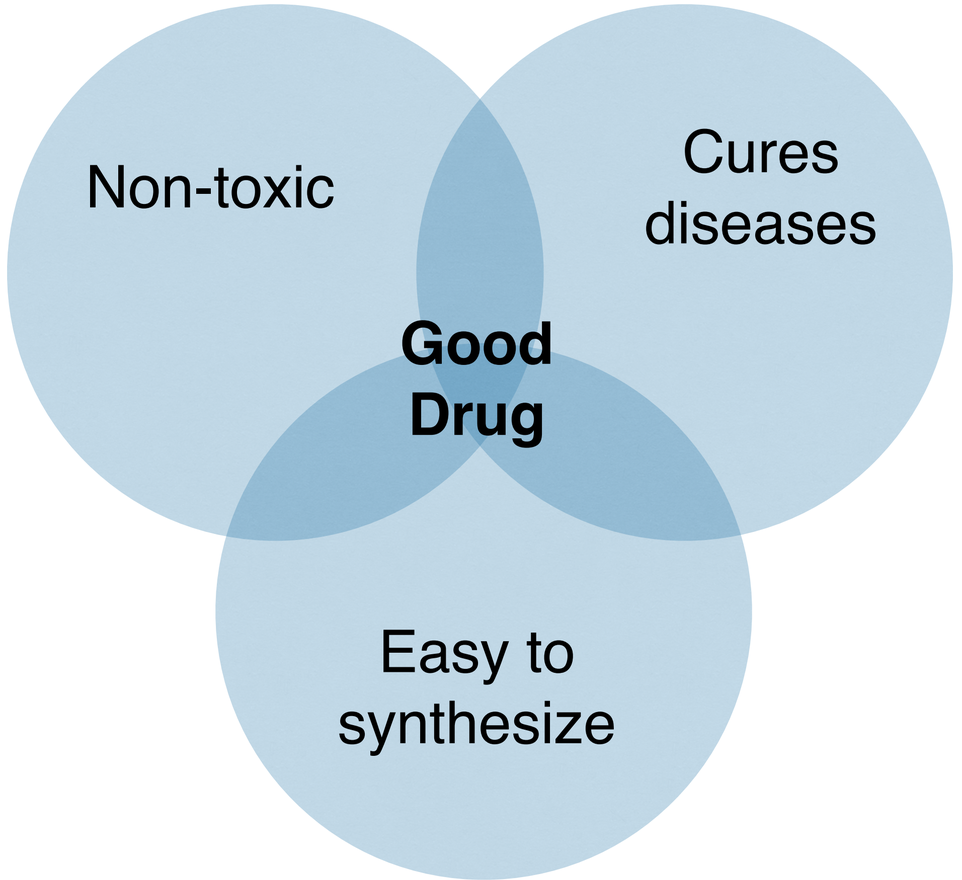
Multi-objective Molecule Generation using Interpretable Substructures

Wengong Jin, Regina Barzilay, Tommi Jaakkola
MIT CSAIL

Drug Discovery

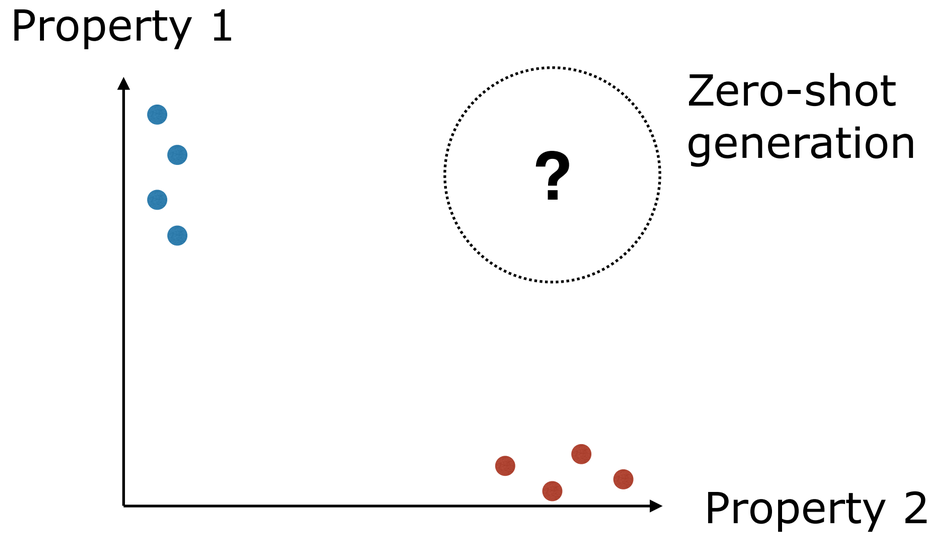
- Drug discovery: finding molecules with desired chemical properties
- A good drug needs to satisfy multiple objectives



Drug Discovery

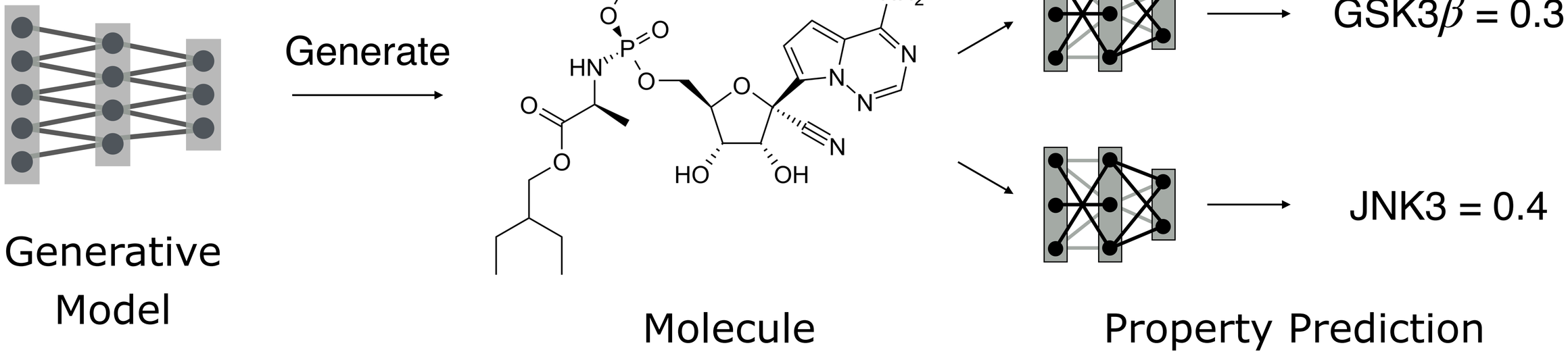
- Drug discovery: finding molecules with desired chemical properties
- A good drug needs to satisfy multiple objectives

- Multi-property optimization is challenging!
  - **Many** examples of molecules with a single property
  - **Few** instances of molecules that satisfy multiple property constraints
  - **Challenge:** How do we find compounds that satisfy all the criteria with few (or zero) examples of such molecules?



Formulation: Reinforcement Learning (RL)

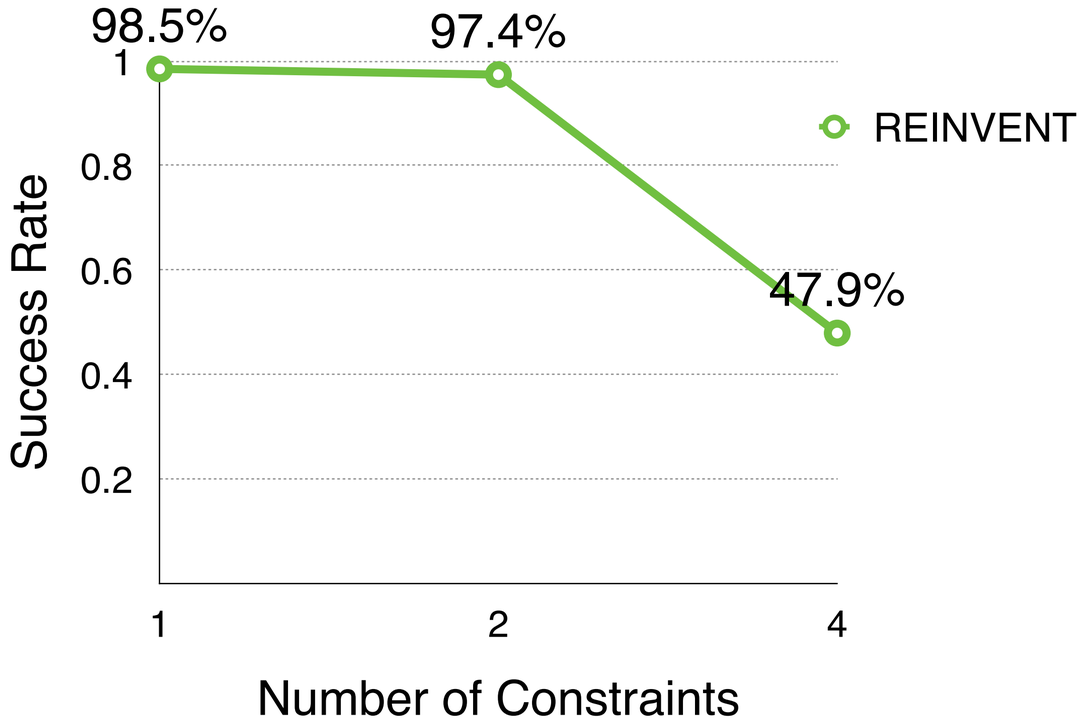
- De novo drug design using generative models
- The model learns to generate new drugs that satisfy all the property constraints
- Example: Design dual inhibitor (GSK3β + JNK3) to treat Alzheimers disease
- Maximize the reward using RL: reward(x) = GSK3β(x) + JNK3(x)



