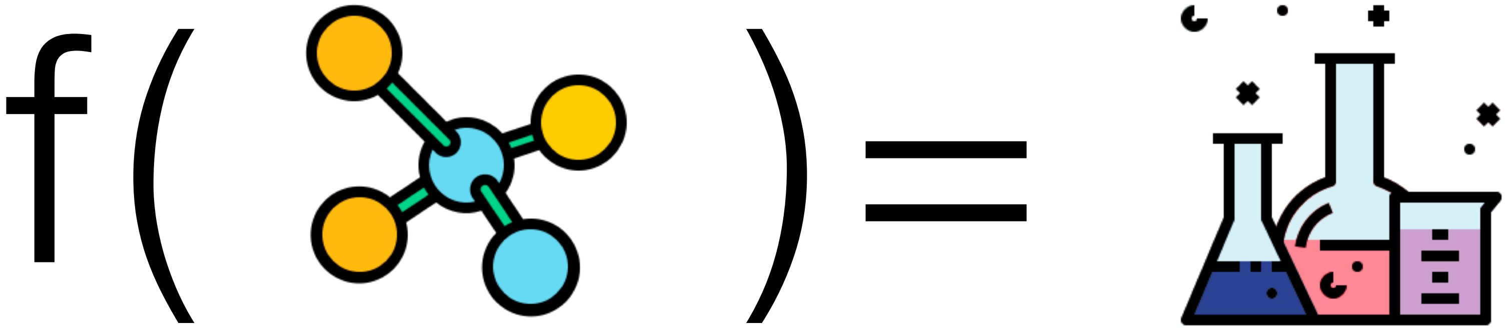
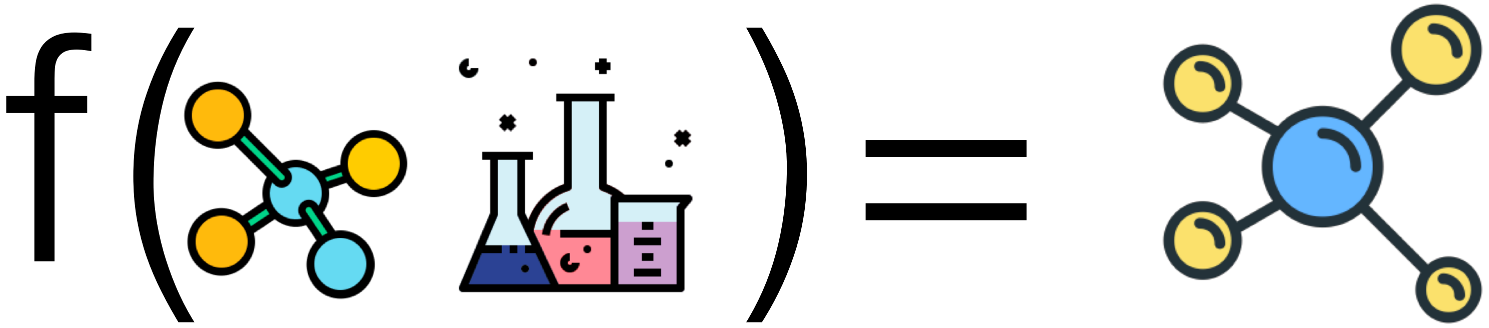
# 4. De Novo Design of Drug Molecules

# De novo design as the inverse task of molecule property prediction

**QSAR**: given the molecular descriptors, predict the chemical property.



**De novo**: want a molecule with certain property.



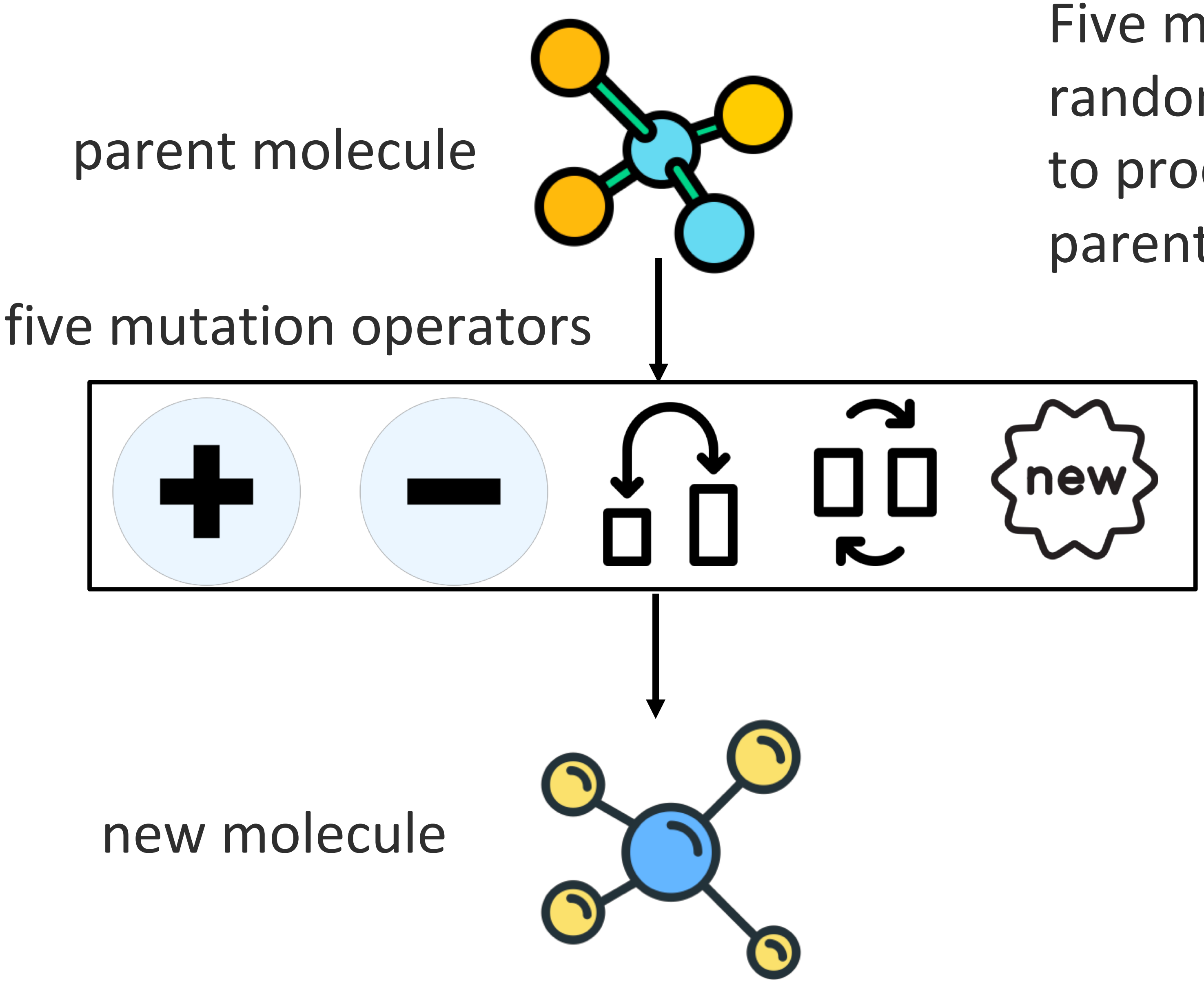
# Why Need De Novo Design

Design new therapeutic molecules

Generate molecules with high potency

Modify molecules to increase potency

# Challenges of Traditional De Novo Methods



parent molecule

five mutation operators

new molecule

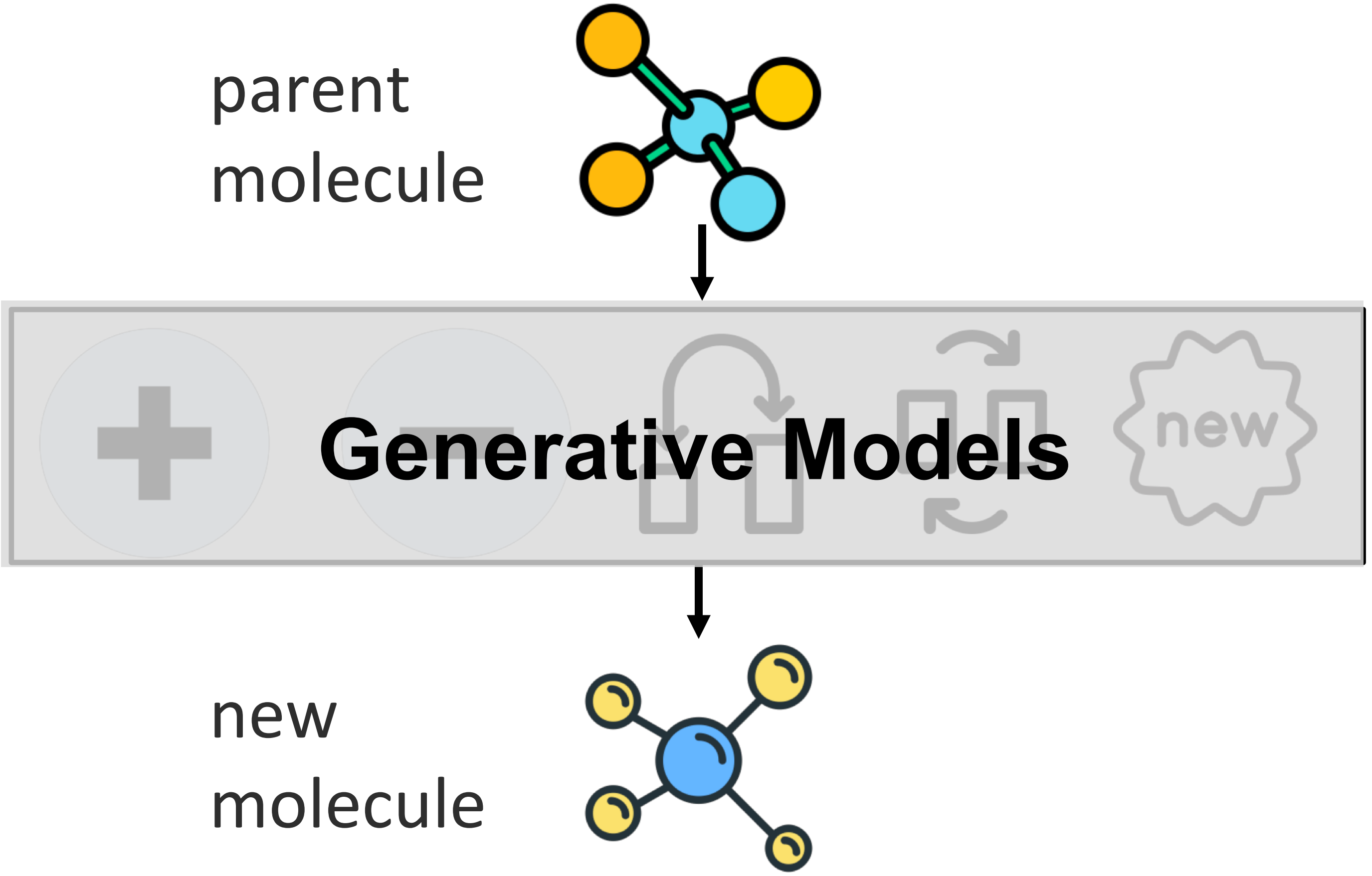
Five mutation operators, i.e., add, cut, replace random, replace like, and new random, is used to produce a new molecule from the selected parent molecule.

→ Combinatorial optimization, thus intractable

→ Large output space and big validation cost

The range of potential drug-like molecules is estimated to be between $10^{23}$ and $10^{60}$.

# Generative Models for De Novo Design



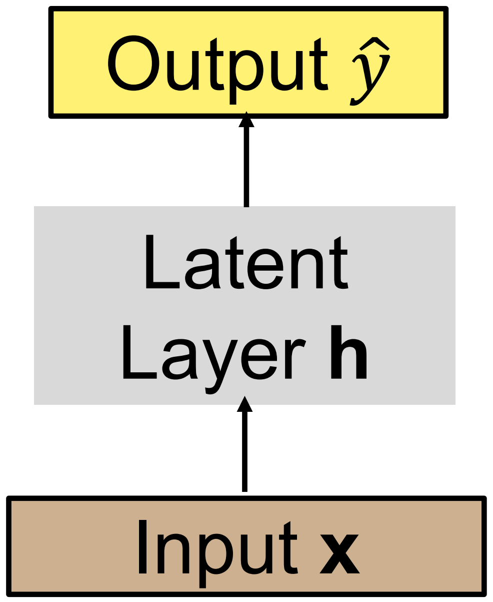
**Variational Autoencoders**

CVAE (2016)
Grammar VAE (ICML 2017)
JT-VAE (ICML 2018)
Constrained VAE (NIPS 2018)
GCPN (NIPS 2018)

❑ learn the probability distribution of molecule structures (e.g., characters in a SMILES string) and then generate new structures (e.g., strings) which correspond to chemically meaningful molecule compound.

