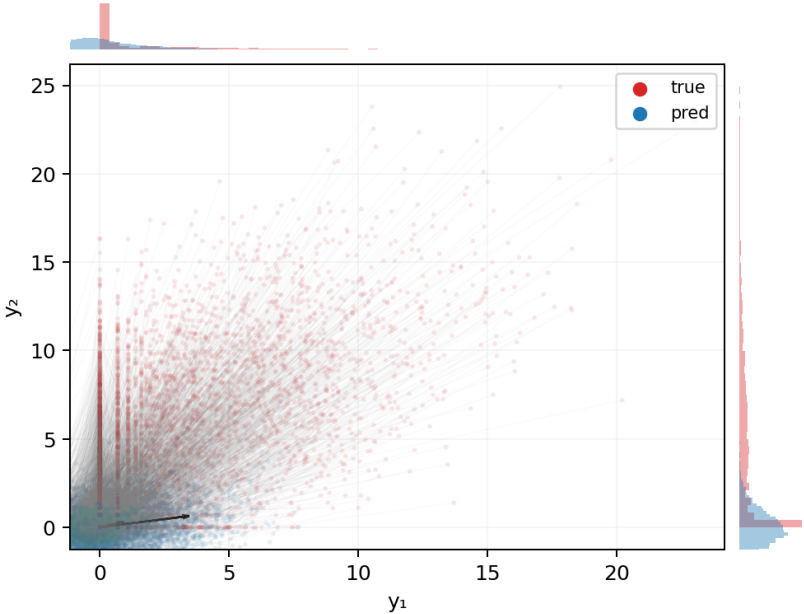
Negative log-likelihood I

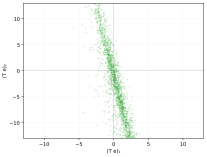
Let T be a $n \times n$ matrix with $\det T \neq 0$. Then

$$p(\mathbf{e} \mid T) = (2\pi)^{-n/2} (\det T) \exp(-\tfrac{1}{2} \| T\mathbf{e} \|_2^2)$$

is a probability density function on $\mathbb{R}^n$. It's the multivariate normal with mean *zero* and covariance $(T^\top T)^{-1}$. We might want to minimize w.r.t. T the negative log-likelihood:

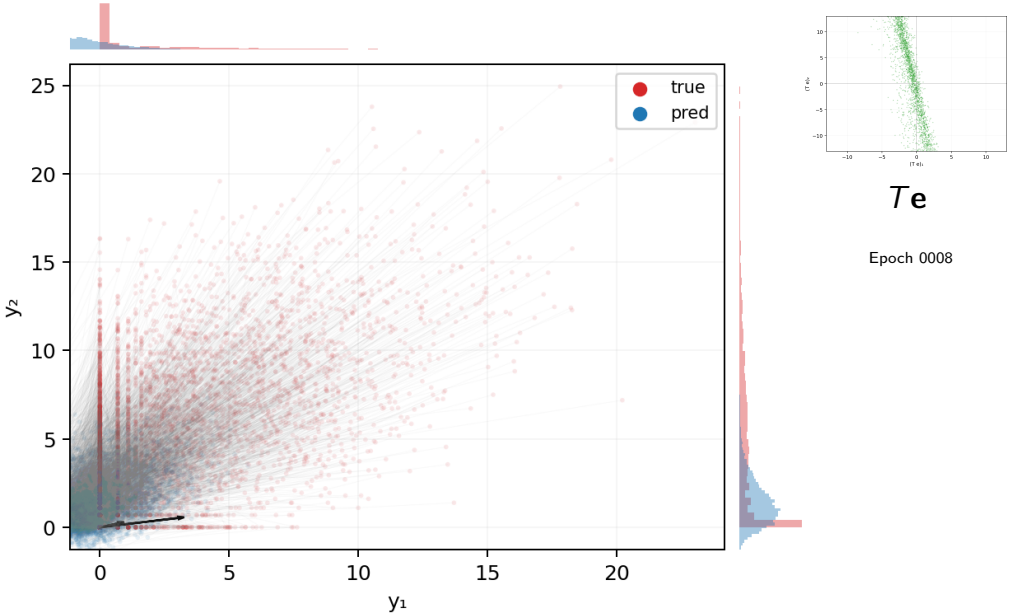
$$-\log p(\mathbf{e} \mid T) = \tfrac{1}{2} \| T\mathbf{e} \|_2^2 - \log \det T + \tfrac{n}{2} \log 2\pi.$$