Challenge: Sparsity of Rewards

- De novo drug design using generative models
- The model learns to generate new drugs that satisfy all the property constraints
- Example: Design dual inhibitor (GSK3β + JNK3) to treat Alzheimers disease
- Maximize the reward using RL: reward(x) = GSK3β(x) + JNK3(x)

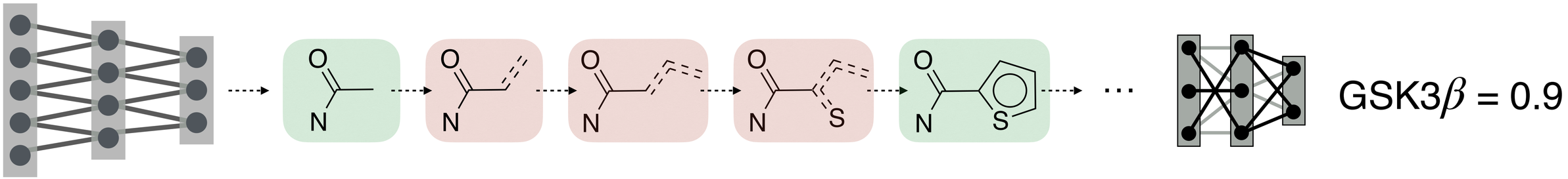
- Challenge: reward sparsity
  - We tested REINVENT (Olivecrona et al.), a state-of-the-art RL method for drug design
  - The more property constraints, the harder for RL to get positive rewards



Olivecrona et al., “Molecular de-novo design through deep reinforcement learning”, Journal of Cheminformatics (2017)

Challenge: Sparsity of Rewards

- Molecules are often generated via an autoregressive process:
  - In each step, the model adds one atom to the molecule
  - Reward are evaluated at the very end
  - Requires a lot of steps to complete a molecule!



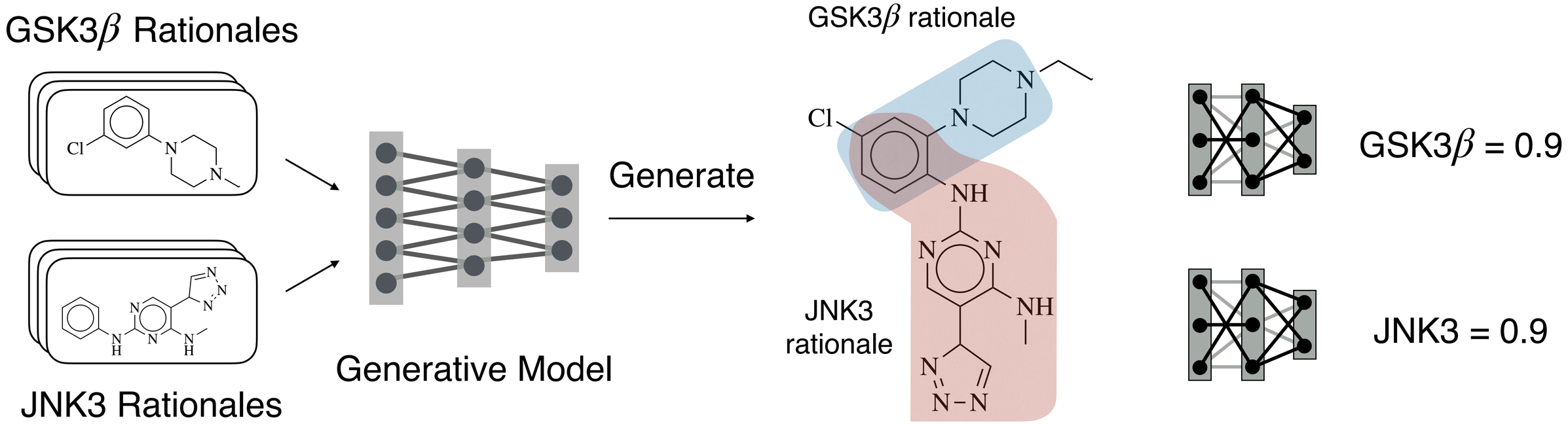
Generative Model

Autoregressive process (many generation steps)

Reward evaluated at the end.

Hierarchical Reinforcement Learning

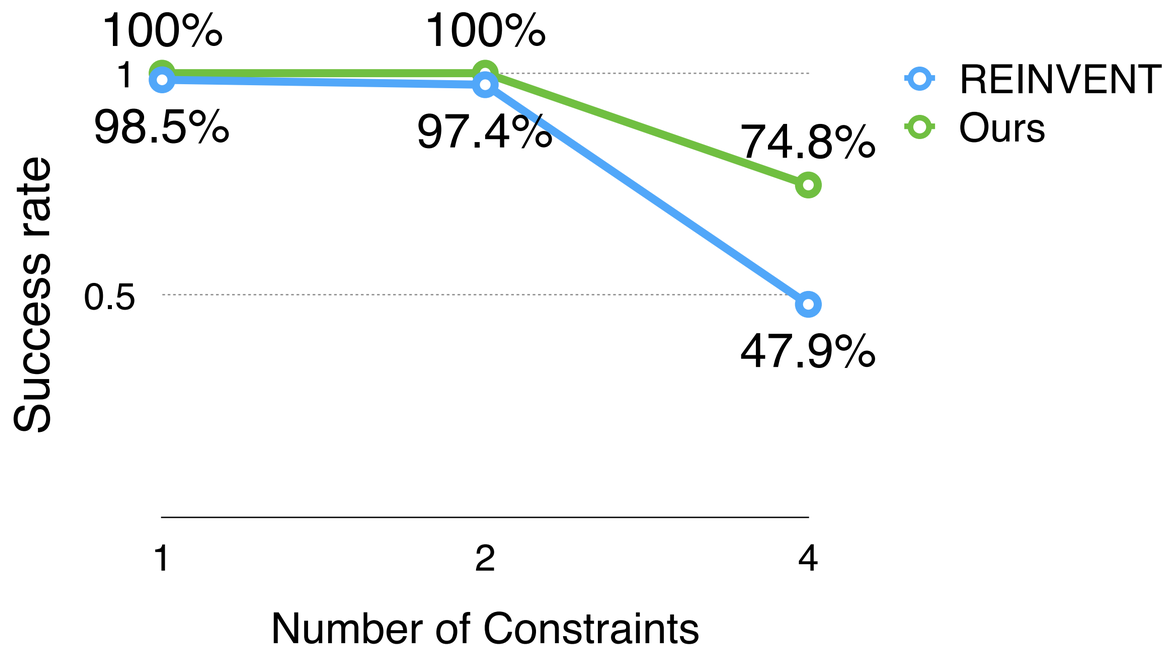
- Maximize the reward using RL: reward(x) = GSK3β(x) + JNK3(x)
- Learn property-specific **rationales** — subgraphs active to GSK3β or JNK3 individually.
- Rationales play similar roles to **options** in hierarchical RL (Sutton et al., 1999)



Sutton et al., Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2): 181–211, 1999.

Hierarchical Reinforcement Learning

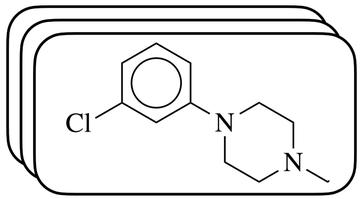
- Maximize the reward using RL: reward(x) = GSK3β(x) + JNK3(x)
- Learn property-specific **rationales** — subgraphs active to GSK3β or JNK3 individually.
- Rationales provide faster feedbacks and alleviate reward sparsity issue

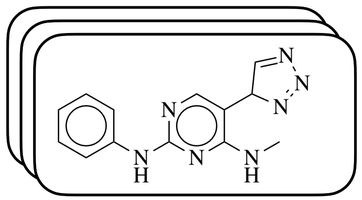


Sutton et al., Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2): 181–211, 1999.