# Autoencoders for De Novo vs. Classifiers for QSAR

## Classifier



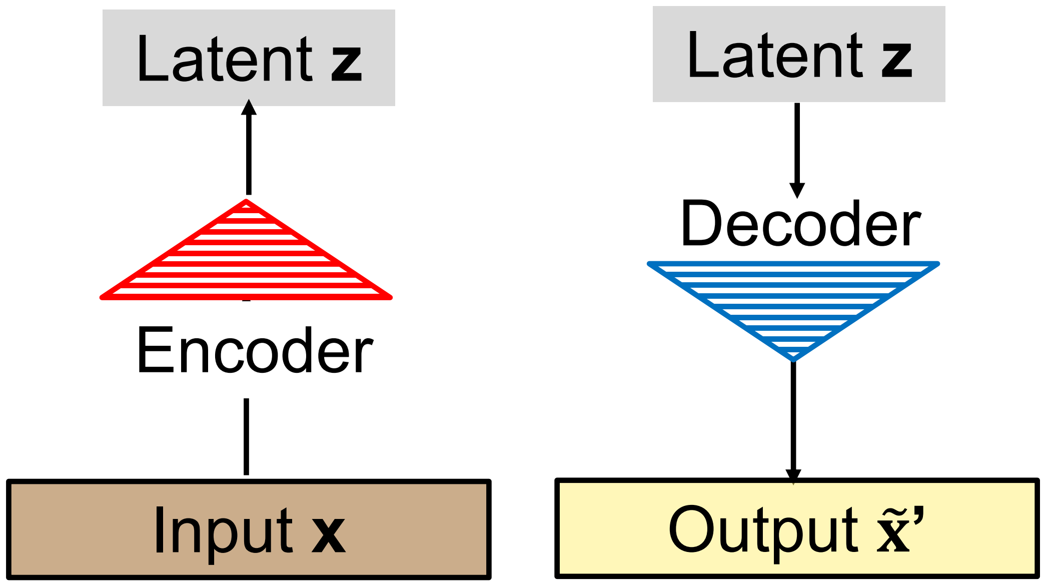
Output $\hat{y}$

Latent Layer **h**

Input **x**

Input raw molecule data or descriptors, Output drug properties

## Autoencoders



Latent **z**

Encoder

Input **x**

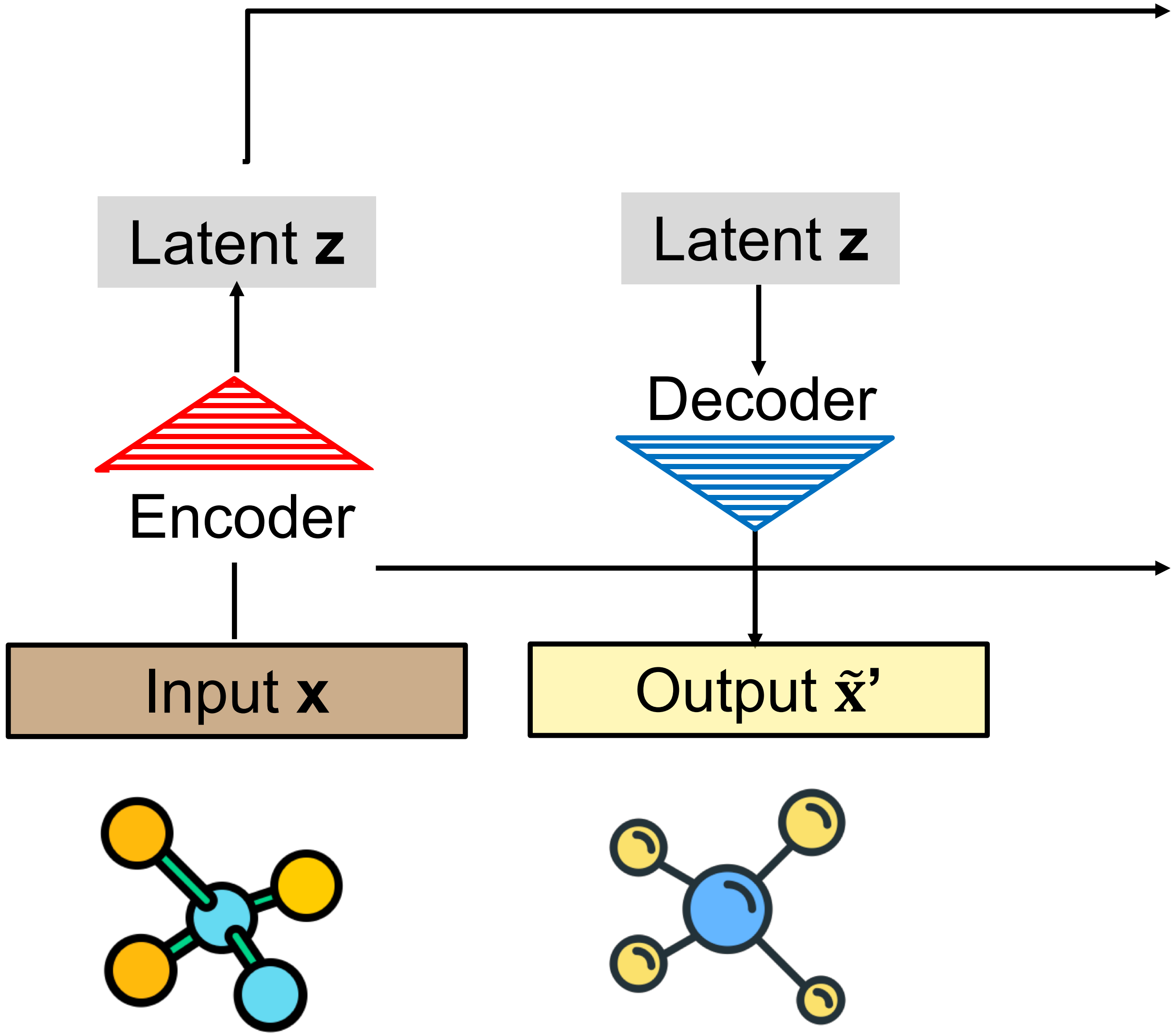
Latent **z**

Decoder

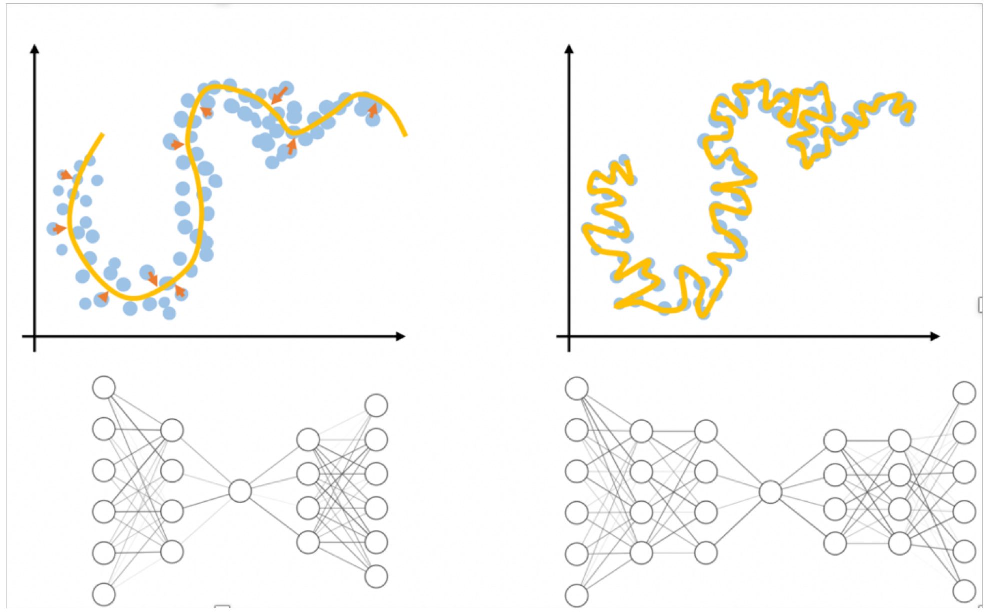
Output $\tilde{\mathbf{x}}'$

Reconstruct input molecule data by squeezing the data through a latent layer

# Autoencoders

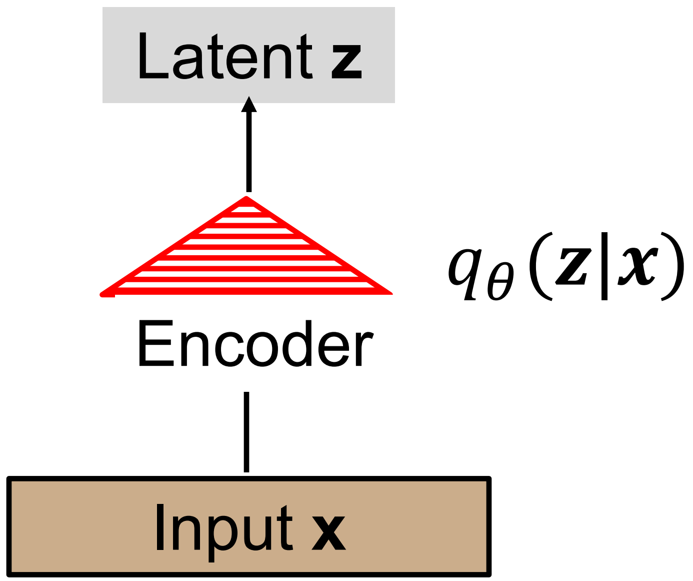


Subspaces whose dimensions correspond to meaningful concepts where most data lies



The # of hidden layers in encoder and decoder control the nonlinearity allowed

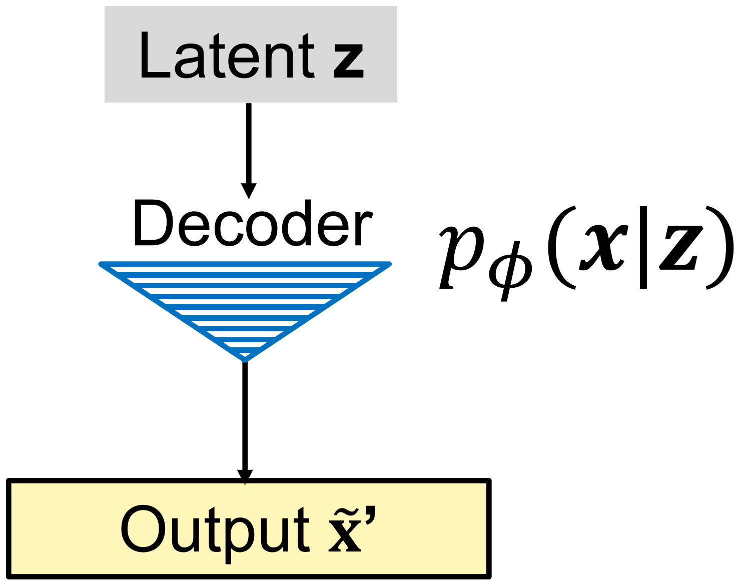
# Variational Autoencoders: Encoder



- The encoder learn an efficient compression of the data into this lower-dimensional space.
- It outputs parameters to $q_\theta(\boldsymbol{z}|\boldsymbol{x})$, a Gaussian probability density.

Diederik P Kingma; Welling, Max (2013). Auto-Encoding Variational Bayes. arXiv:1312.6114

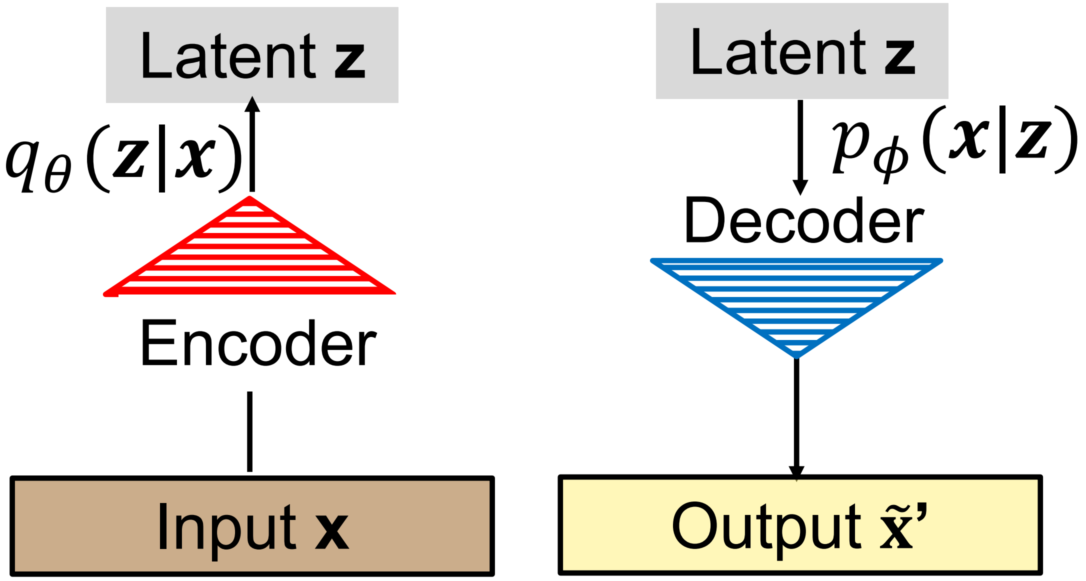
# Variational Autoencoders: Decoder


Latent **z**
Decoder
$p_\phi(\boldsymbol{x}|\boldsymbol{z})$
Output $\tilde{\mathbf{x}}$’

- The decoder learn learned to reconstruct the input data given its latent representation.
- It achieves this via sampling from the output distribution of the encoder to get noisy values of the representations.