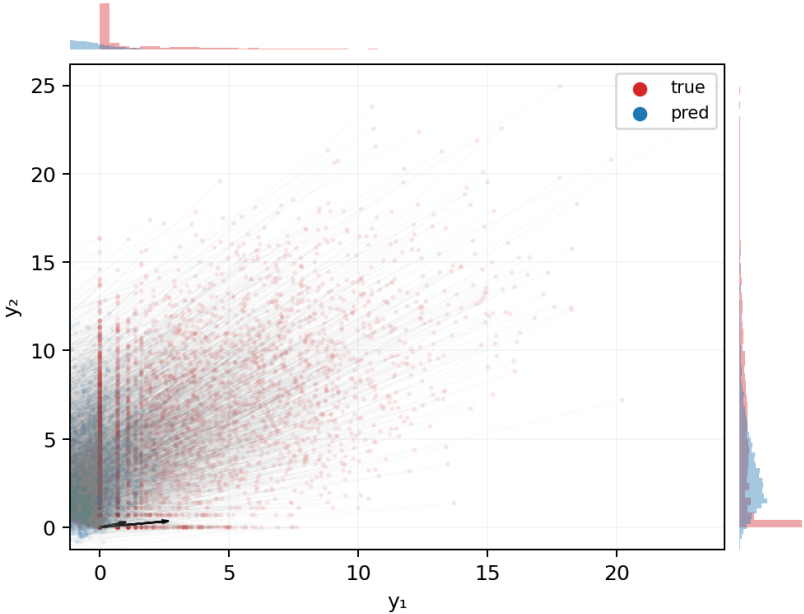


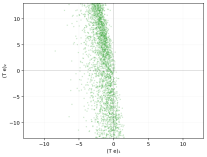


$T\mathbf{e}$

Epoch 0002

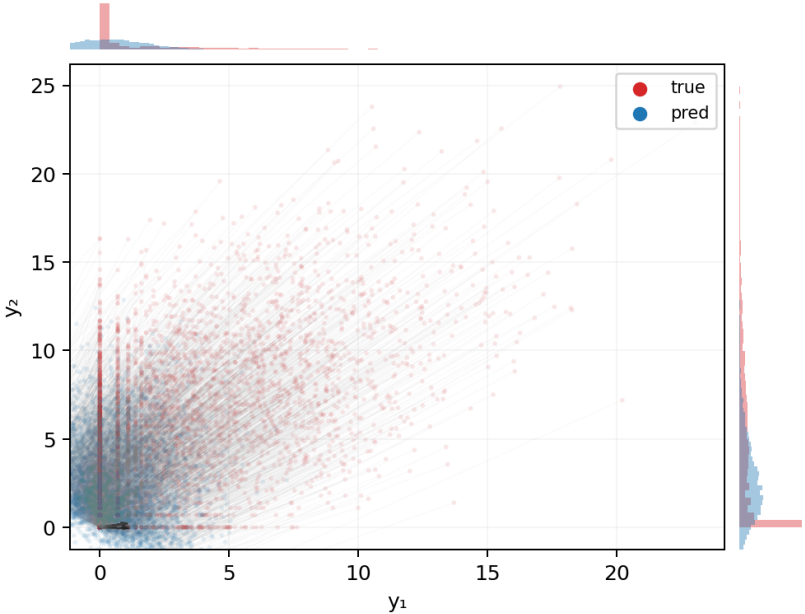

true
pred
y₂
y₁
0
5
10
15
20
25
$T\mathbf{e}$
Epoch 0008

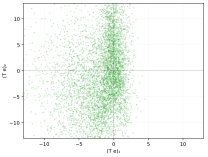




$T\mathbf{e}$

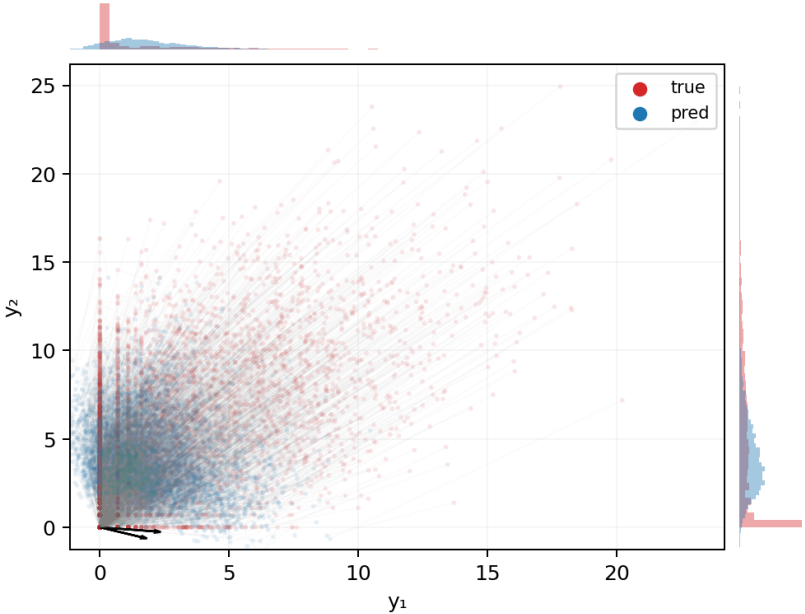
Epoch 0032

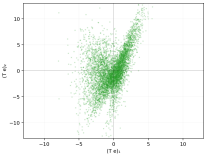




$T\mathbf{e}$

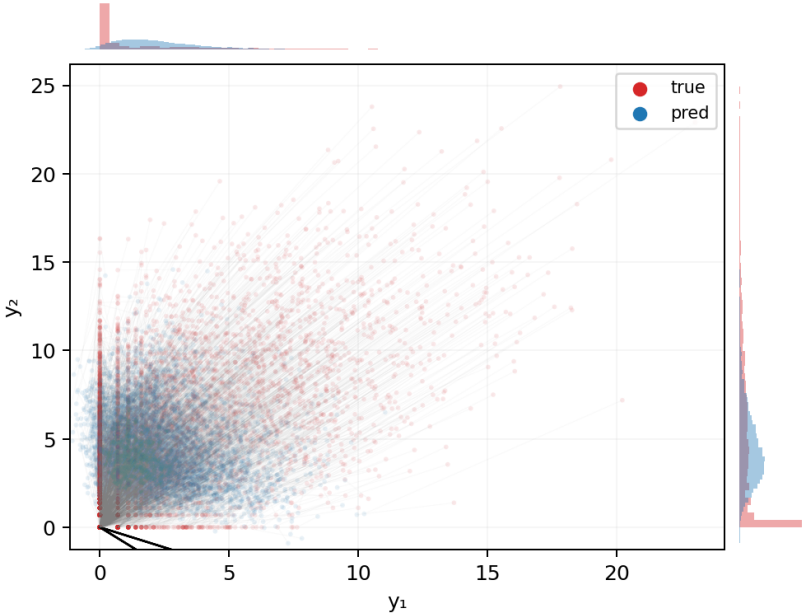
Epoch 0128

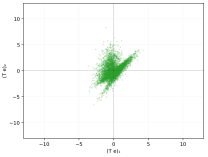




$T\mathbf{e}$