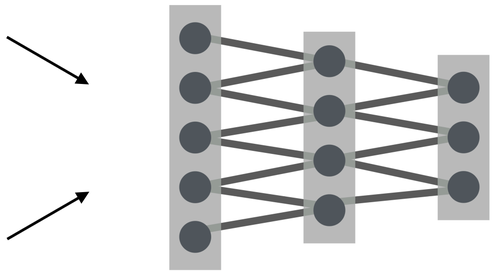
Model Components

GSK3β Rationales

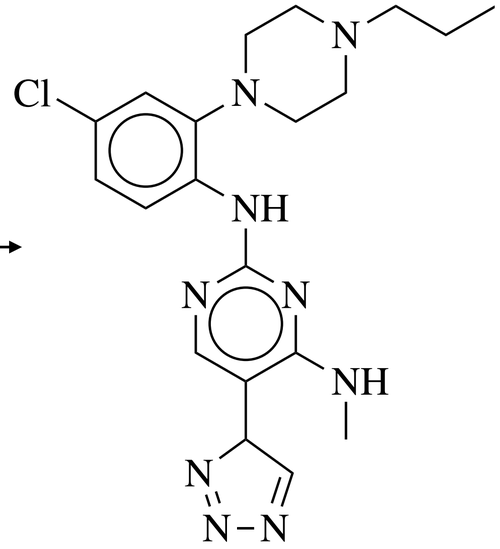


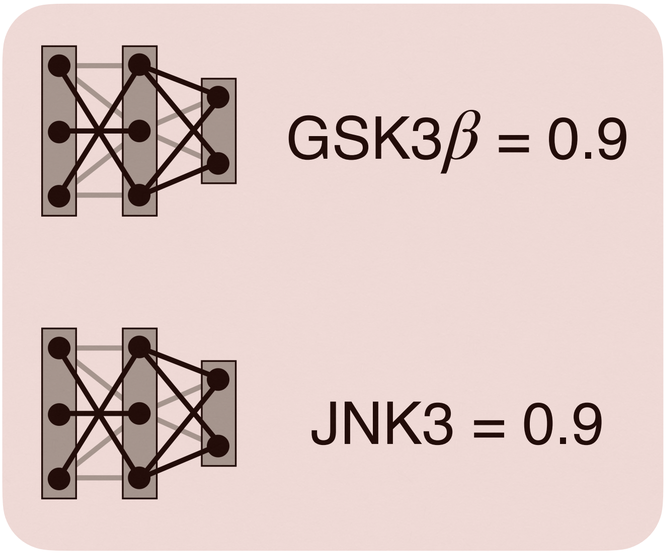


JNK3 Rationales



Generative Model





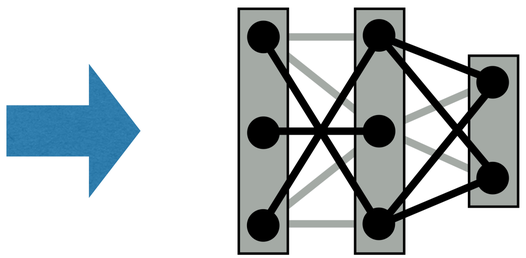
Property Predictors

Property Predictors

- To quickly evaluate the property of generated compounds, we train a property predictor over reference drugs with measured properties.
- This strategy is commonly adopted for drug de novo design (Olivecrona et al., 2017; Popova et al., 2018)
- The property predictor is fixed when training the generative model.

drug name	structure	active
COLISTIN SULFATE	CCC(C)CCCC(=O)NC(CCN	1
CHLORHEXIDINE DIHYDROCHLORIDE	Cl.Cl.NC(=NCCCCCCN=C(	0
GEMIFLOXACIN MESYLATE	CON=C1CN(c2nc3c(cc2	0
PYRITHIONE ZINC	[O-]n1ccccc1=S.[O-]n1c	0
CLEROCIDIN equilibrates in solution	CC1CCC2(C)C(C=O)=CCC	1
BENZETHONIUM CHLORIDE	CC(C)(C)CC(C)(C)c1ccc((	1
CEFPIRAMIDE	Cc1cc(=O)c(C(=O)NC(C(	0
SARAFLOXACIN HYDROCHLORIDE	Cl.O=C(O)c1cn(-c2ccc(F	0
GATIFLOXACIN	COc1c(N2CCNC(C)C2)c(	0

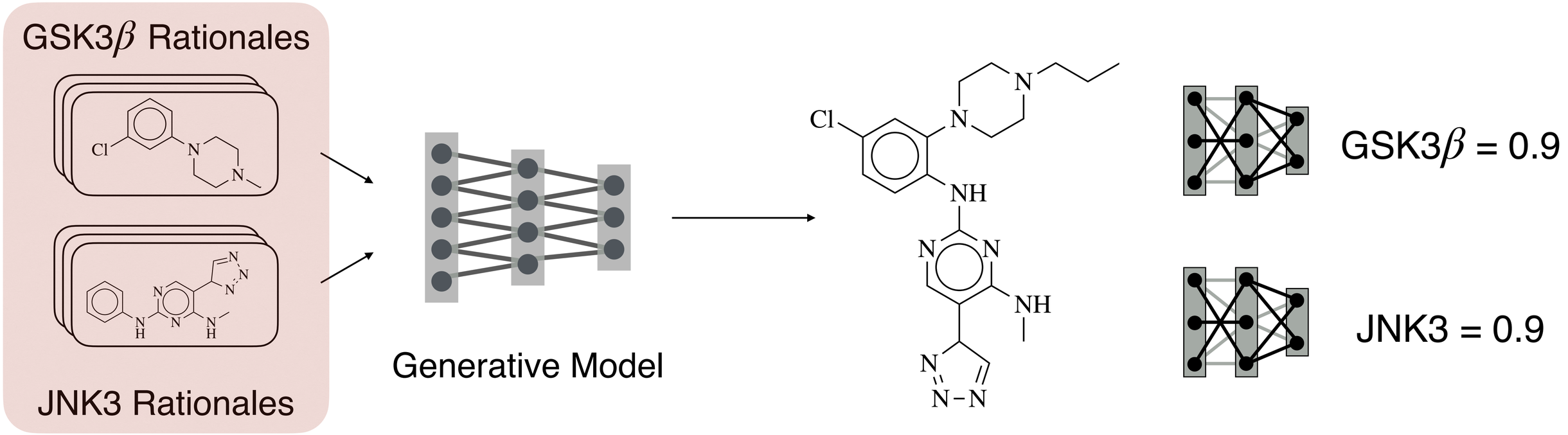
Reference drugs



Property predictor

Popova et al., "Deep reinforcement learning for de novo drug design." *Science advances* (2018)

Model Components



Rationale Extraction

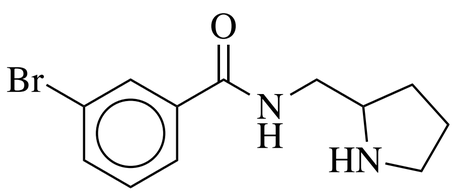
- In most cases, rationales are not provided to our models
- How to discover such rationales without direct supervision?

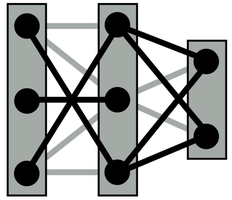
Rationale Extraction via Model Interpretation

- Our goal: given a molecule G, find a minimal subgraph S such that S retains desired property scores
- Extract rationales from active molecules in the training set

structure	active
Brc1cccc2C(=O)	1
Cl.Cl.NC(=NCCCC	0
CON=C1CN(c2n	0
[O-]n1ccccc1=S.	0
CC1CCC2(C)C(C=	1
CC(C)(C)CC(C)(C	1
Cc1cc(=O)c(C(=(	0
Cl.O=C(O)c1cn(-	0

Reference drugs





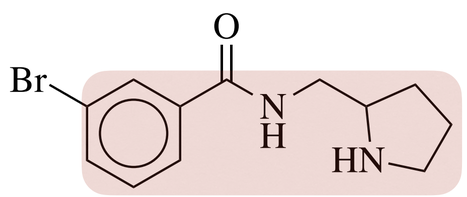
JNK3 = 0.9

Rationale Extraction via Model Interpretation

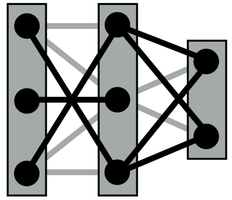
- Our goal: given a molecule G, find a minimal subgraph S such that S retains desired property scores
- Extract rationales from <u>active molecules</u> in the training set

structure	active
Brc1cccc2C(=O)	1
Cl.Cl.NC(=NCCCC	0
CON=C1CN(c2n	0
[O-]n1ccccc1=S.	0
CC1CCC2(C)C(C=	1
CC(C)(C)CC(C)(C	1
Cc1cc(=O)c(C(=(	0
Cl.O=C(O)c1cn(-	0

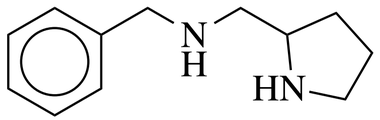
Reference drugs

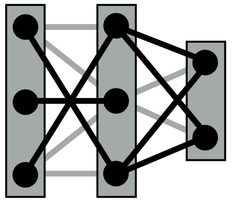


rationale

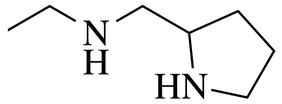


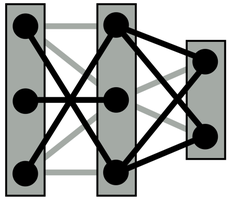
JNK3 = 0.9





JNK3 = 0.8

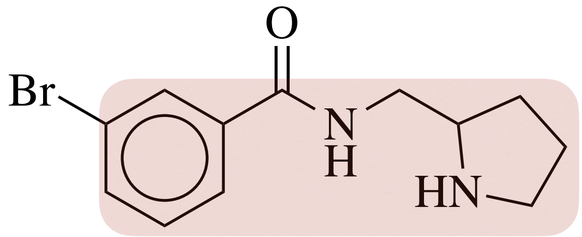


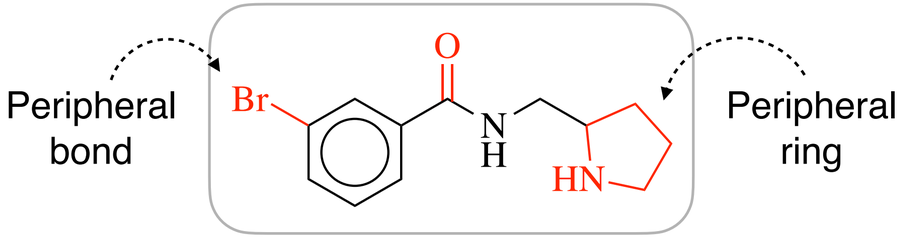


JNK3 = 0.1

Find Rationales by Monte Carlo Tree Search

- Our goal: given a molecule G, find a minimal subgraph S such that S retains desired property scores