# Variational Autoencoders: Training

$p_\phi(\boldsymbol{z})$: prior distribution of the latent representation



Latent **z**
$q_\theta(\boldsymbol{z}|\boldsymbol{x})$
Encoder
Input **x**
Latent **z**
$p_\phi(\boldsymbol{x}|\boldsymbol{z})$
Decoder
Output $\tilde{\mathbf{x}}$'

❑ The reconstruction error of the decoder is reduced by maximizing the log-likelihood of $p_\phi(\boldsymbol{x}|\boldsymbol{z})$

❑ Simultaneously, the encoder is regularized to approximate the latent variable distribution $p_\phi(\boldsymbol{z})$ by minimizing the Kullback-Leibler divergence

$$KL(q_\theta(\boldsymbol{z}|\boldsymbol{x}), p_\phi(\boldsymbol{z}))$$

❑ If the prior follow a multivariate Gaussian distribution with zero mean and unit variance, then the loss function is

$$L(\theta, \phi) = -E_{q_\theta(\boldsymbol{z}|\boldsymbol{x})}\left[\log(p_\phi(\boldsymbol{x}|\boldsymbol{z}))\right] + KL(q_\theta(\boldsymbol{z}|\boldsymbol{x}), p_\phi(\boldsymbol{z}))$$

# Challenges of Molecule Generation

Generate molecules

- ○ with desired property
- ○ syntactically correct molecules
- ○ semantically correct
- ○ High molecular property scores

# Challenges of Molecule Generation

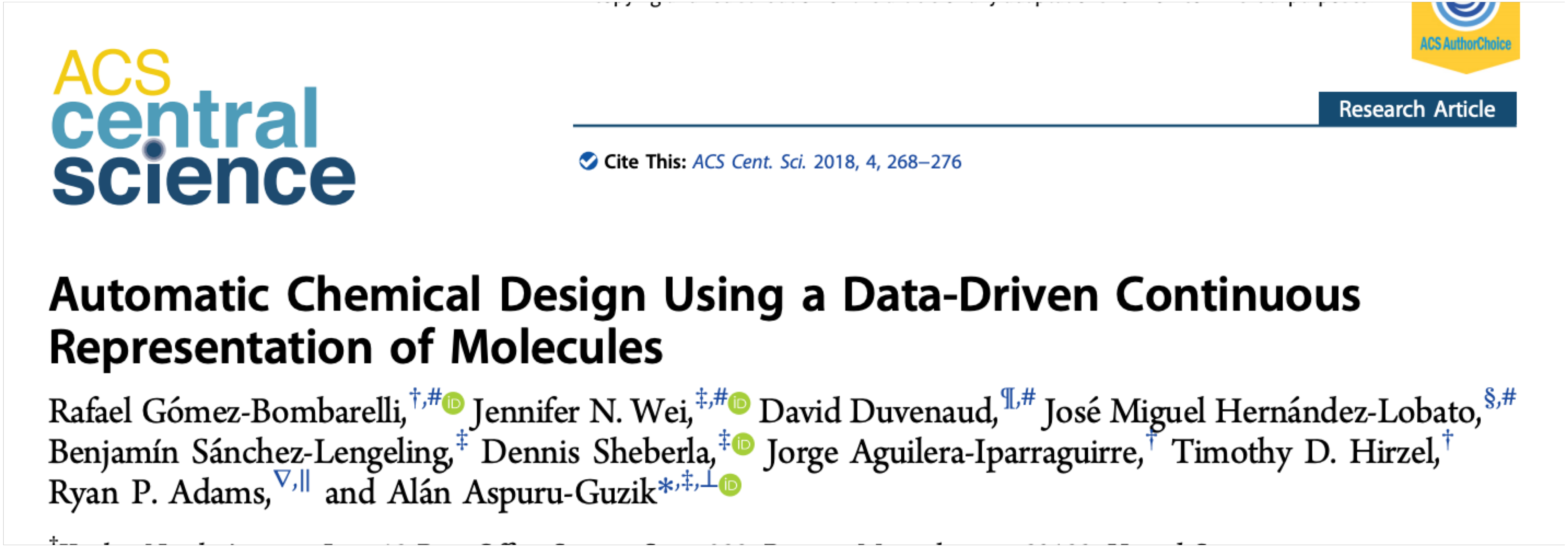
Generate molecules

✓ with desired property

○ syntactically correct molecules

○ semantically correct

○ High molecular property scores



ACS central science

ACS AuthorChoice

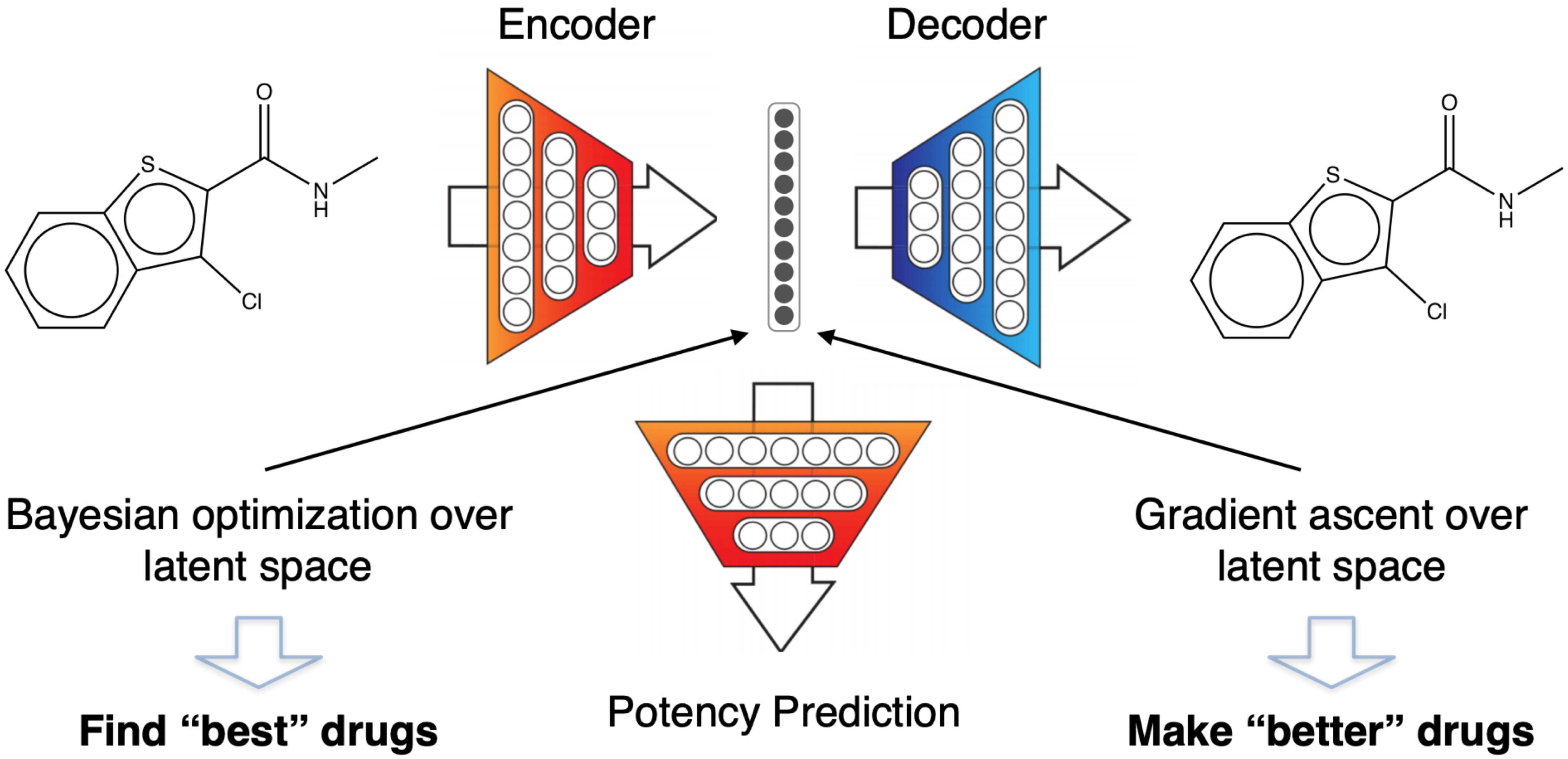
Research Article

Cite This: ACS Cent. Sci. 2018, 4, 268−276

# Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules

Rafael Gómez-Bombarelli,[†,#] Jennifer N. Wei,[‡,#] David Duvenaud,[¶,#] José Miguel Hernández-Lobato,[§,#] Benjamín Sánchez-Lengeling,[‡] Dennis Sheberla,[‡] Jorge Aguilera-Iparraguirre,[†] Timothy D. Hirzel,[†] Ryan P. Adams,[∇,||] and Alán Aspuru-Guzik[*,‡,⊥]

# De Novo Design with VAE (CVAE, ACS Central Science 2018)



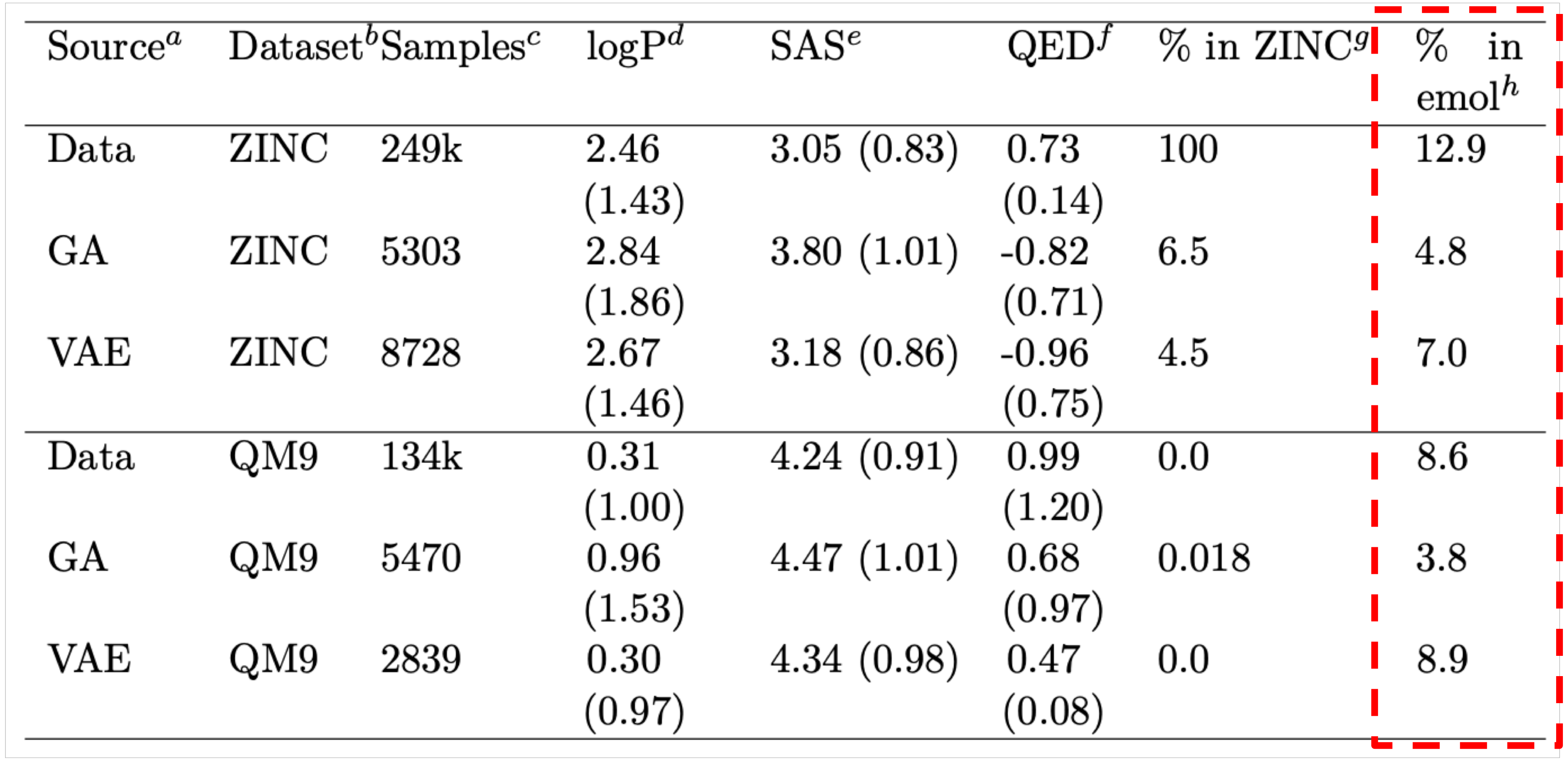
Train Gaussian Process (GP) to maximize property scores.
A new point was then selected by sequentially maximizing
the expected improvement acquisition based on GP model.