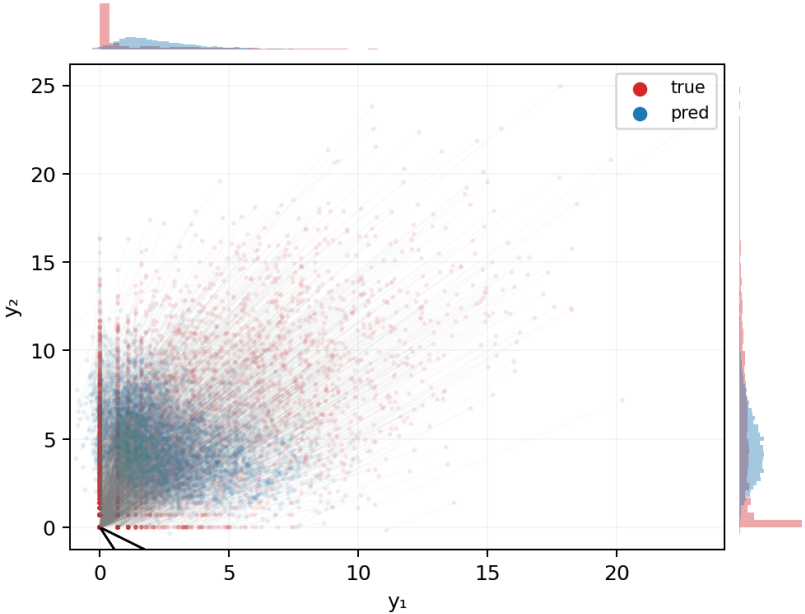
Epoch 0512

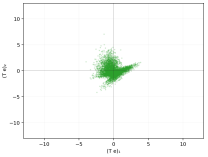




$T\mathbf{e}$

Epoch 2048





$T\mathbf{e}$

Epoch 8192

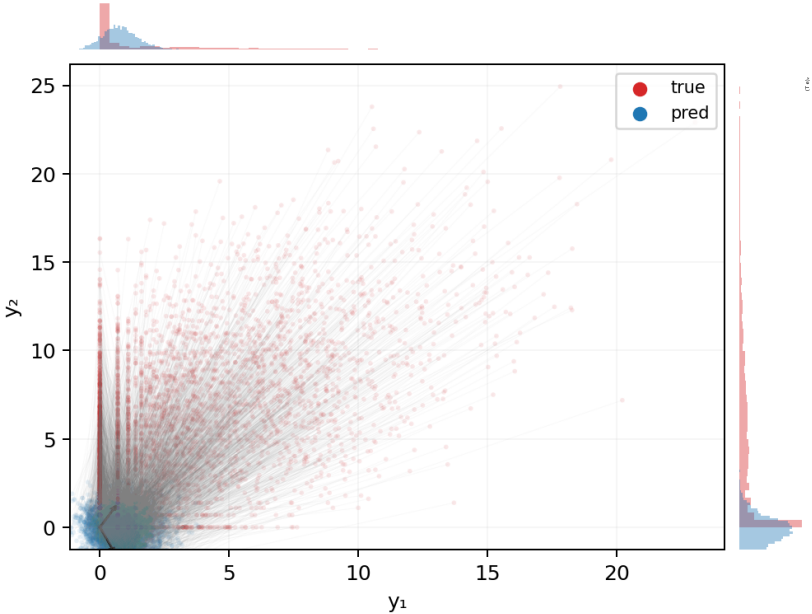
Negative log-likelihood II

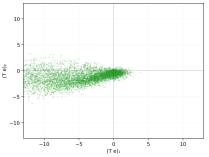
Let T be a $n \times n$ matrix with $\det T > 0$. Then

$$p(\mathbf{e} \mid T) = 2^{-n}(\det T)\exp(-\|T\mathbf{e}\|_1)$$

is a probability density function on $\mathbb{R}^n$. We might want to minimize w.r.t. T the negative log-likelihood:

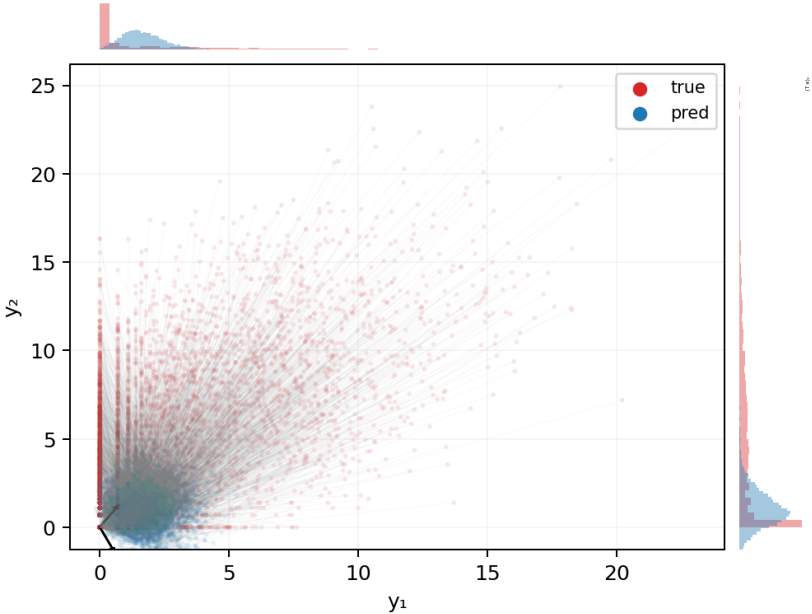
$$-\log p(\mathbf{e} \mid T) = \|T\mathbf{e}\|_1 - \log\det T + n\log 2.$$