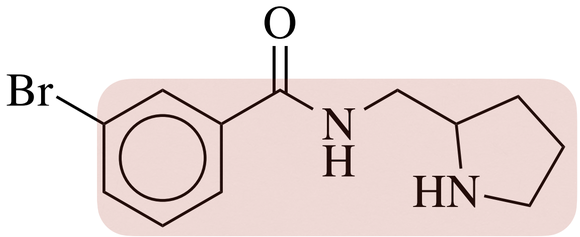




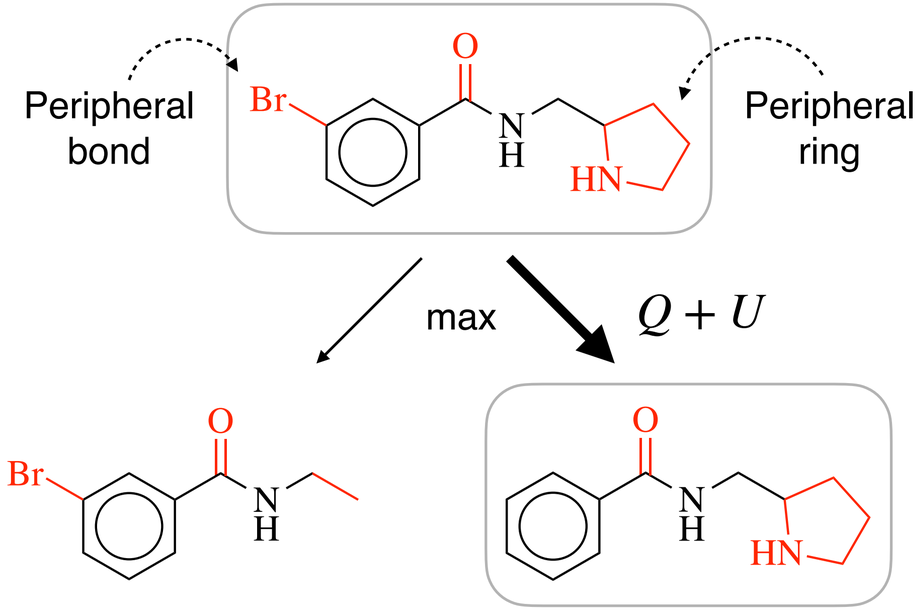
- How to solve this?
  - Iteratively remove peripheral bonds and rings to find subgraph S
  - Evaluate each subgraph using the (fixed) property predictor
  - Q and U functions are MCTS parameters that guides the search process
  - MCTS is much faster than exhaustive search.

Find Rationales by Monte Carlo Tree Search

- Our goal: given a molecule G, find a minimal subgraph S such that S retains desired property scores

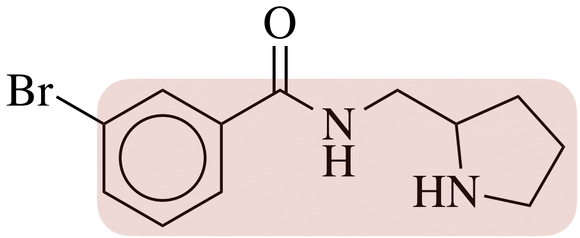


- How to solve this?
  - Iteratively remove peripheral bonds and rings to find subgraph S
  - Evaluate each subgraph using the (fixed) property predictor
  - Q and U functions are MCTS parameters that guides the search process
  - MCTS is much faster than exhaustive search.

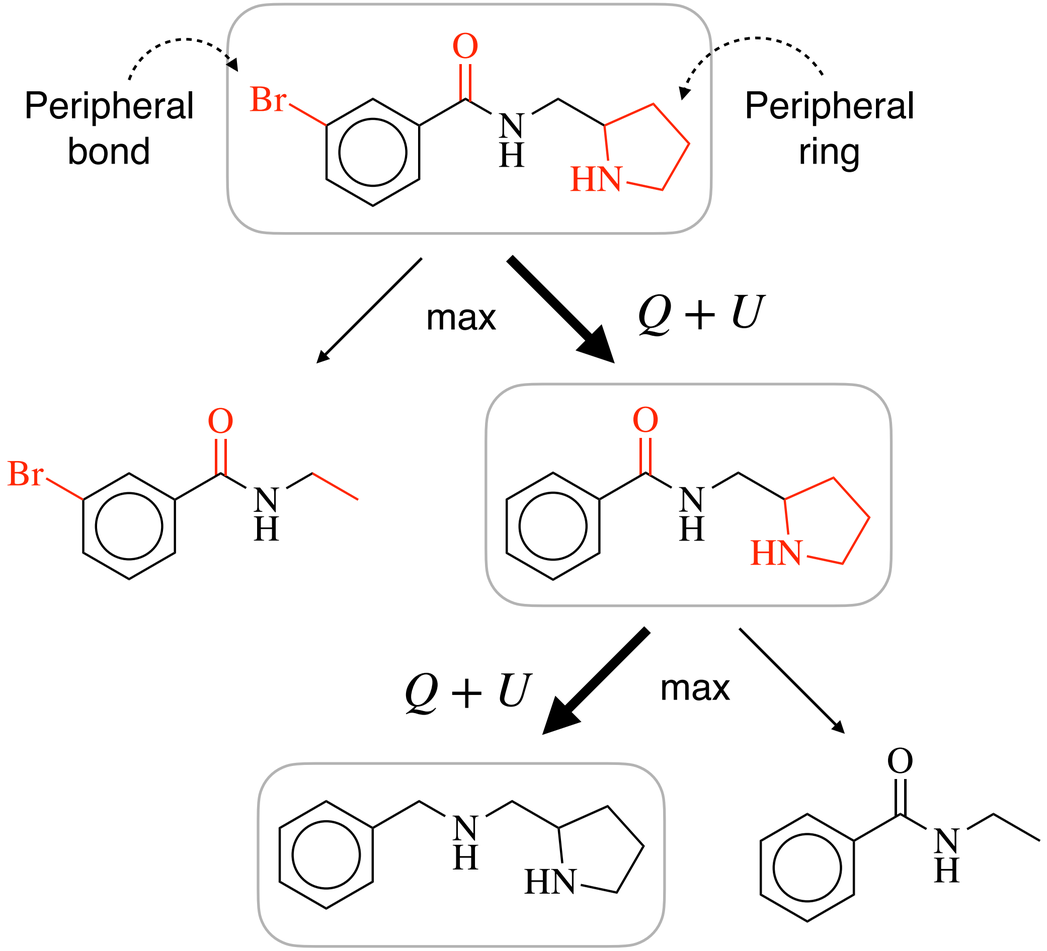


Find Rationales by Monte Carlo Tree Search

- Our goal: given a molecule G, find a minimal subgraph S such that S retains desired property scores

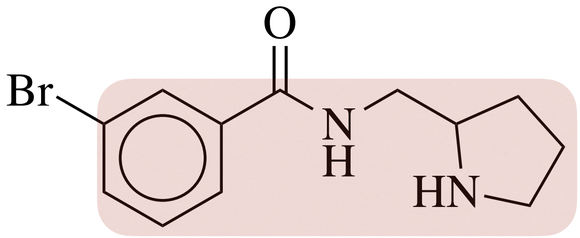


- How to solve this?
  - Iteratively remove peripheral bonds and rings to find subgraph S
  - Evaluate each subgraph using the (fixed) property predictor
  - Q and U functions are MCTS parameters that guides the search process
  - MCTS is much faster than exhaustive search.