Gomez-Bombarelli et al.,Automatic chemical design using a data-driven continuous representation of molecules, ACS Central Science 2018

# De Novo Design with VAE (CVAE, ACS Central Science 2018)

QM9: 108,000 molecules with fewer than 9 heavy atoms

ZINC: 250,000 drug-like commercially available molecules



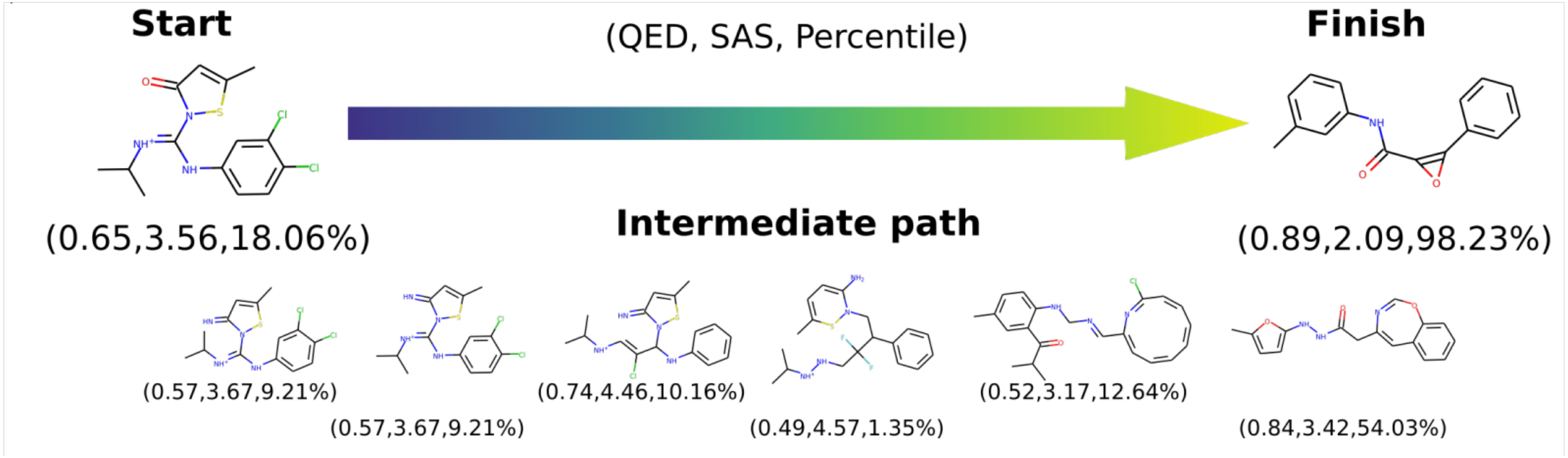
| Source[a] | Dataset[b] | Samples[c] | logP[d] | SAS[e] | QED[f] | % in ZINC[g] | % in emol[h] |
|---|---|---|---|---|---|---|---|
| Data | ZINC | 249k | 2.46 (1.43) | 3.05 (0.83) | 0.73 (0.14) | 100 | 12.9 |
| GA | ZINC | 5303 | 2.84 (1.86) | 3.80 (1.01) | -0.82 (0.71) | 6.5 | 4.8 |
| VAE | ZINC | 8728 | 2.67 (1.46) | 3.18 (0.86) | -0.96 (0.75) | 4.5 | 7.0 |
| Data | QM9 | 134k | 0.31 (1.00) | 4.24 (0.91) | 0.99 (1.20) | 0.0 | 8.6 |
| GA | QM9 | 5470 | 0.96 (1.53) | 4.47 (1.01) | 0.68 (0.97) | 0.018 | 3.8 |
| VAE | QM9 | 2839 | 0.30 (0.97) | 4.34 (0.98) | 0.47 (0.08) | 0.0 | 8.9 |

% generated molecules found in e-molecule database

Gomez-Bombarelli et al.,Automatic chemical design using a data-driven continuous representation of molecules, ACS Central Science 2018

# De Novo Design with VAE (CVAE, ACS Central Science 2018)

The property scores improve during the optimization



Gomez-Bombarelli et al.,Automatic chemical design using a data-driven continuous representation of molecules, ACS Central Science 2018

# Challenges of Molecule Generation

Generate molecules

✓ with desired property

✓ syntactically correct molecules

○ semantically correct

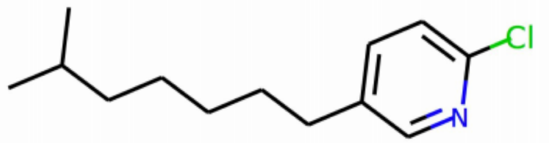
○ High molecular property scores

# Grammar Variational Autoencoder

Matt J. Kusner, Brooks Paige, José Miguel Hernández-Lobato

ICML' 17

# Grammar VAE (ICML’ 17)



CC(C)CCCCCc1ccc(Cl)nc1

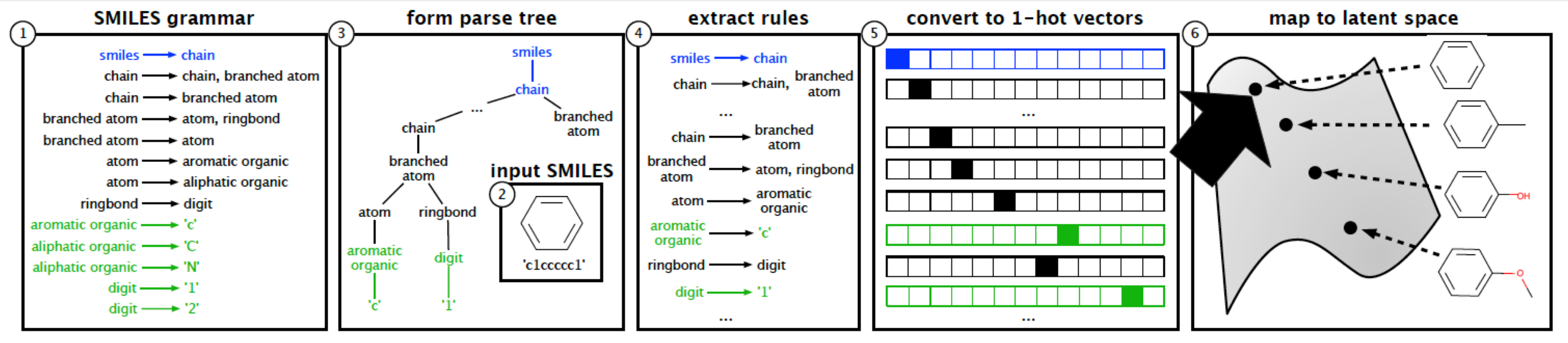
**Challenge**: Molecule is constructed using a “formal language”, any syntax change will cause error

**Opportunity**: Syntax is known and fixed. Parse is unique.

**Goal:** Learning syntactic rules to produce valid outputs

# Grammar VAE (ICML’ 17)

Learning syntactic rules to produce valid outputs



Input SMILES string and grammar

SMILES grammar to parse SMILES string into a parse tree

Decompose tree into a sequence of production rules by pre-order traversal on the branches