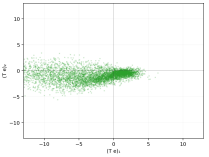




$T\mathbf{e}$

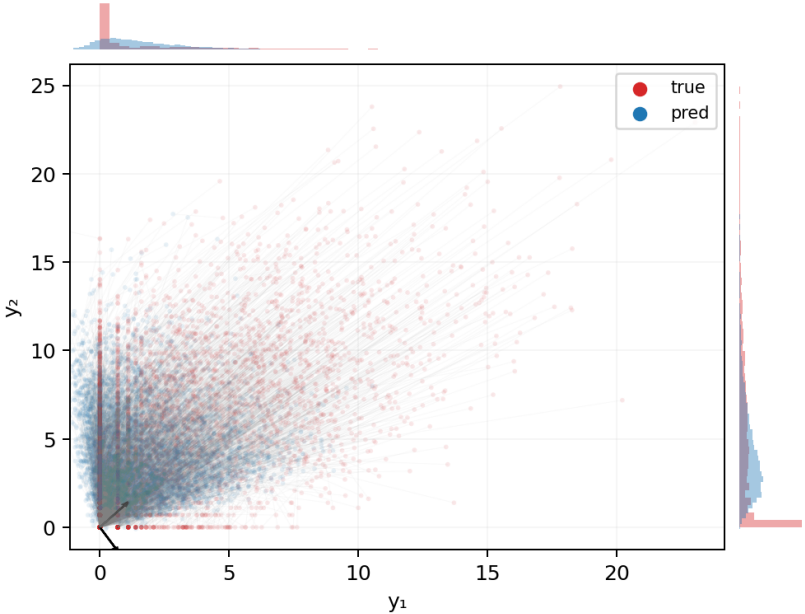
Epoch 0002

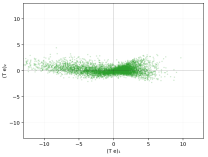




$T\mathbf{e}$

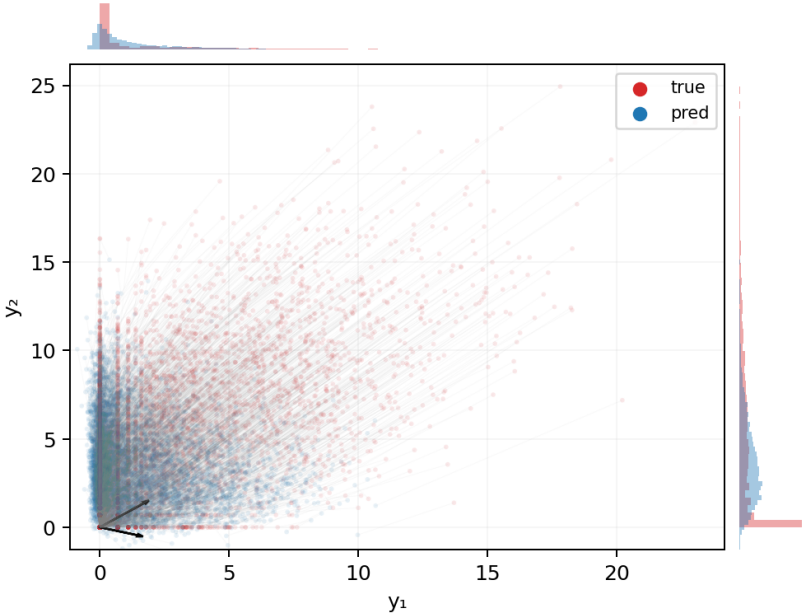
Epoch 0008

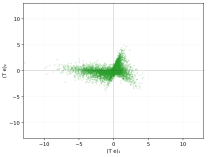




$T\mathbf{e}$

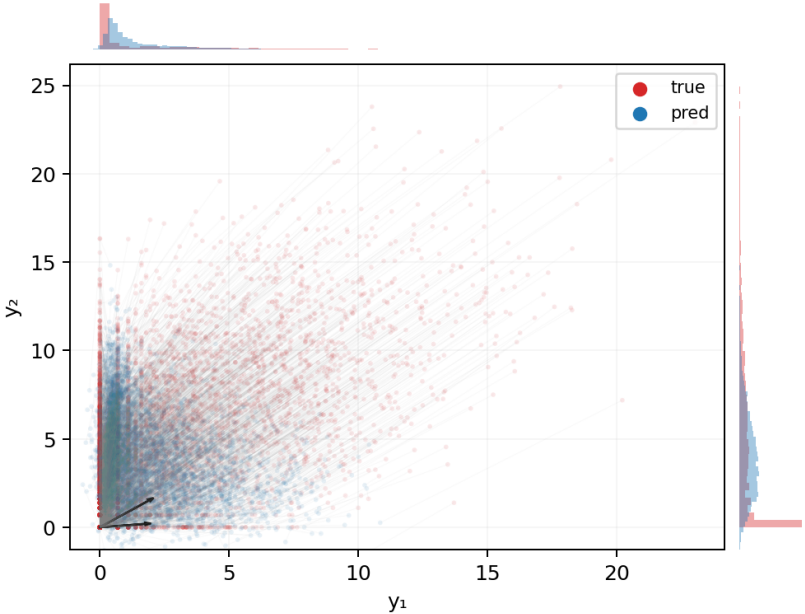
Epoch 0032

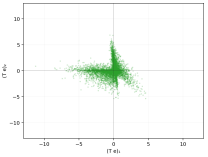




$T\mathbf{e}$

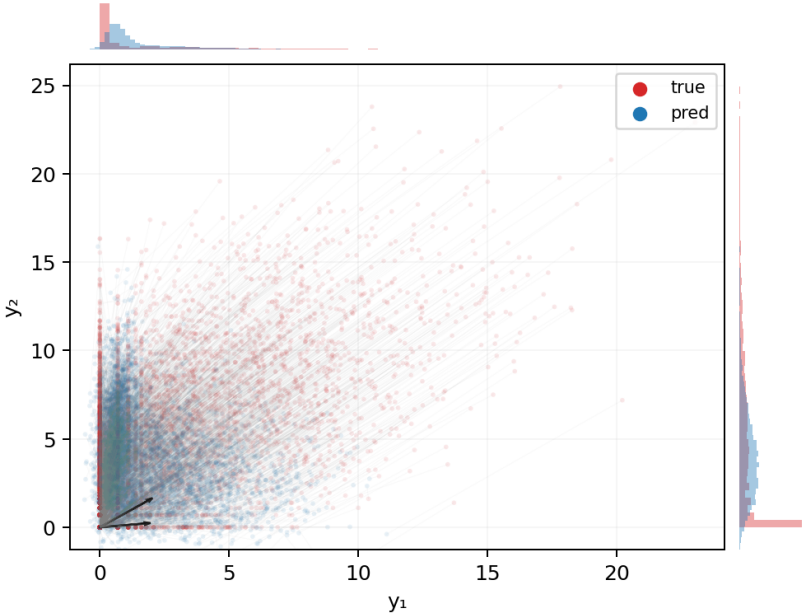
Epoch 0128

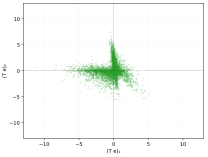




$T\mathbf{e}$

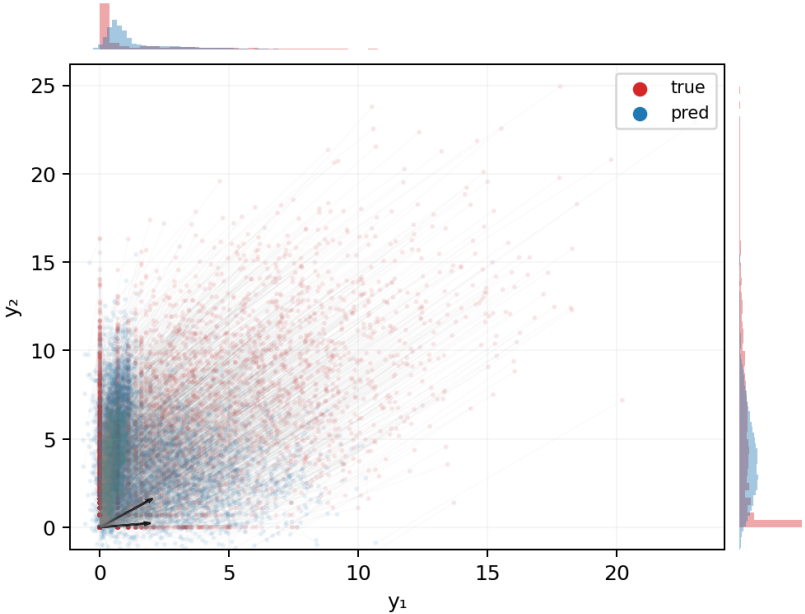
Epoch 0512

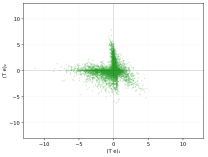




$T\mathbf{e}$

Epoch 2048





$T\mathbf{e}$

Epoch 8192

thx.