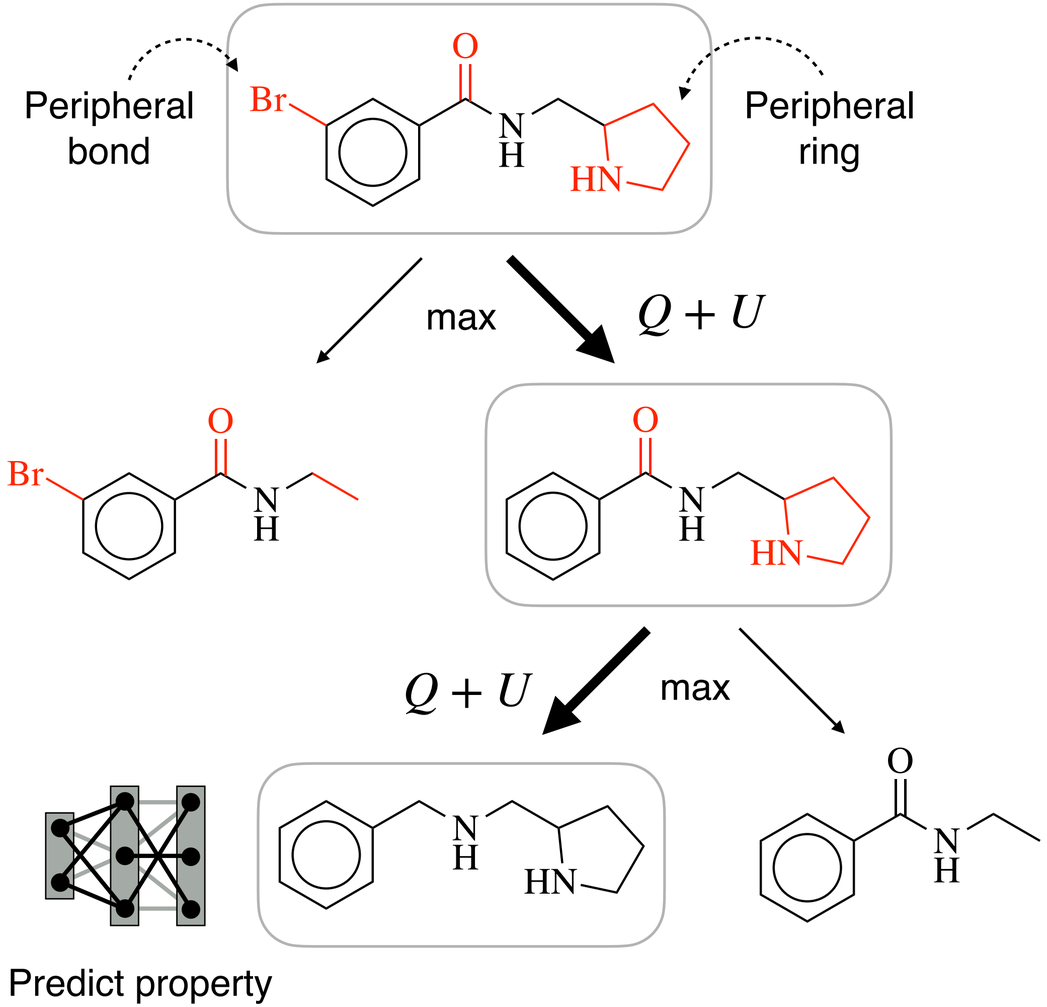


Find Rationales by Monte Carlo Tree Search

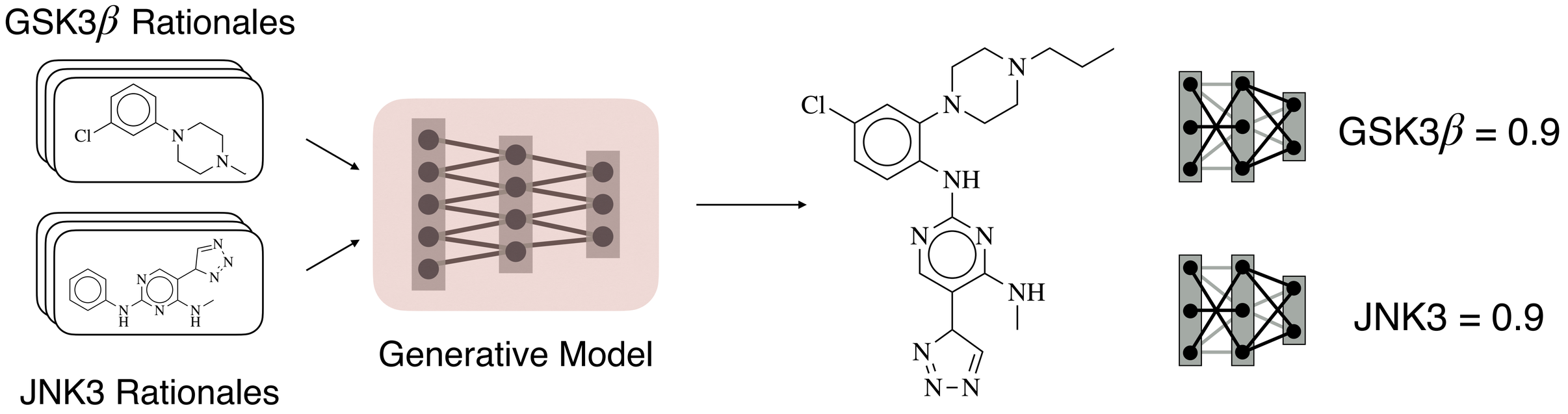
- Our goal: given a molecule G, find a minimal subgraph S such that S retains desired property scores



- How to solve this?
  - Iteratively remove peripheral bonds and rings to find subgraph S
  - Evaluate each subgraph using the (fixed) property predictor
  - Q and U functions are MCTS parameters that guides the search process
  - MCTS is much faster than exhaustive search.

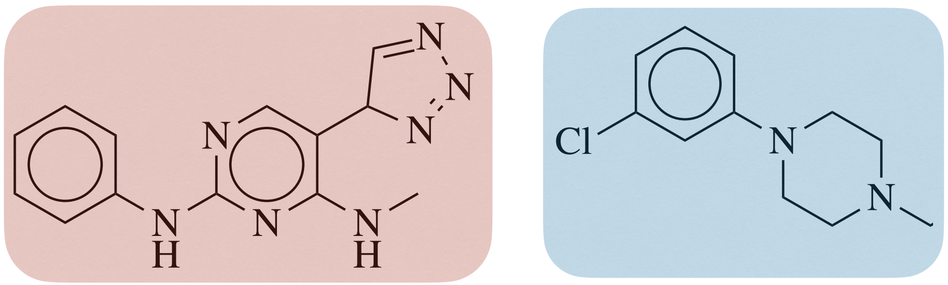


Model Components

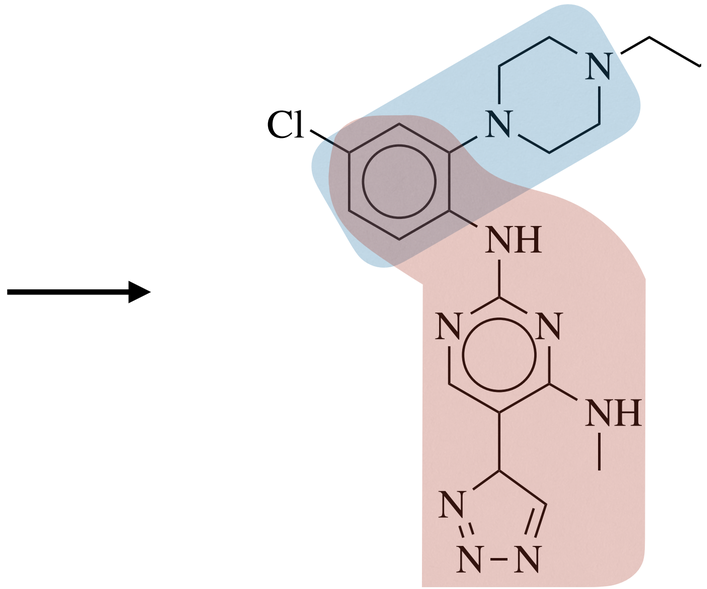


Molecule Completion

- Rationales are “partial” molecules
- We need to complete them into a full molecule
  - Rationales from different properties are disconnected.
- Learn a molecule completion model $P(G|S)$ to connect the rationales.



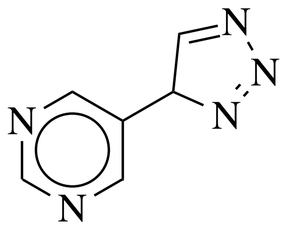
Rationale S

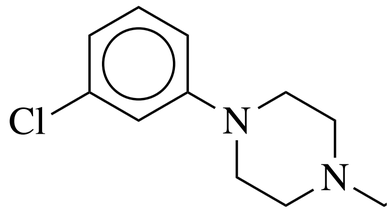


Expanded Molecule
$G \sim P(G|S)$

Molecule Completion

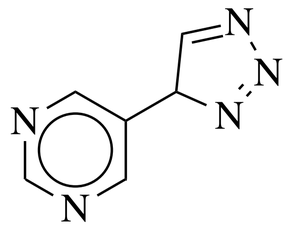
- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds



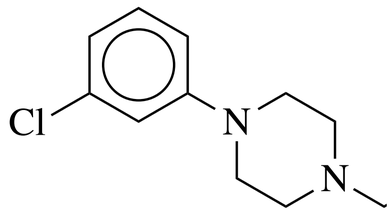


Molecule Completion

- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds

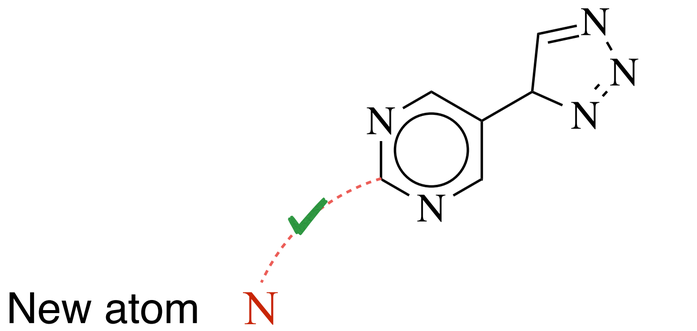


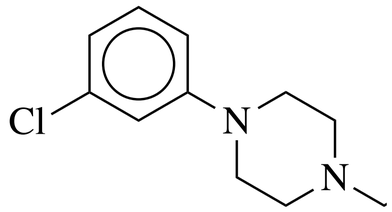
New atom N



Molecule Completion

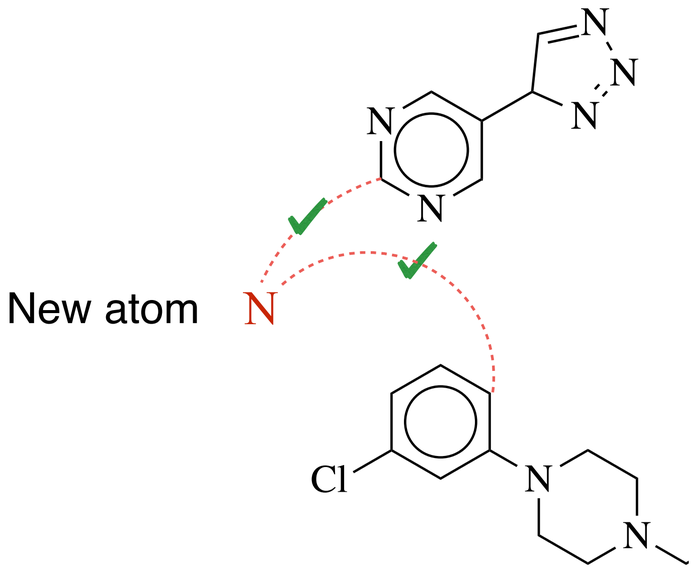
- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds





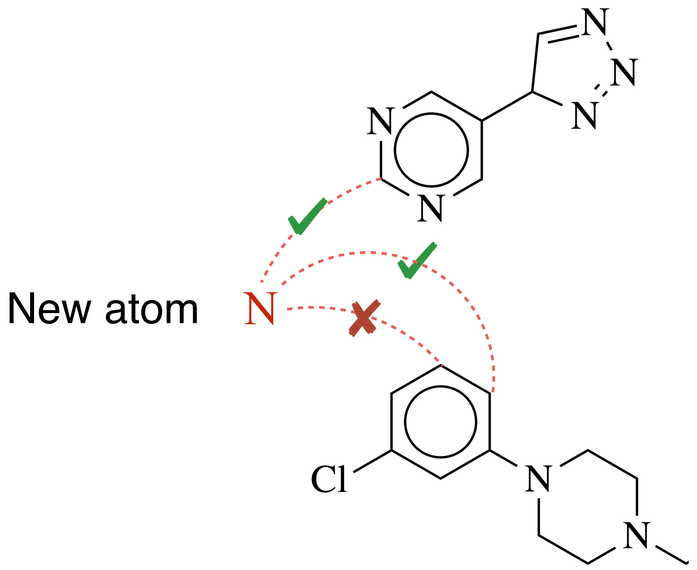
Molecule Completion

- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds



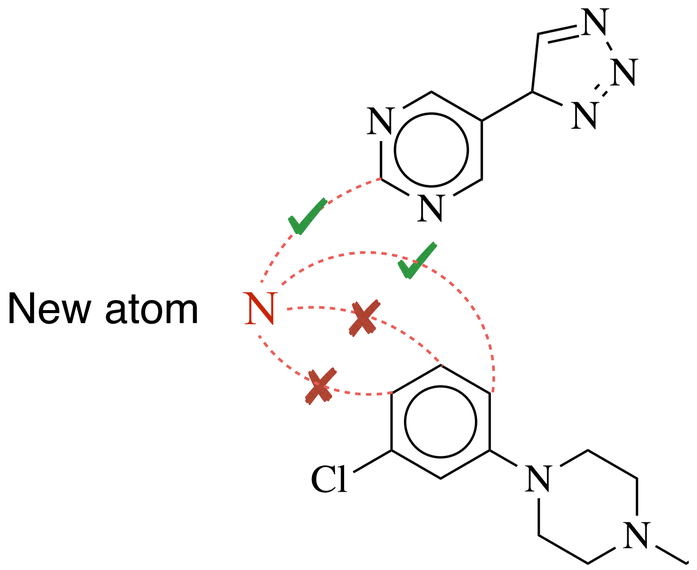
Molecule Completion

- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds



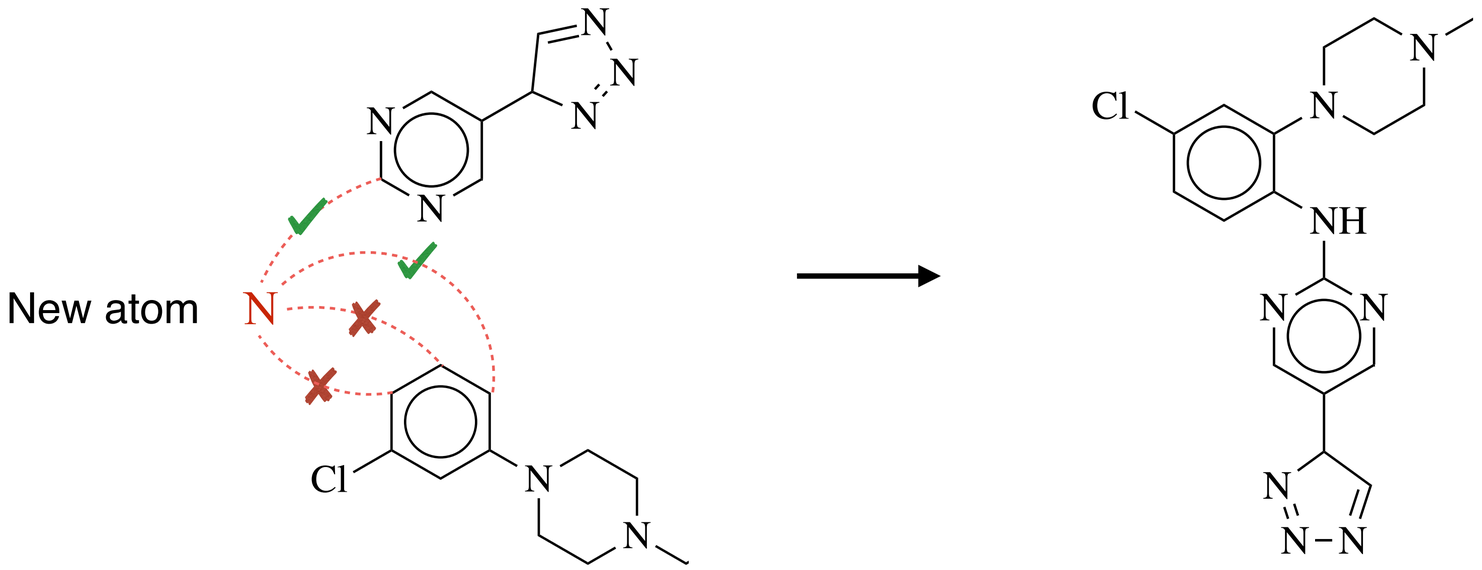
Molecule Completion

- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds



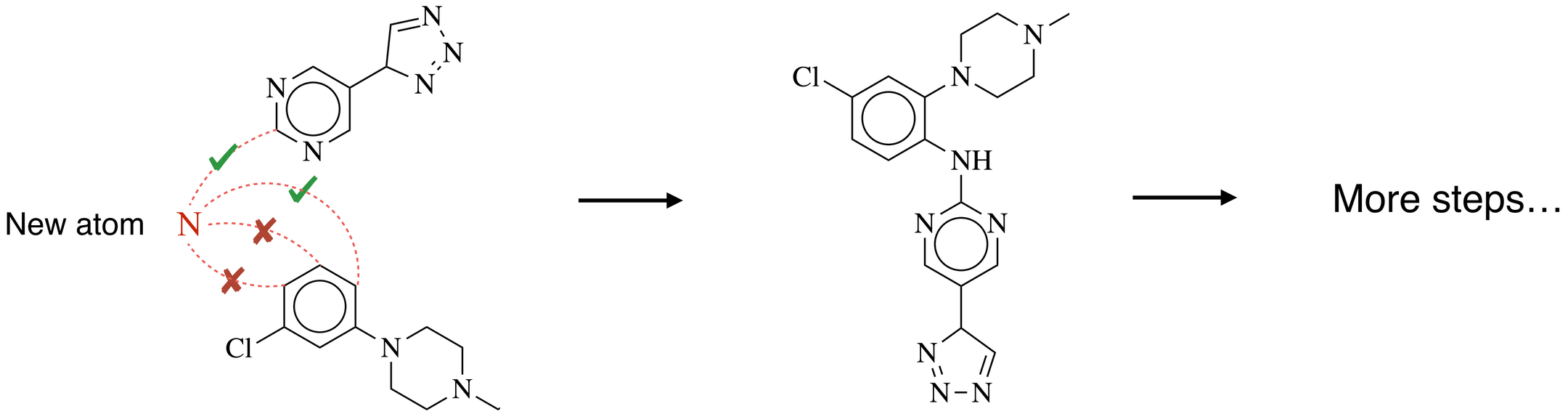
Molecule Completion

- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds



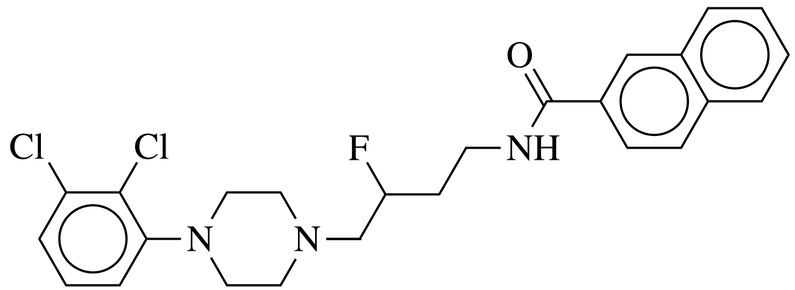
Molecule Completion

- We model $P(G|S)$ as an autoregressive process
- For simplicity, we use a simple atom-by-atom molecule completion model
  - More advanced architectures are certainly beneficial
- In each step, we add an atom to the current molecule, and predict its associated bonds



Pre-training Molecule Completion

- Molecule completion model can be trained without “property” predictors
- Pre-train molecule completion on a large set of unlabeled molecules (e.g., ChEMBL)

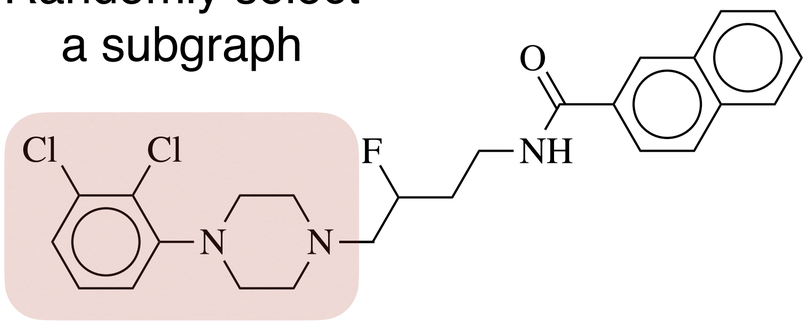


Molecule from ChEMBL

Pre-training Molecule Completion

- Molecule completion model can be trained without “property” predictors
- Pre-train molecule completion on a large set of unlabeled molecules (e.g., ChEMBL)