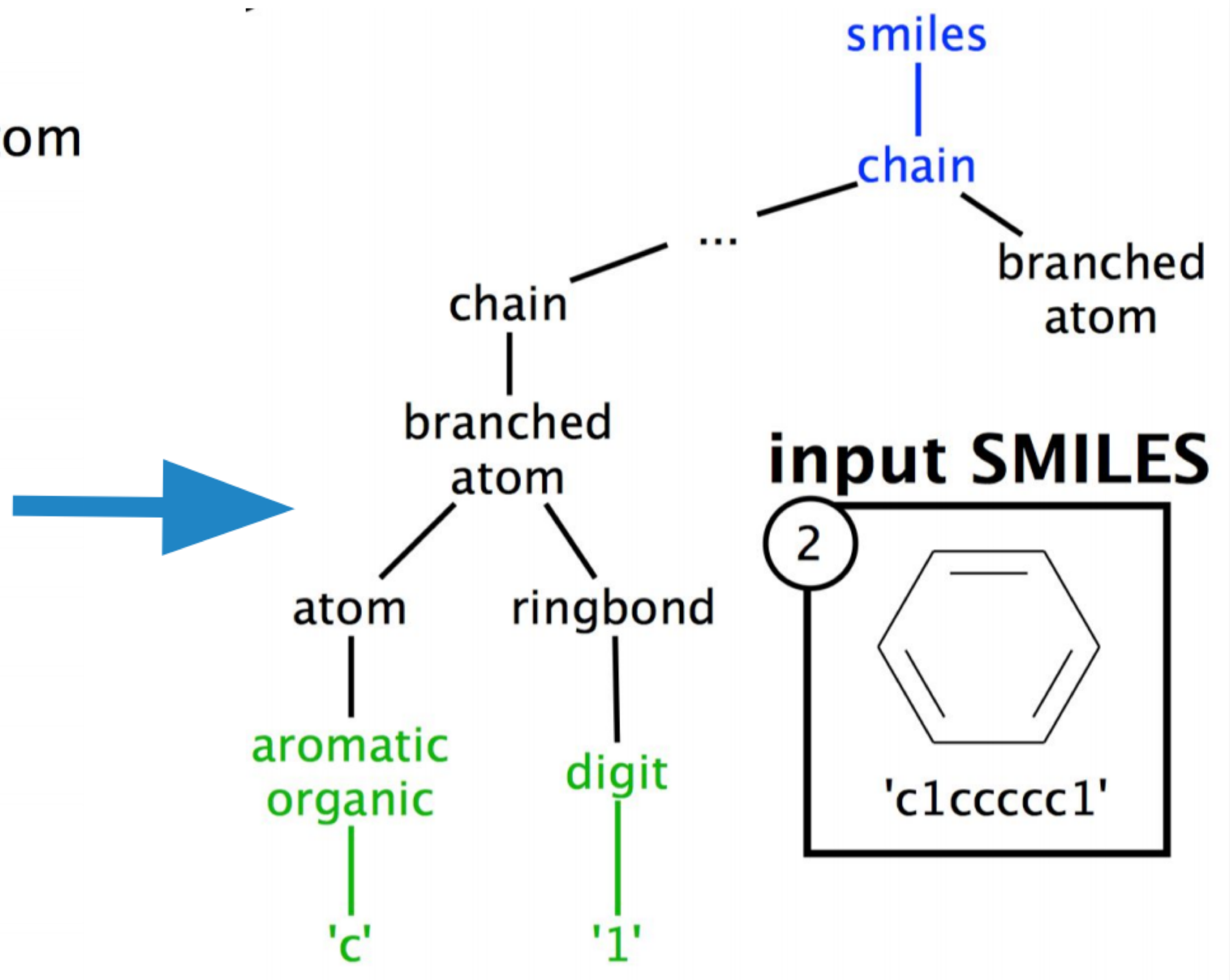
Convert rules into 1-hot vector

Map into a continuous vector using CNN

Kusner et al., Grammar Variational Autoencoder, ICML' 17

# Grammar VAE (ICML’ 17)

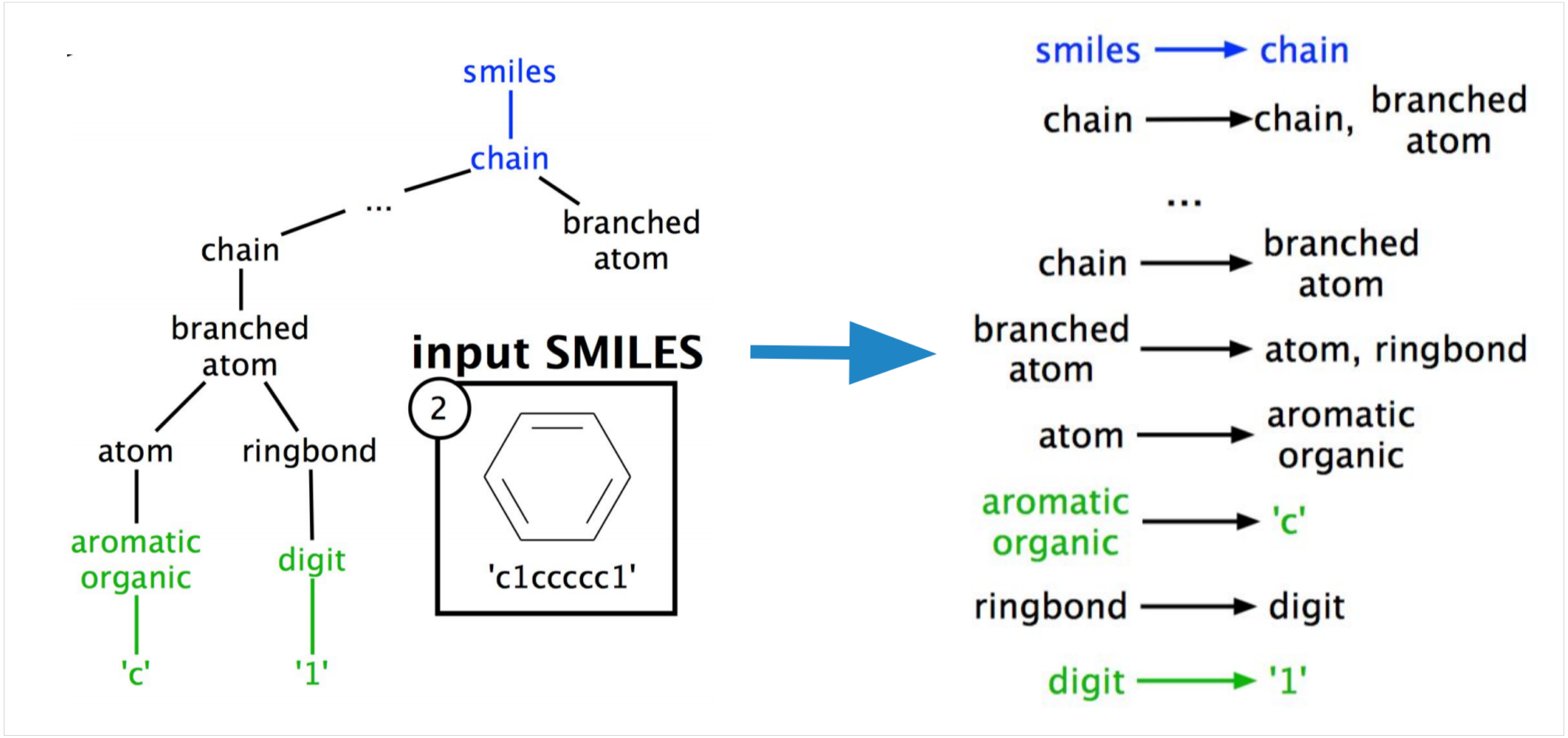
smiles → chain
chain → chain, branched atom
chain → branched atom
branched atom → atom, ringbond
branched atom → atom
atom → aromatic organic
atom → aliphatic organic
ringbond → digit
aromatic organic → 'c'
aliphatic organic → 'C'
aliphatic organic → 'N'
digit → '1'
digit → '2'



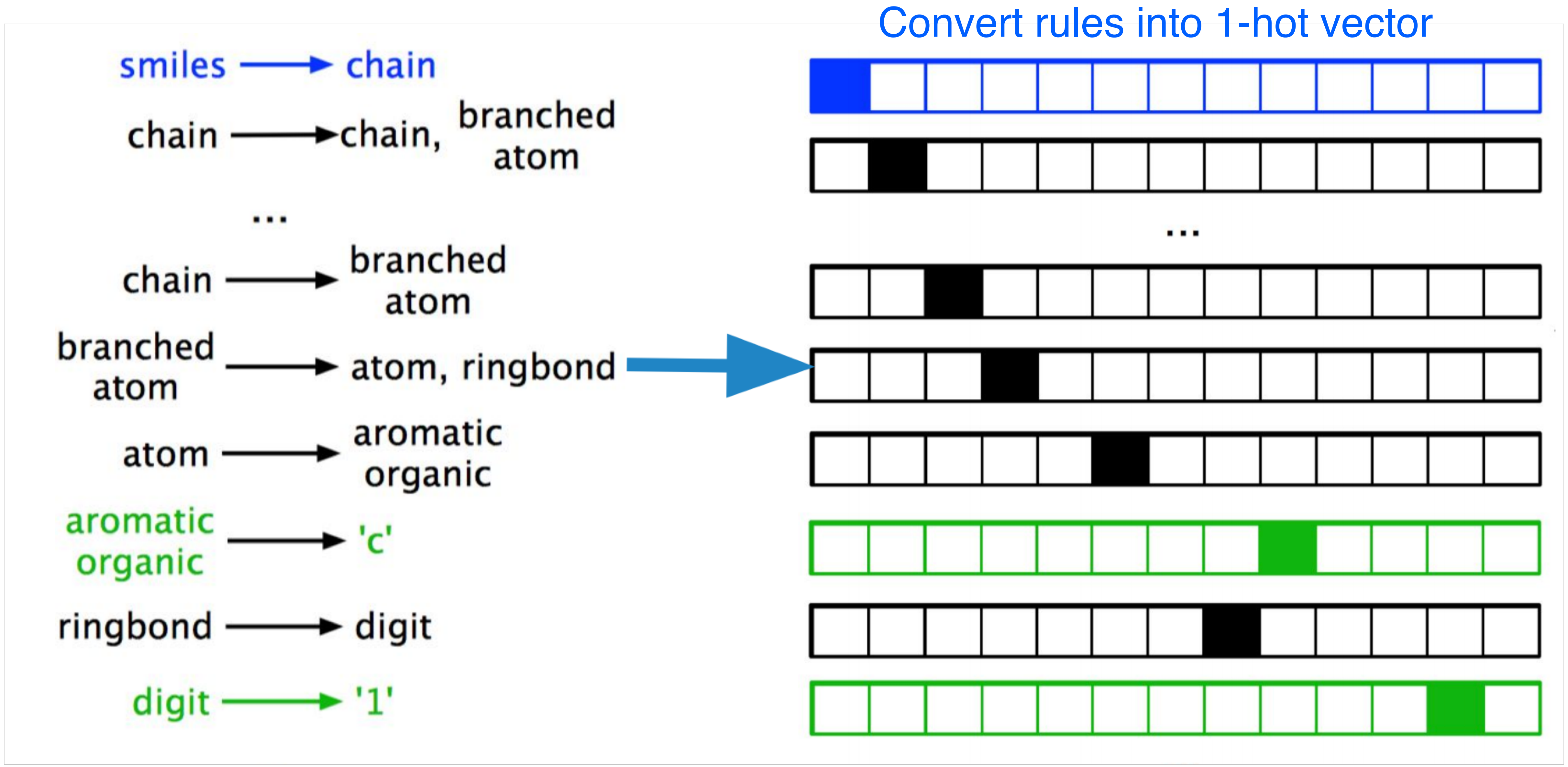
SMILES grammar to parse SMILES string into a parse tree

# Grammar VAE (ICML' 17)

Extract production rules by pre-order traversal on the branches.