References I

[1] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. 2019. [url].

[2] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. "ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction". In: *Machine Learning for Molecules Workshop at NeurIPS 2020*. 2020. [url].

[3] Walid Ahmad, Elana Simon, Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. "ChemBERTa-2: Towards Chemical Foundation Models". In: *ELLIS Machine Learning for Molecule Discovery Workshop 2021*. 2022. [url].

References II

[4] Guwon Jung, Son Gyo Jung, and Jacqueline M. Cole. "Automatic materials characterization from infrared spectra using convolutional neural networks". In: *Chemical Science* 14.13 (2023), pp. 3600–3609. [url].

[5] Marvin Alberts, Oliver Schilter, Federico Zipoli, Nina Hartrampf, and Teodoro Laino. "Unraveling Molecular Structure: A Multimodal Spectroscopic Dataset for Chemistry". In: *NeurIPS 2024 Datasets and Benchmarks Track*. 2024. [url].

[6] Marvin Alberts, Teodoro Laino, and Alain C. Vaucher. "Leveraging Infrared Spectroscopy for Automated Structure Elucidation". In: *Communications Chemistry* 7.268 (2024). [url].

[7] Daniel Lowe. *Chemical reactions from US patents (1976–Sep2016)*. figshare, 2017. [url].

References III

[8] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. *Language models are unsupervised multitask learners*. Tech. rep. 2019.

[9] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter". In: *Proceedings of EMC^2: Efficient Machine Learning, Co-located with NeurIPS 2019* (2019).

[10] Esben Jannik Bjerrum. "SMILES enumeration as data augmentation for neural network modeling of molecules". In: (2017).

References IV

[11] Josep Arús-Pous, Simon Viet Johansson, Oleksii Prykhodko, Esben Jannik Bjerrum, Christian Tyrchan, Jean-Louis Reymond, Hongming Chen, and Ola Engkvist. "Randomized SMILES strings improve the quality of molecular generative models". In: *Journal of Cheminformatics* (2019), p. 71. [url].

[12] Xiao-Chen Zhang, Cheng-Kun Wu, Jia-Cai Yi, Xiang-Xiang Zeng, Can-Qun Yang, Ai-Ping Lu, Ting-Jun Hou, and Dong-Sheng Cao. "Pushing the boundaries of molecular property prediction for drug discovery with multitask learning BERT enhanced by SMILES enumeration". In: *Research* 2022 (2022), Article ID: 0004. [url].

[13] Michael A. Skinnider. "Invalid SMILES are beneficial rather than detrimental to chemical language models". In: *Nature Machine Intelligence* 6 (2024), pp. 437–448. [url].

References V

[14] Jeffrey J. Sutherland, Lee A. O'Brien, and Donald F. Weaver. "Spline-fitting with a genetic algorithm: A method for developing classification structure–activity relationships". In: *Journal of Chemical Information and Computer Sciences* 43.6 (2003), pp. 1906–1915. [url].

[15] Alex Kendall, Yarin Gal, and Roberto Cipolla. "Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2018). [url].

[16] Garrett B. Goh, Nathan Oken Hodas, Charles Siegel, and Abhinav Vishnu. "SMILES2Vec: An interpretable general-purpose deep neural network for predicting chemical properties". In: *ICLR 2018*. 2017. [url].

References VI

[17] Sheng Wang, Yuzhi Guo, Yuhong Wang, Hongmao Sun, and Junzhou Huang. "SMILES-BERT: Large scale unsupervised pre-training for molecular property prediction". In: *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '19)* (2019), pp. 429–436. [url].

[18] Gabriel A. Pinheiro, Juarez L. F. Da Silva, and Marcos G. Quiles. "SMICLR: Contrastive learning on multiple molecular representations for semisupervised and unsupervised representation learning". In: *Journal of Chemical Information and Modeling* 62.17 (2022), pp. 3948–3960. [url].

References VII

[19] Zhihui Guo, Pramod Sharma, Andy Martinez, Liang Du, and Robin Abraham. "Multilingual molecular representation learning via contrastive pre-training". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2022, pp. 3441–3453. [url].

[20] Yuyang Wang, Jianren Wang, Zhonglin Cao, and Amir Barati Farimani. "Molecular contrastive learning of representations via graph neural networks". In: *Nature Machine Intelligence* 4 (2022), pp. 279–287. [url].

[21] Jun Xia, Chengshuai Zhao, Bozhen Hu, Zhangyang Gao, Cheng Tan, Yue Liu, Siyuan Li, and Stan Z. Li. "Mole-BERT: Rethinking pre-training graph neural networks for molecules". In: *ICLR 2023*. 2023. [url].