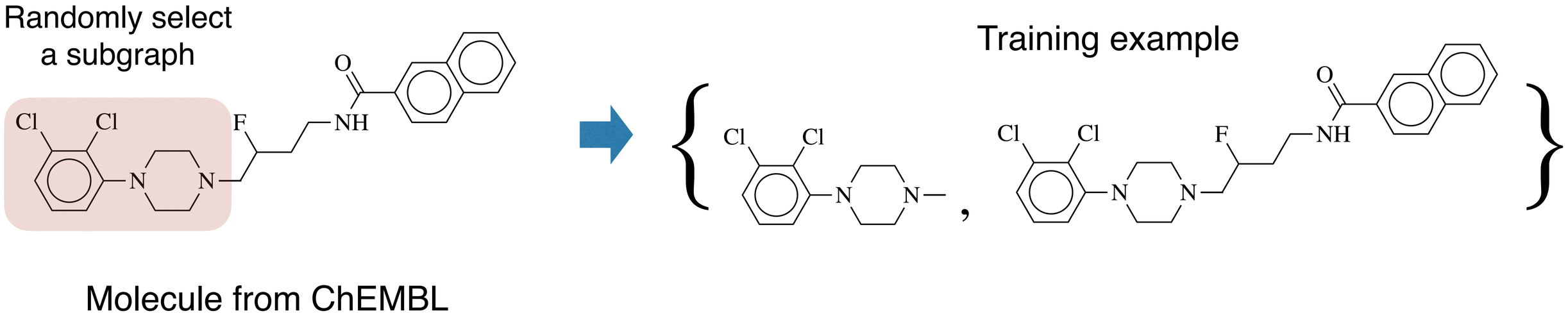
Randomly select
a subgraph



Molecule from ChEMBL

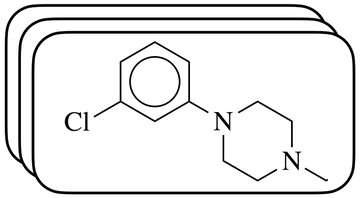
Pre-training Molecule Completion

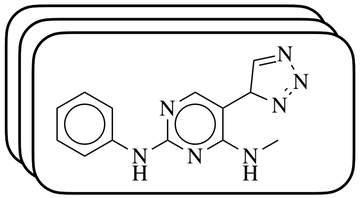
- Molecule completion model can be trained without “property” predictors
- Pre-train molecule completion on a large set of unlabeled molecules (e.g., ChEMBL)