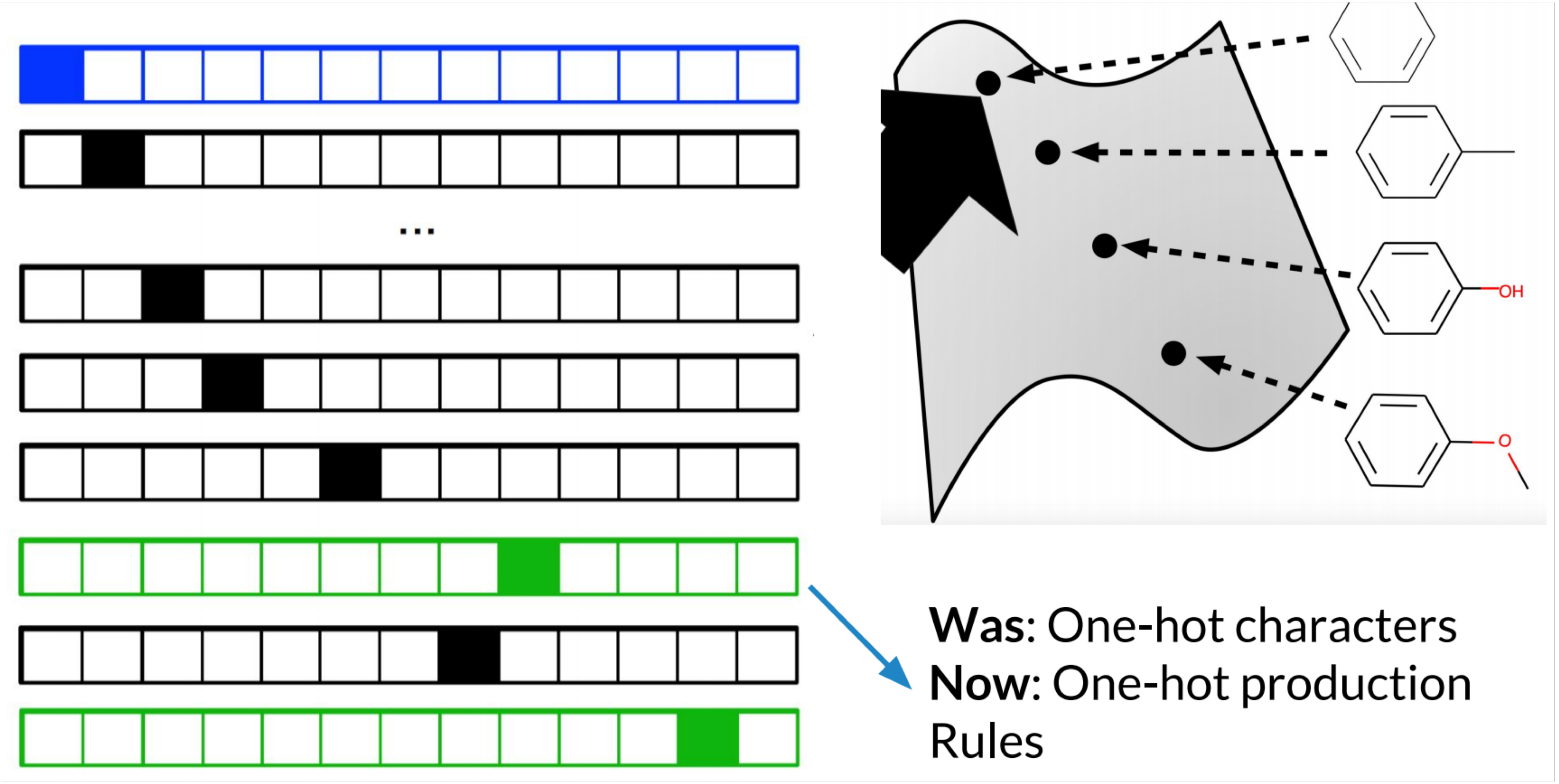


# Grammar VAE (ICML’ 17)



# Grammar VAE (ICML' 17)

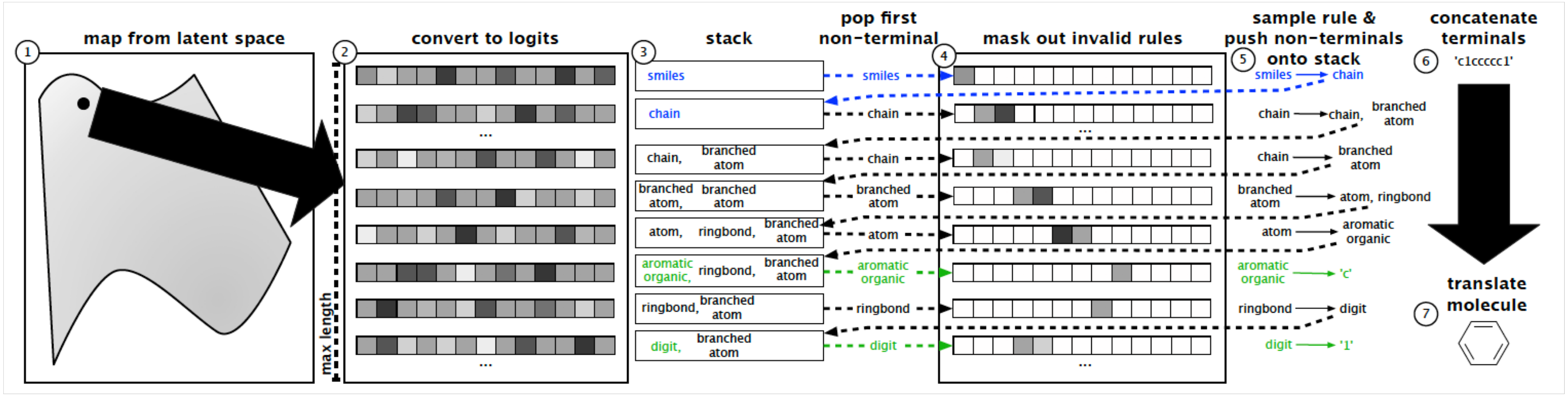
Map into a continuous vector using CNN



**Was**: One-hot characters
**Now**: One-hot production Rules

# Grammar VAE (ICML' 17)

## Decode continuous vectors back to SMILES strings

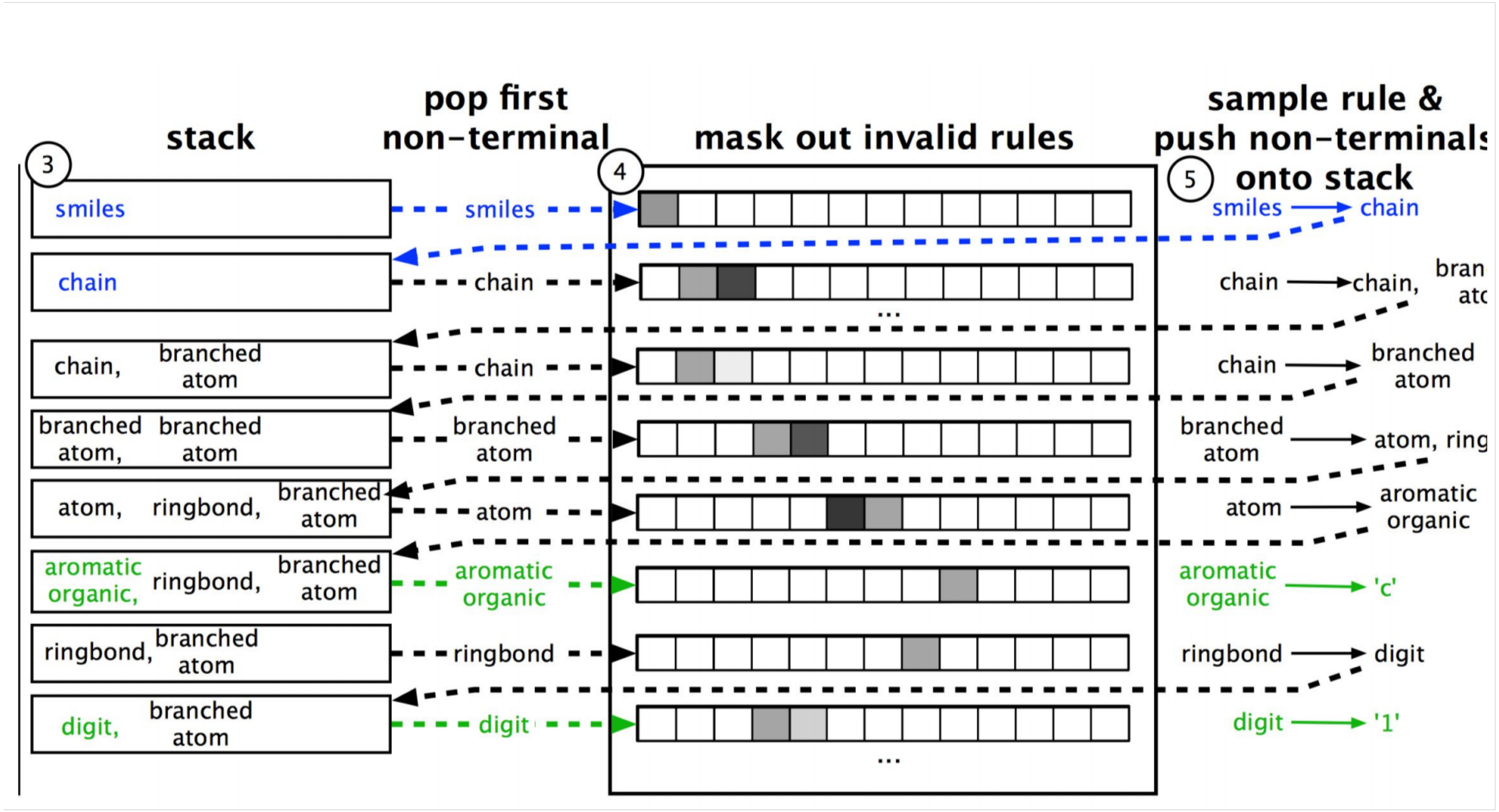


Pass the continuous vector using RNN to produce vectors or logits

A "pushdown automation" algorithm to select valid rules and construct SMILES

Kusner et al., Grammar Variational Autoencoder, ICML' 17

# Grammar VAE (ICML’ 17)



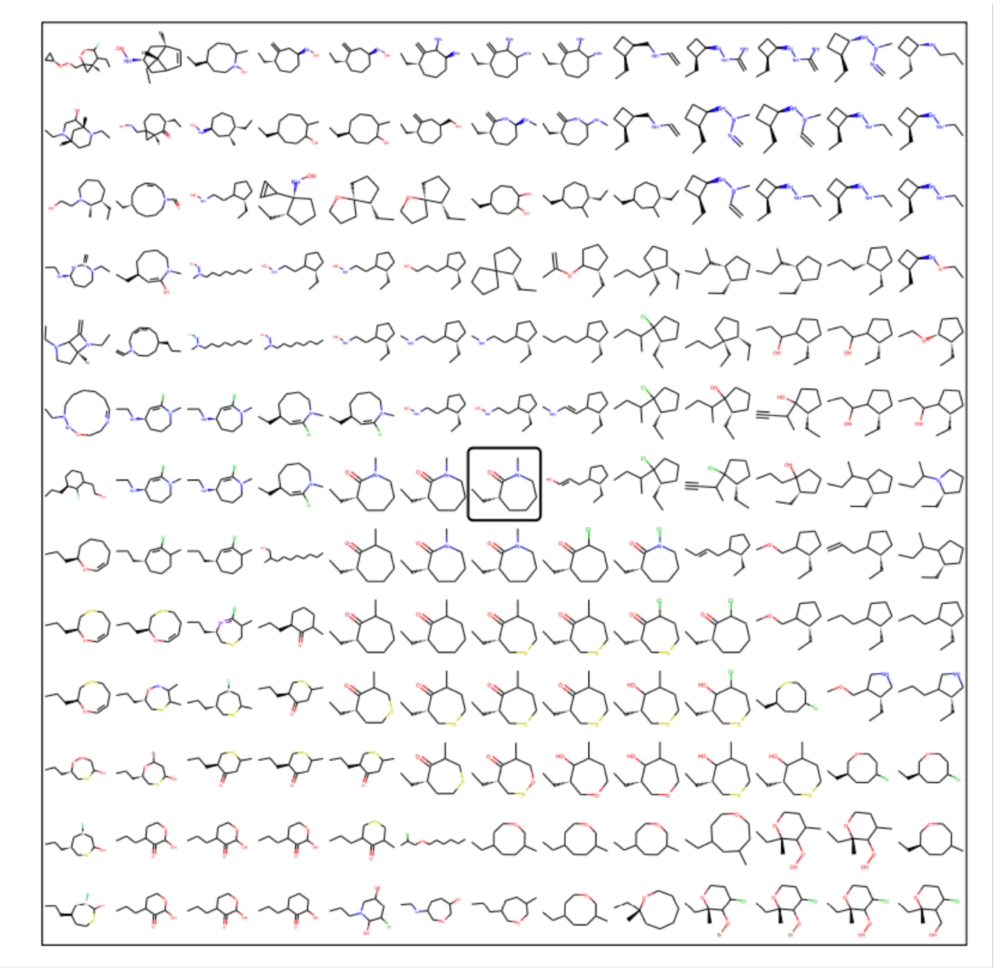
Kusner et al., Grammar Variational Autoencoder, ICML' 17

# Grammar VAE (ICML’ 17)

Goal: maximize the water-octanol partition coefficient (logP), an important metric in drug design that characterizes the drug-likeness of a molecule.

| | % of valid | Avg. Score |
|---|---|---|
| CVAE | 0.17 (0.05) | -54.66 (2.66) |
| GVAE | 0.31 (0.07) | -9.57 (1.77) |

GVAE produces a coherent latent space of molecules.



# Challenges of Molecule Generation

Generate molecules

✓ with desired property

✓ syntactically correct molecules

✓ semantically correct

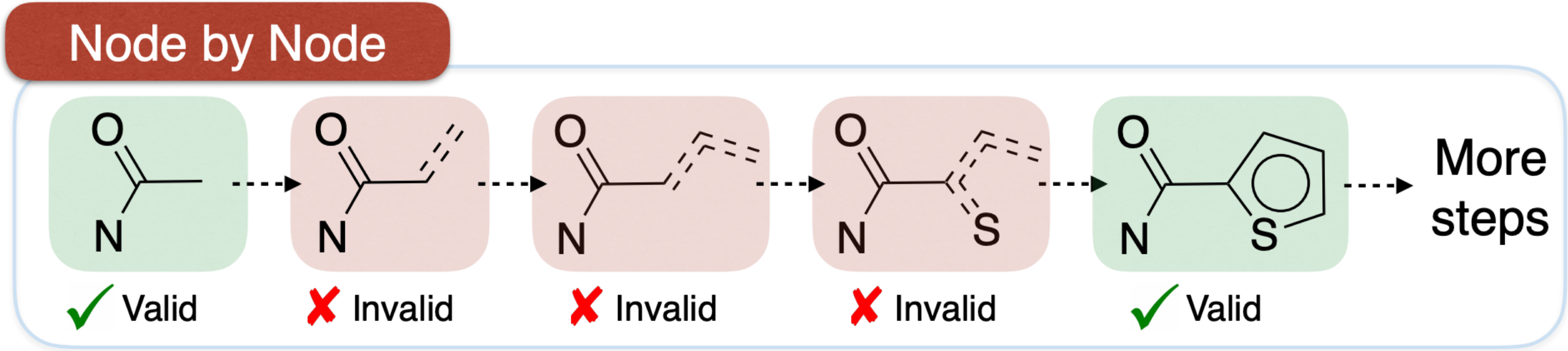
○ High molecular property scores

# Junction Tree Variational Autoencoder for Molecular Graph Generation

Wengong Jin, Regina Barzilay, Tommi Jaakkola

ICML’ 18

# Challenges with earlier model in molecule generation



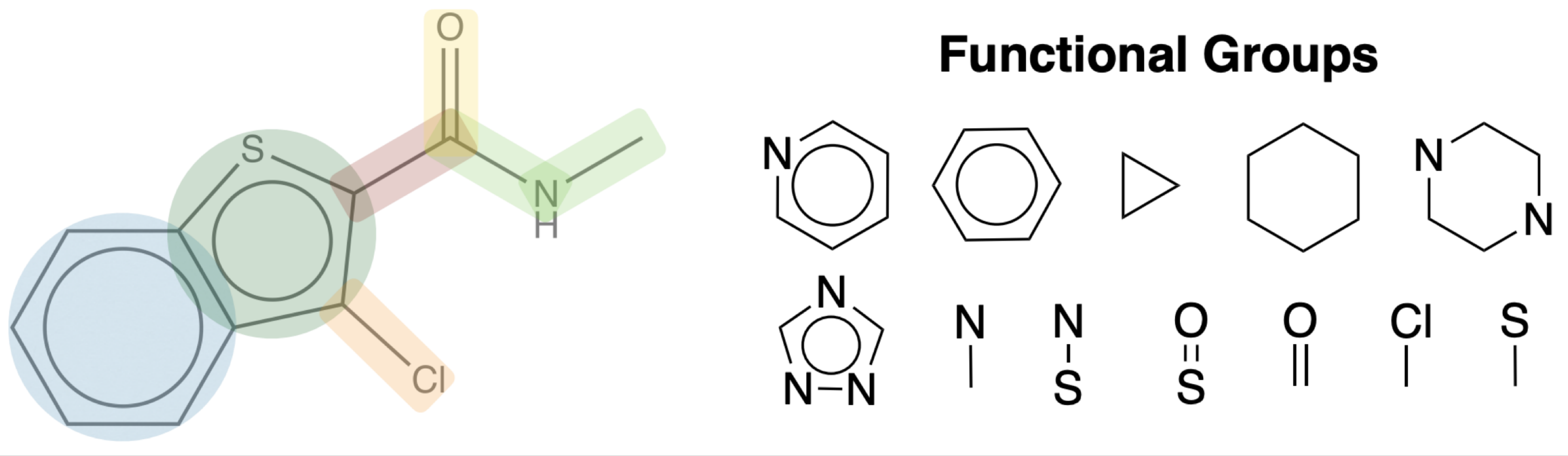
- Not every graphs is chemically valid
- Invalid intermediate states ⟶ hard to validate
- Very long intermediate steps ⟶ difficult to train (Li et al., 2018)