References VIII

[22] Kathryn E. Kirchoff, Travis Maxfield, Alexander Tropsha, and Shawn M. Gomez. "SALSA: Semantically-aware latent space autoencoder". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 38.12 (2024), pp. 13211–13219. [url].

[23] Jun-Hyung Park, Hyuntae Park, Yeachan Kim, Woosang Lim, and SangKeun Lee. "Moleco: Molecular contrastive learning with chemical language models for molecular property prediction". In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*. Association for Computational Linguistics, 2024, pp. 408–420. [url].

[24] Ying Qian, Minghua Shi, and Qian Zhang. "CONSMI: Contrastive learning of SMILES representations for improved molecular generation". In: *Molecules* 29.2 (2024), p. 495. [url].

References IX

[25] Marcelo G. Quiles, Pedro A. L. Ribeiro, Gabriel A. Pinheiro, Ronaldo C. Prati, and João L. F. da Silva. "Enhancing low-cost molecular property prediction with contrastive learning on SMILES representations". In: *Computational Science and Its Applications – ICCSA 2024 Workshops. Lecture Notes in Computer Science, vol 14823*. Springer, Cham, 2024. [url].

[26] Ao Shen, Mingzhi Yuan, Yingfan Ma, Jie Du, and Manning Wang. "Complementary multi-modality molecular self-supervised learning via non-overlapping masking for property prediction". In: *Briefings in Bioinformatics* 25.4 (2024), bbae256. [url].

References X

[27] Jianwen Chen, Shuangjia Zheng, Ying Song, Jiahua Rao, and Yuedong Yang. "Learning attributed graph representations with communicative message passing transformer". In: *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21)* (2021), pp. 2242–2248. [url].

[28] Shikun Feng, Lixin Yang, Yanwen Huang, Yuyan Ni, Weiying Ma, and Yanyan Lan. "UniMAP: Universal SMILES-Graph representation learning". In: *arXiv preprint arXiv:2310.14216* (2023).

Appendix

Self-supervised and contrastive models for molecules

a somewhat random selection

SMILES2Vec by Goh et al., 2017 [16] is a character-level CNN-GRU model selected via Bayesian optimization and trained on one-hot encoded SMILES strings for molecular property prediction. Evaluated on Tox21 (8,014), HIV (41,193), FreeSolv (643), and ESOL (1,128), it achieves AUCs of 0.81 (Tox21), 0.80 (HIV), and RMSEs of 1.2 (FreeSolv) and 0.63 (ESOL), with a saliency-based interpretability mask identifying predictive atoms with 88% top-3 accuracy.

[16] Garrett B. Goh et al. "SMILES2Vec: An interpretable general-purpose deep neural network for predicting chemical properties". In: *ICLR 2018*. 2017. [url]

SMILES-BERT by Wang et al., 2019 [17] is a semi-supervised Transformer with six encoder layers, pre-trained on approximately 18.7 million SMILES strings via a masked token recovery task masking 15% of characters. A prepended <GO> token provides a sequence-level embedding for downstream prediction, yielding accuracies of 91.5% (LogP), 75.9% (PM2), and 87.8% (PCBA), outperforming baselines such as CircularFP and NeuralFP.

[17] Sheng Wang et al. "SMILES-BERT: Large scale unsupervised pre-training for molecular property prediction". In: *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics (BCB '19)* (2019), pp. 429–436. [url]

SMICLR by Pinheiro et al., 2022 [18] aligns representations from an LSTM over SMILES and a GCN over graphs, using modality-specific augmentations (e.g., node dropping, attribute masking, SMILES noise) and NT-Xent loss. Trained on QM9 with 5k labeled examples, it achieves up to 50% MAE reduction over supervised baselines and 89.98% accuracy on MUTAG, using only cross-modal positive pairs and showing robustness to low-cost geometries.

[18] Gabriel A. Pinheiro et al. "SMICLR: Contrastive learning on multiple molecular representations for semisupervised and unsupervised representation learning". In: *Journal of Chemical Information and Modeling* 62.17 (2022), pp. 3948–3960. [url]

Guo et al., 2022 [19] present MM-Deacon, a molecular representation model trained to align SMILES and IUPAC embeddings using bidirectional InfoNCE loss over 10M PubChem pairs. Inputs are tokenized (SMILES via BPE, IUPAC via regex) and encoded with separate 6-layer BERT-style Transformers (768 hidden units, 12 heads), then pooled and projected to a shared 512-d space. The model achieves strong performance on MoleculeNet and enables SMILES-to-IUPAC retrieval with over 85% top-5 accuracy from a 100K pool.

[19] Zhihui Guo et al. "Multilingual molecular representation learning via contrastive pre-training". In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2022, pp. 3441–3453. [url]

Wang et al., 2022 [20] propose MolCLR, a self-supervised framework that pretrains GCN and GIN encoders on $\sim$10M PubChem molecules using contrastive learning. Graphs are augmented via atom masking, bond deletion, and subgraph removal, and encoded into a latent space via a projection head; NT-Xent loss aligns views of the same molecule. After pretraining, an MLP is fine-tuned on MoleculeNet tasks (e.g., BBBP, ClinTox, QM9, ESOL), yielding improved performance on 5/7 classification and 5/6 regression datasets, especially low-resource ones. GIN benefits more than GCN (e.g., +50.3% MAE gain on QM9), attributed to greater capacity. Ablation shows subgraph removal (25%) is most effective, while combining augmentations can degrade performance. t-SNE and fingerprint analysis confirm that MolCLR learns chemically meaningful structure.

---

[20] Yuyang Wang et al. "Molecular contrastive learning of representations via graph neural networks". In: *Nature Machine Intelligence* 4 (2022), pp. 279–287. [url]

Xia et al., 2023 [21] first train a grouped VQ-VAE tokenizer on supervised atom embeddings to assign chemically contextualized discrete codes, grouped by element type. These codes supervise Masked Atoms Modeling, where a new GNN is trained to recover masked tokens from molecular context. Then, Triplet Masked Contrastive Learning further refines graph-level embeddings by combining triplet and contrastive losses over differently masked views of the same molecule. The model is pretrained on ZINC15, then fine-tuned and evaluated on MoleculeNet, achieving 1–2% ROC-AUC improvements over prior GNN pretraining methods.

[21] Jun Xia et al. "Mole-BERT: Rethinking pre-training graph neural networks for molecules". In: *ICLR 2023*. 2023. [url]

SALSA by Kirchoff et al., 2024 [22] is a SMILES-based transformer autoencoder with causal reconstruction and supervised contrastive loss (on pooled, L2-normalized embeddings). The model is trained on 1.26M ChEMBL anchors, each paired with 10 one-edit mutants, filtered by data-inferred Mahalanobis distance on 10 RDKit descriptors for plausibility, totaling 13.8M molecules.

[22] Kathryn E. Kirchoff et al. "SALSA: Semantically-aware latent space autoencoder". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 38.12 (2024), pp. 13211–13219. [url]

Park et al., 2024 [23] propose a contrastive framework for chemical language models that pulls together SMILES of structurally similar molecules, using ECFP4 cosine similarity to define positive pairs. A substructure prediction head jointly predicts binary fingerprints, guiding representation learning toward structural awareness. Applied to MoLFormer-XL, Moleco achieves +1.8% ROC-AUC (classification) and –7.3% RMSE (regression) over baselines on MoleculeNet, with ablations confirming the necessity of both training signals.

[23] Jun-Hyung Park et al. "Moleco: Molecular contrastive learning with chemical language models for molecular property prediction". In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*. Association for Computational Linguistics, 2024, pp. 408–420. [url]

Qian et al., 2024 [24] pretrain a SMILES encoder via contrastive learning on enumerated SMILES pairs using NT-Xent loss. Each molecule yields two randomized SMILES; all other batch samples serve as negatives. The encoder is trained on 37.96M molecules from ZINC Clean Leads using a Transformer + MLP projection head. A GPT-style decoder (CON-GPT) uses the encoder for molecule generation on "MOSES" and "GuacaMol"; a dual-encoder model (CON-UniT) uses it for compound–protein interaction (CPI) prediction on "DAVIS" and "Celegans".

[24] Ying Qian et al. "CONSMI: Contrastive learning of SMILES representations for improved molecular generation". In: *Molecules* 29.2 (2024), p. 495. [url]

Quiles et al., 2024 [25] pretrain using contrastive learning with NT-Xent loss on 1.3M ZINC15 molecules ($\leq$50 atoms; H, C, N, O, F, S, Cl); two randomized SMILES are encoded with two separate LSTMs and passed through a shared 2-layer projection head. For QM9, the frozen encoder feeds a 2-layer MLP trained to regress 7 properties; MAE improves on 5/7 (e.g., U_0: 0.3053 $\rightarrow$ 0.2439), but worsens (0.2787) with ZINC-added pretraining. On a 10k-sample ionic liquid dataset, the model improves cation property prediction (e.g., logP, HOMO-LUMO gap), but shows inconsistent results for anions.

[25] Marcelo G. Quiles et al. "Enhancing low-cost molecular property prediction with contrastive learning on SMILES representations". In: *Computational Science and Its Applications – ICCSA 2024 Workshops. Lecture Notes in Computer Science, vol 14823.* Springer, Cham, 2024. [url]

MoleSG by Shen et al., 2024 [26] fuses SMILES (ChemBERTa-based [2]) and graphs (CoMPT graph encoder [27]) using separate encoders. Outputs are concatenated with embeddings and passed to a RoBERTa-style backbone. Non-overlapping masks align atoms in only one modality at a time. Reconstructs masked SMILES tokens with a Transformer decoder and masked graph nodes with a GIN decoder. Total loss: masked language modeling for SMILES and "Structure-aware Contrastive Error" for graphs, which penalizes misalignment between predicted and true *node* embeddings using fixed atom-wise correspondence (no Sinkhorn). Pretrained on 250K ZINC15 molecules (mask ratio: 15% SMILES, 25% graph), and evaluated across 14 MoleculeNet datasets. Outperforms models like MolCLR [20], UniMAP [28]; ablations show that non-overlapping masking and graph-only fine-tuning are critical.

---

[26] Ao Shen et al. "Complementary multi-modality molecular self-supervised learning via non-overlapping masking for property prediction". In: *Briefings in Bioinformatics* 25.4 (2024), bbae256. [url]

this slide is intentionally left blank