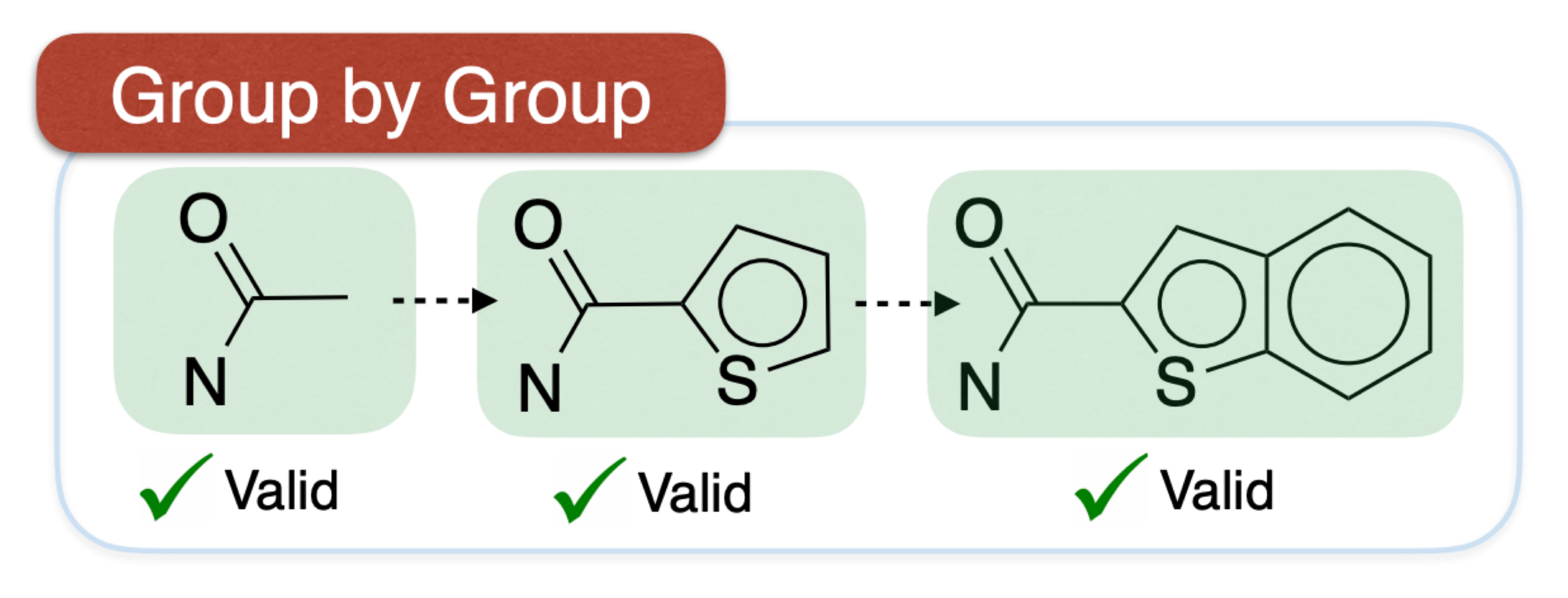
Jin et al., Junction Tree Variational Autoencoder for Molecular Graph Generation, ICML' 18

# De Novo Design with VAE (ICML 2018)

Task: Generating valid molecular graph directly to graph instead of SMILES string

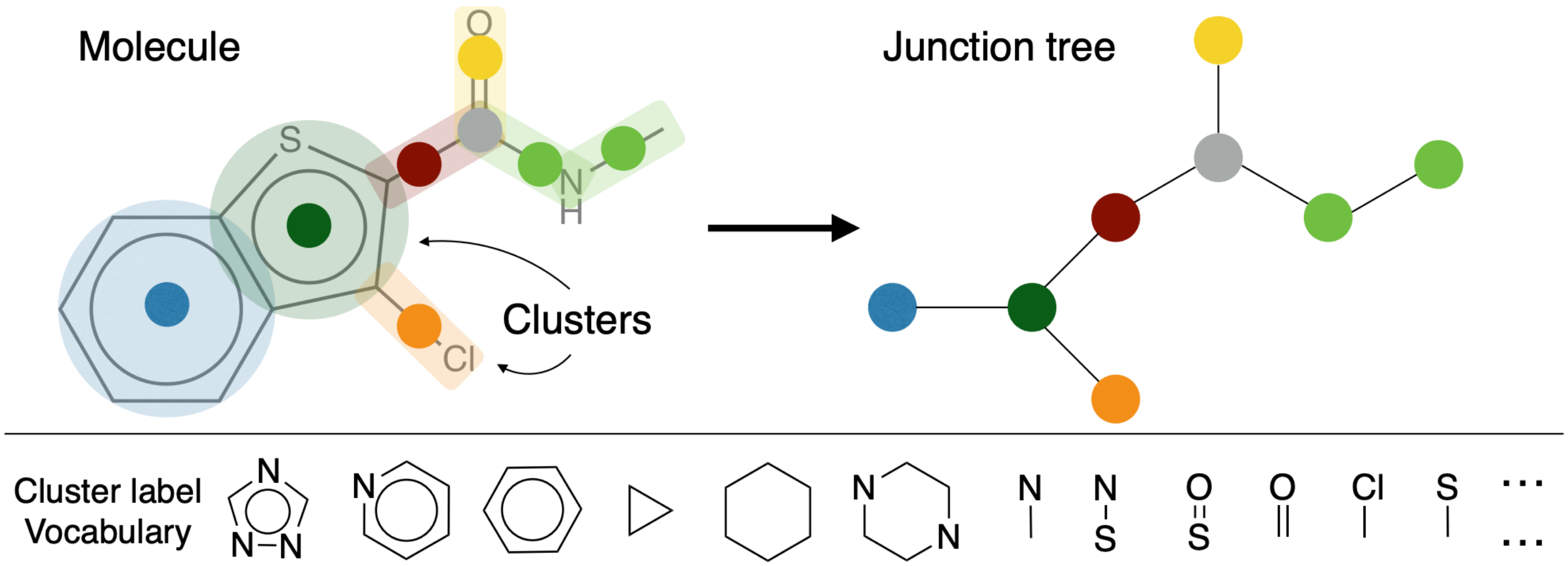
Method: instead of node to node generation, it uses the knowledge of functional group and performs group by group generation.





Jin et al., Junction Tree Variational Autoencoder for Molecular Graph Generation, ICML’ 18

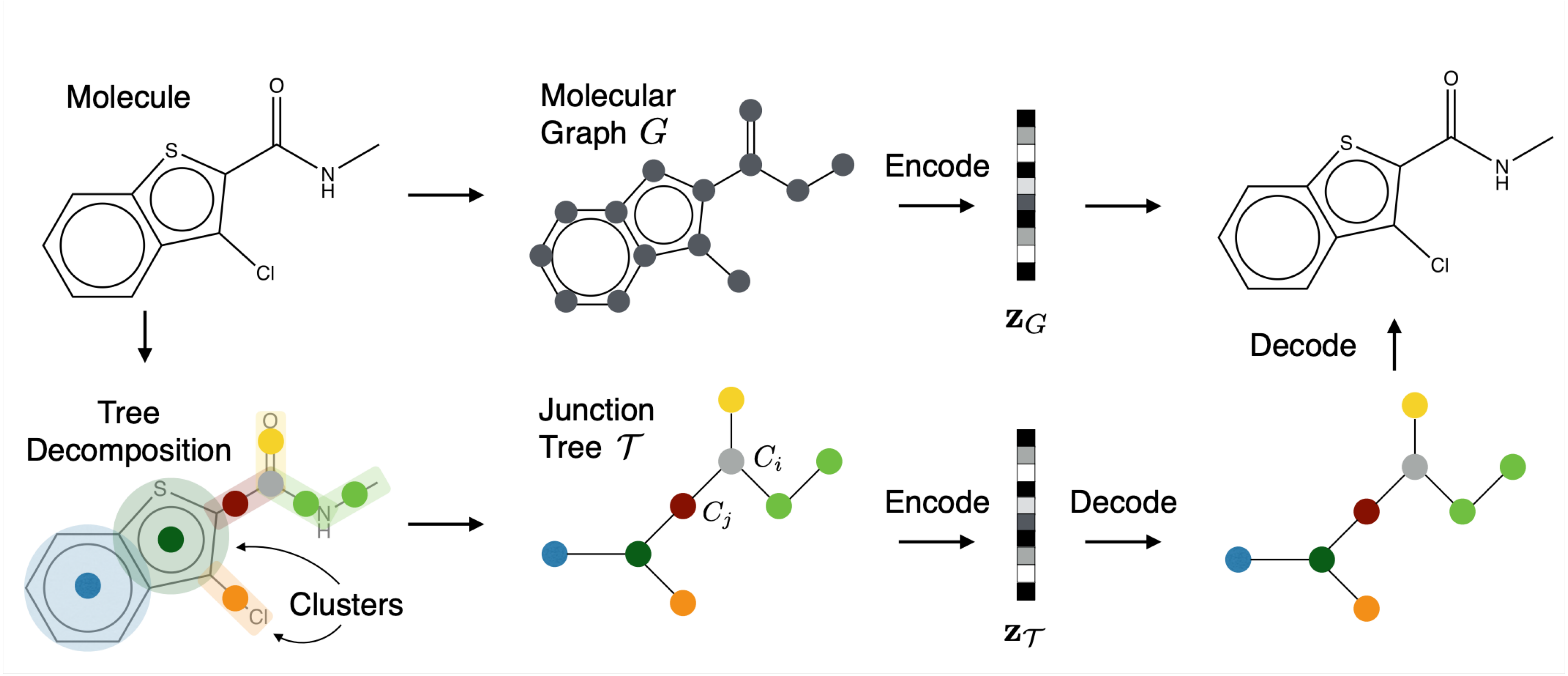
# De Novo Design with VAE (JT-VAE, 2018)



- Generate junction tree → Generate graph group by group
- Vocabulary size: less than 800 given 250K molecules

Jin et al., Junction Tree Variational Autoencoder for Molecular Graph Generation, ICML' 18

# De Novo Design with VAE (JT-VAE, 2018)



Jin et al., Junction Tree Variational Autoencoder for Molecular Graph Generation, ICML’ 18

# Constrained Generation of Semantically Valid Graphs via Regularizing Variational Autoencoders

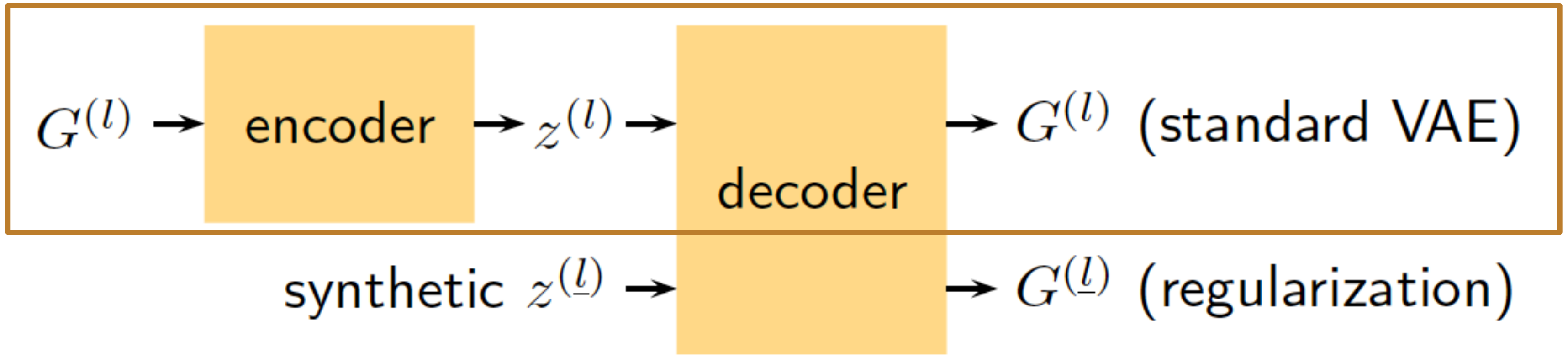
Tengfei Ma, Jie Chen, Cao Xiao,

NeurIPS 18

# Constrained Graph Generation (NeurIPS 2018)

  - How to guarantee the generated sample is a valid graph?
- Ideas:
  - Represent  graphs as concatenation of its node matrix and edge matrix and treat it as an image –> so we can use the same decoder as image
  - an approach to imposing validity constraints in the training of VAEs.

# Constrained Graph Generation (NeurIPS 2018)



- A graph auto-encoder used to generate the graph
- In addition to a standard VAE (within the rectangle), we add a regularization term.
- f(x) is the original VAE loss
- h and g are regularization terms

$$\begin{aligned} \min_{x} \quad & f(x) \\ \text{subject to} \quad & \text{for almost all } z \sim p_x(z), \\ & h_1(x,z) = 0, \dots, h_m(x,z) = 0, \\ & g_1(x,z) \le 0, \dots, g_r(x,z) \le 0. \end{aligned}$$

# Constrained Graph Generation (NeurIPS 2018)

- A Lagrangian relaxation $$-L_{\mathrm{ELBO}}(\theta,\phi)+\mu\sum_i\left[\int g_i(\theta,z)_+^2\,p_\theta(z)\,dz\right]^{\frac{1}{2}}$$

- Training in Standard VAE

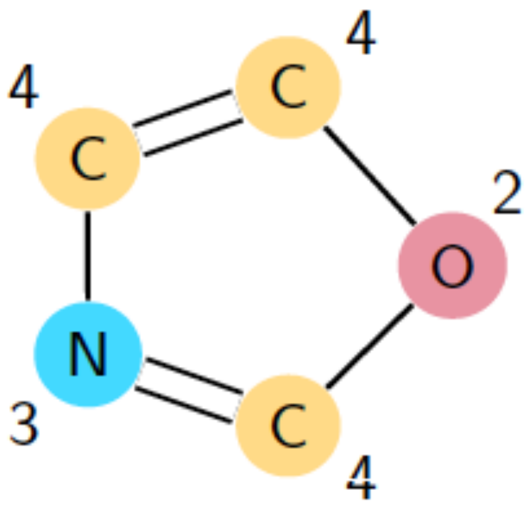
$$\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi};\mathbf{x}^{(i)})=-D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})||p_{\boldsymbol{\theta}}(\mathbf{z}))+\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})}\left[\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)}|\mathbf{z})\right]$$

  - Monte Carlo sampling $\mathbf{z}^{(l)}\sim q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}^{(i)})$

- *Similarly for the regularization term* $$\frac{1}{L}\sum_{l=1}^{L}\log p_{\boldsymbol{\theta}}(\mathbf{x}^{(i)},\mathbf{z}^{(i,l)})$$

$$-L_{\mathrm{ELBO}}(\theta,\phi)+\mu\sum_i g_i(\theta,z)_+,\quad \text{where}\quad z\sim p_\theta(z)$$

# Constrained Graph Generation (NeurIPS 2018)



- **constraints**
  - Valence
    - Expected node capacity (sum of edges) <= valence
  - Connectivity
    - Every node pair much be connected by a path

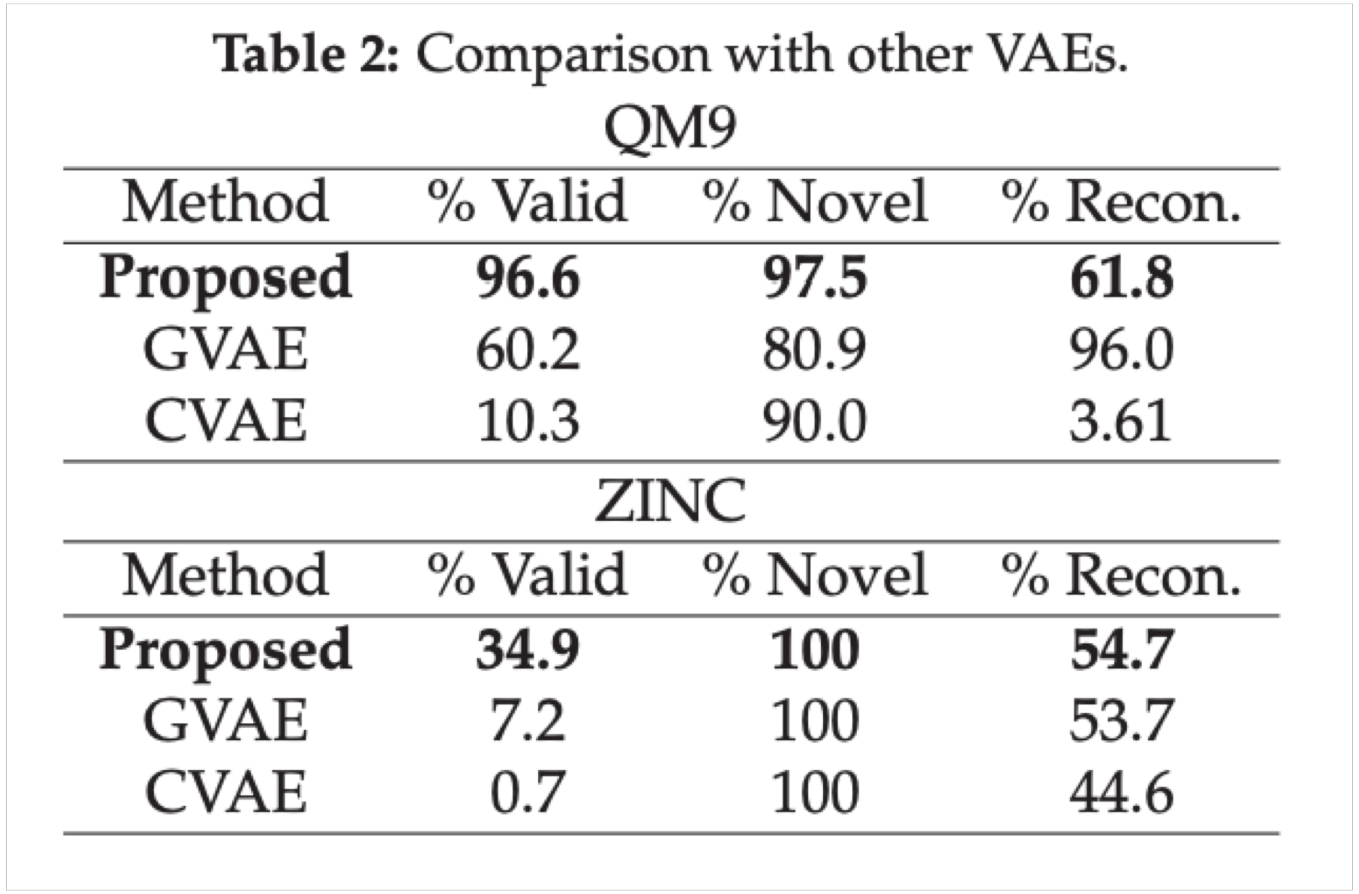


**Table 2:** Comparison with other VAEs.

QM9

| Method | % Valid | % Novel | % Recon. |
|---|---|---|---|
| **Proposed** | **96.6** | **97.5** | **61.8** |
| GVAE | 60.2 | 80.9 | 96.0 |
| CVAE | 10.3 | 90.0 | 3.61 |

ZINC

| Method | % Valid | % Novel | % Recon. |
|---|---|---|---|
| **Proposed** | **34.9** | **100** | **54.7** |
| GVAE | 7.2 | 100 | 53.7 |
| CVAE | 0.7 | 100 | 44.6 |

# Challenges of Molecule Generation

Generate molecules

✓ with desired property

✓ syntactically correct molecules

✓ semantically correct

✓ High molecular property scores

# Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation

Jiaxuan You, Bowen Liu, Rex Ying, Vijay Pande, Jure Leskovec

NeurIPS 18

# GCPN (NIPS 2018)

Generate molecules

- ✓ syntactically correct molecules
- ✓ semantically correct

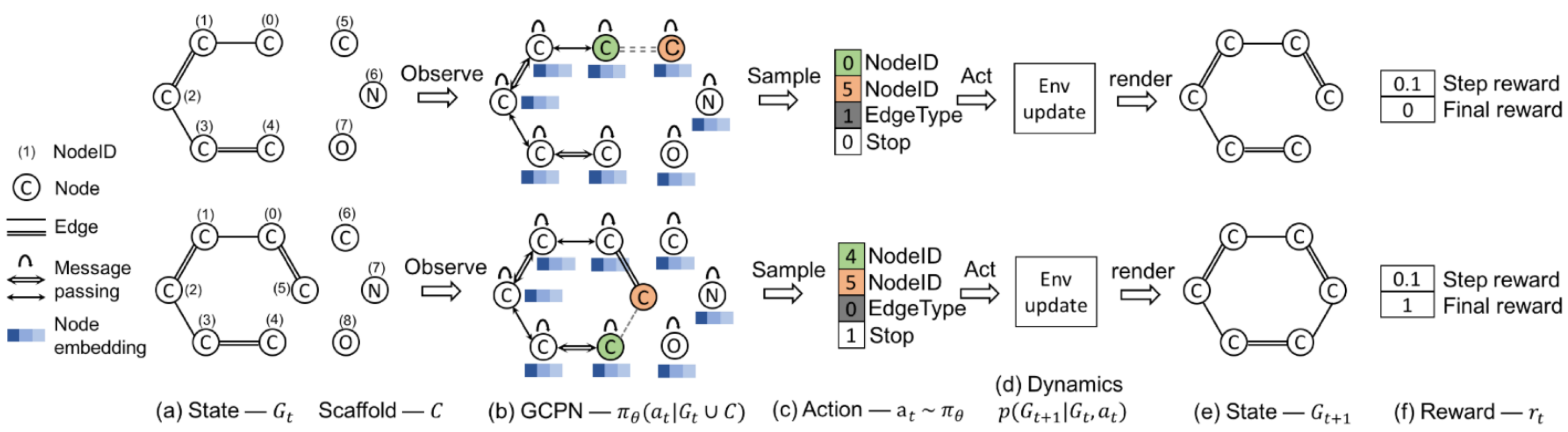
Graph representation enables validity check in each state transition; Adversarial training imitates examples in given data.

- ✓ with desired property
- ✓ High molecular property scores

Reinforcement learning optimizes intermediate and final rewards.

You et al., Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation, NIPS 2018

# GCPN (NIPS 2018)



(a) State — $G_t$ Scaffold — $C$ (b) GCPN — $\pi_\theta(a_t|G_t \cup C)$ (c) Action — $a_t \sim \pi_\theta$ (d) Dynamics $p(G_{t+1}|G_t, a_t)$ (e) State — $G_{t+1}$ (f) Reward — $r_t$

(1) Compute node embedding

$$H^{(l+1)} = \text{AGG}(\text{ReLU}(\{\tilde{D}_i^{-\frac{1}{2}}\tilde{E}_i\tilde{D}_i^{-\frac{1}{2}}H^{(l)}W_i^{(l)}\}, \forall i \in (1, ..., b)))$$

(2) Predict edge, edge type and stop token

(3) Optimize using PPO

https://neurips.cc/media/Slides/nips/2018/220cd(05-15-30)-05-15-35-12656-Graph_Convoluti.pdf

# GCPN (NIPS 2018)

- Generating graphs from scratch:
  - Over 60% higher scores

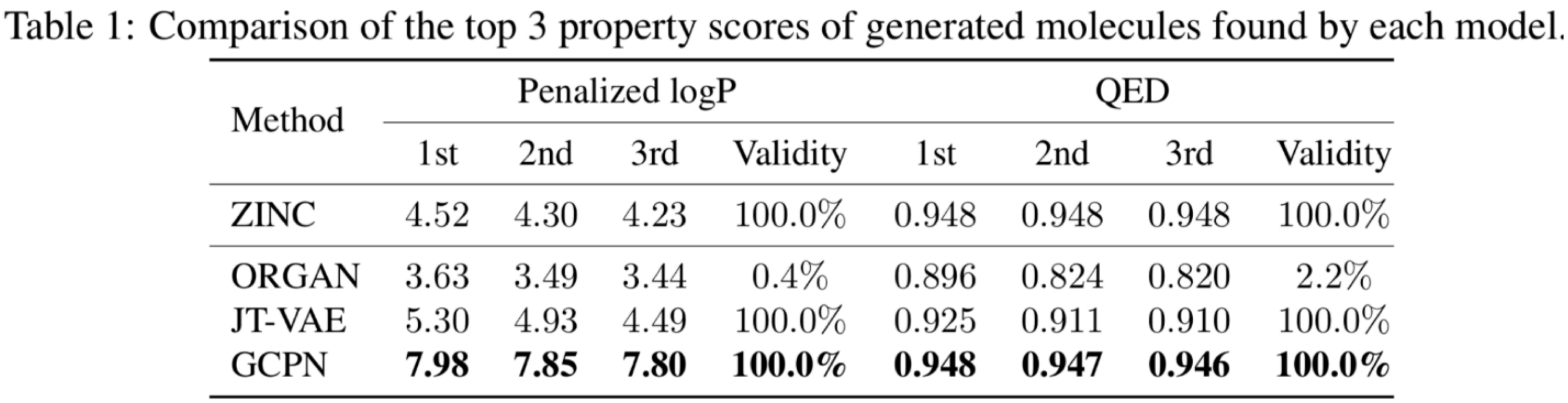


Table 1: Comparison of the top 3 property scores of generated molecules found by each model.

| Method | Penalized logP | | | | QED | | | |
|---|---|---|---|---|---|---|---|---|
| | 1st | 2nd | 3rd | Validity | 1st | 2nd | 3rd | Validity |
| ZINC | 4.52 | 4.30 | 4.23 | 100.0% | 0.948 | 0.948 | 0.948 | 100.0% |
| ORGAN | 3.63 | 3.49 | 3.44 | 0.4% | 0.896 | 0.824 | 0.820 | 2.2% |
| JT-VAE | 5.30 | 4.93 | 4.49 | 100.0% | 0.925 | 0.911 | 0.910 | 100.0% |
| GCPN | **7.98** | **7.85** | **7.80** | **100.0%** | **0.948** | **0.947** | **0.946** | **100.0%** |

- Modifying existing graphs:
  - Over 180% higher scores improvement

You et al., Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation, NIPS 2018