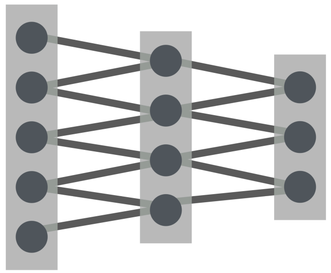
Putting everything together

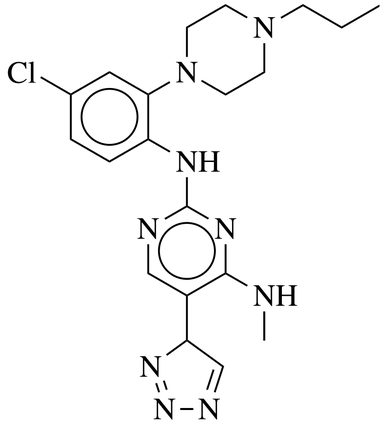
Rationales



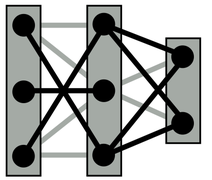


Generative Model

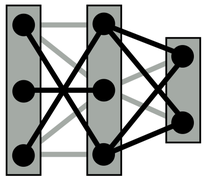




Property Predictor



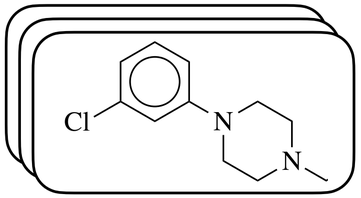
GSK3β = 0.9

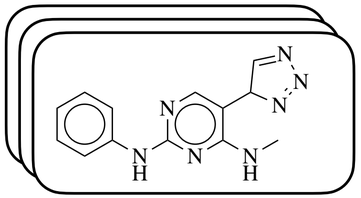


JNK3 = 0.9

Putting everything together

Rationales

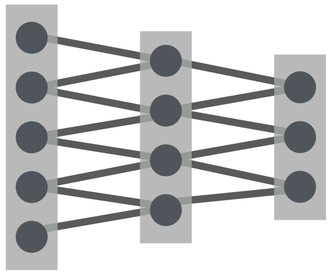


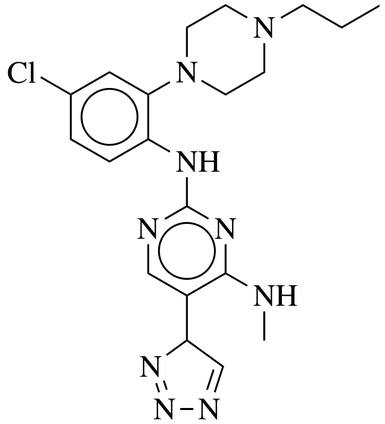


$P(S)$

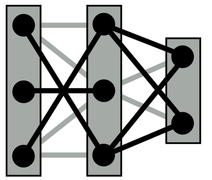
Rationale distribution

Generative Model

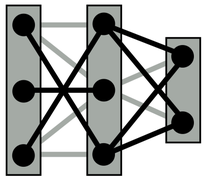




Property Predictor



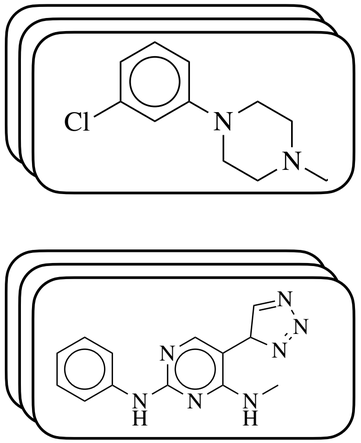
GSK3β = 0.9



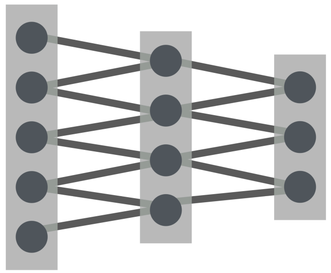
JNK3 = 0.9

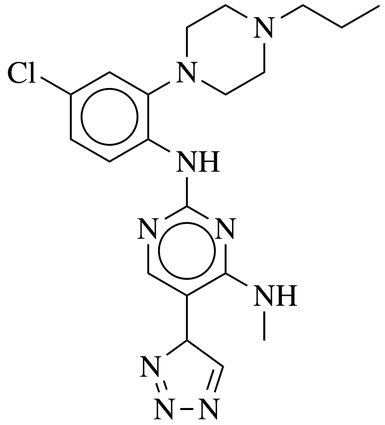
Putting everything together

Rationales

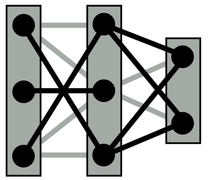


Generative Model

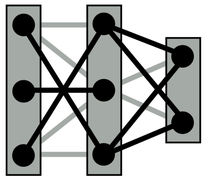




Property Predictor



GSK3β = 0.9



JNK3 = 0.9

$P(S)$

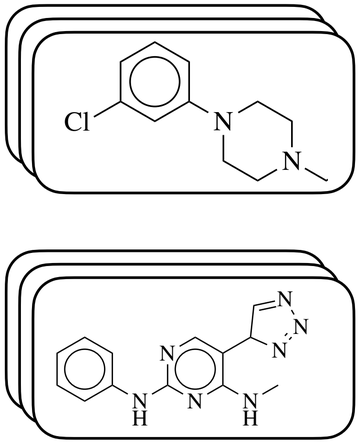
Rationale distribution

$P(G \mid S)$

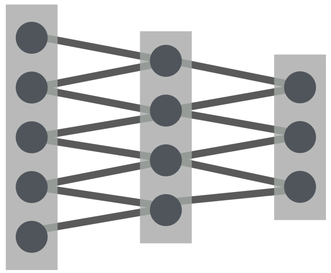
Molecule completion

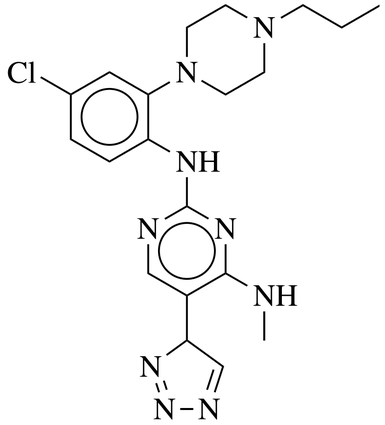
Putting everything together

Rationales

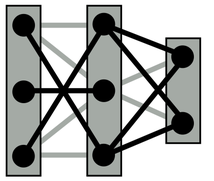


Generative Model

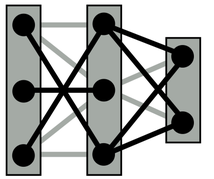




Property Predictor



GSK3β = 0.9



JNK3 = 0.9

$P(S)$

Rationale distribution

$P(G \mid S)$

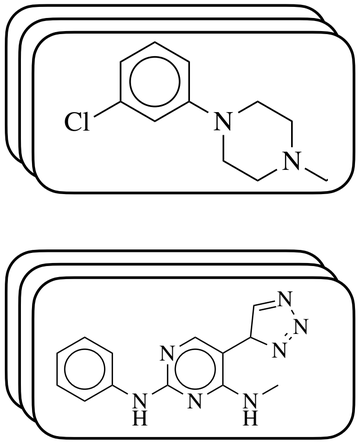
Molecule completion

$$P(G) = \sum_S P(S)P(G \mid S)$$

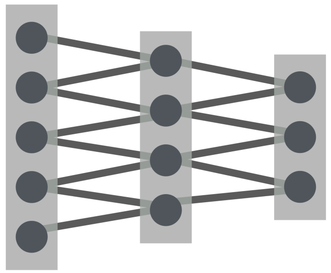
Molecule distribution

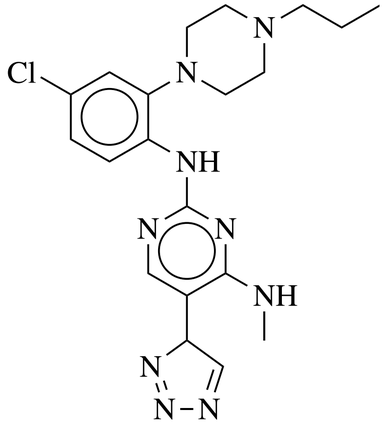
Putting everything together

Rationales

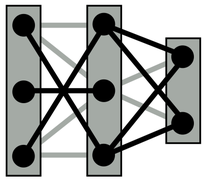


Generative Model

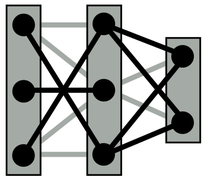




Property Predictor



GSK3β = 0.9



JNK3 = 0.9

$P(S)$

Rationale distribution

$P(G \mid S)$

Molecule completion

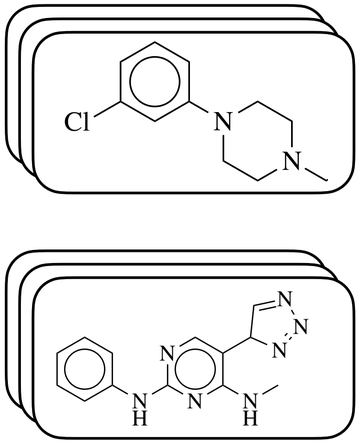
$$P(G) = \sum_S P(S)P(G \mid S)$$

Molecule distribution

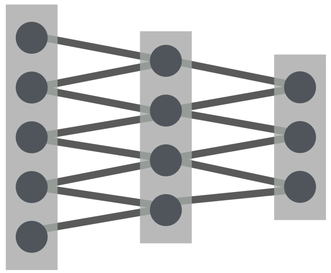
Maximize expected reward: $R = \sum_G R(G)P(G) + \lambda \mathbb{H}[P(S)]$

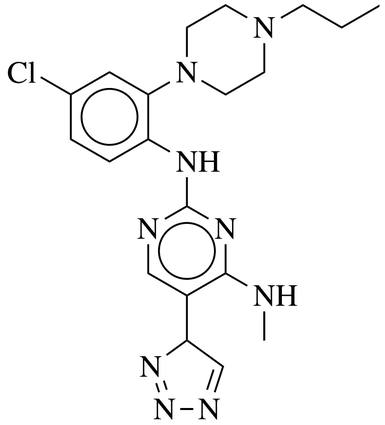
Putting everything together

Rationales

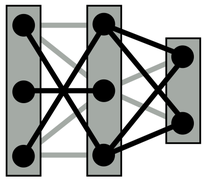


Generative Model

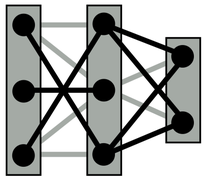




Property Predictor



GSK3β = 0.9



JNK3 = 0.9

$P(S)$

Rationale distribution

$P(G \mid S)$

Molecule completion

$$P(G) = \sum_S P(S)P(G \mid S)$$

Molecule distribution

Maximize expected reward: $R = \sum_G R(G)P(G) + \lambda \mathbb{H}[P(S)]$

Entropy regularization
(explore diverse set of rationales)

Experiments

- Three evaluation metrics
  - we do not explicitly train the model to optimize these metrics, except success rate

Experiments

- Three evaluation metrics
  - we do not explicitly train the model to optimize these metrics, except success rate

- Success rate:
  - How often do generated molecules satisfy all the property constraints?
  - Following Olivecrona et al., we use property predictors to compute this metric

Experiments

- Three evaluation metrics
  - we do not explicitly train the model to optimize these metrics, except success rate

- Success rate:
  - How often do generated molecules satisfy all the property constraints?
  - Following Olivecrona et al., we use property predictors to compute this metric

- Diversity:
  - Average pairwise molecular distance: $\sum_{X,Y} \text{dist}(X, Y)$

Experiments

- Three evaluation metrics
  - we do not explicitly train the model to optimize these metrics, except success rate

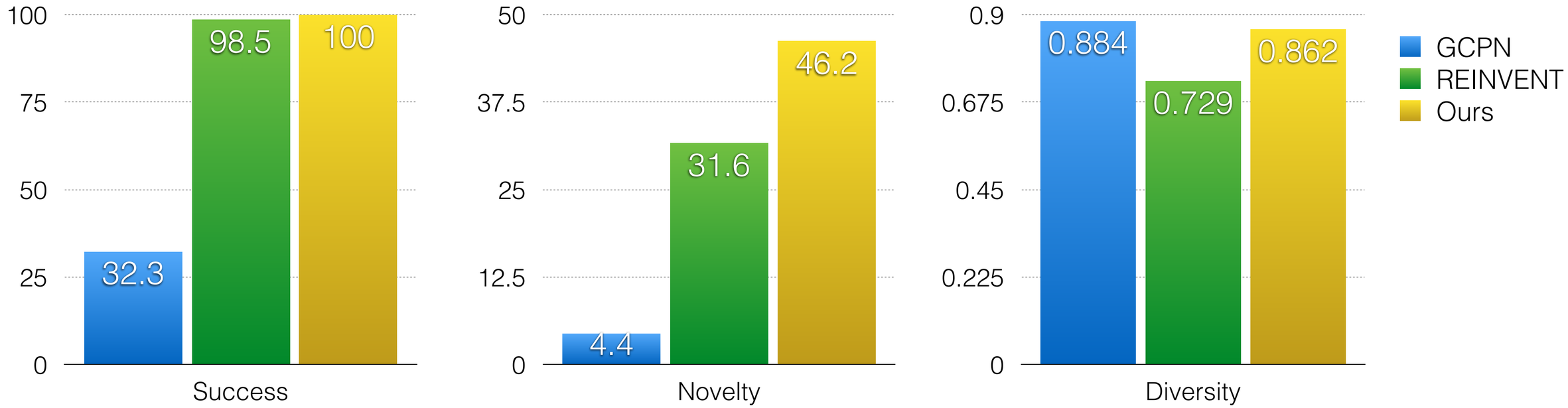
- Success rate:
  - How often do generated molecules satisfy all the property constraints?
  - Following Olivecrona et al., we use property predictors to compute this metric

- Diversity:
  - Average pairwise molecular distance: $\sum_{X,Y} \text{dist}(X, Y)$

- Novelty:
  - We don't want to rediscover existing drugs known to satisfy all the constraints.
  - A molecule G is novel if $\text{dist}(G, G_{NN}) > 0.6$, where G_{NN} is its nearest neighbor in the training set (i.e., not similar to any of the drugs)

Single Constraint: JNK3 Inhibitor Design

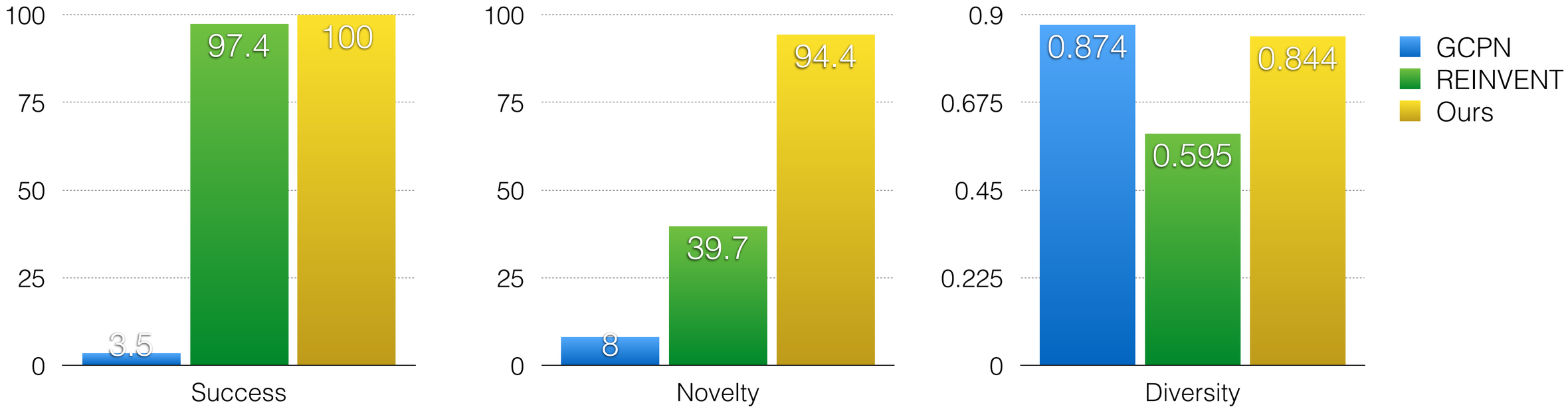
- We compare with two state-of-the-art RL methods
  - GCPN (You et al., 2018)
  - REINVENT (Olivecrona et al., 2017)
- Our model achieves the best success rate and novelty score



You et al., Graph convolutional policy network for goal-directed molecular graph generation. NeurIPS 2018

Two Constraints: GSK3/JNK3 Dual Inhibitor

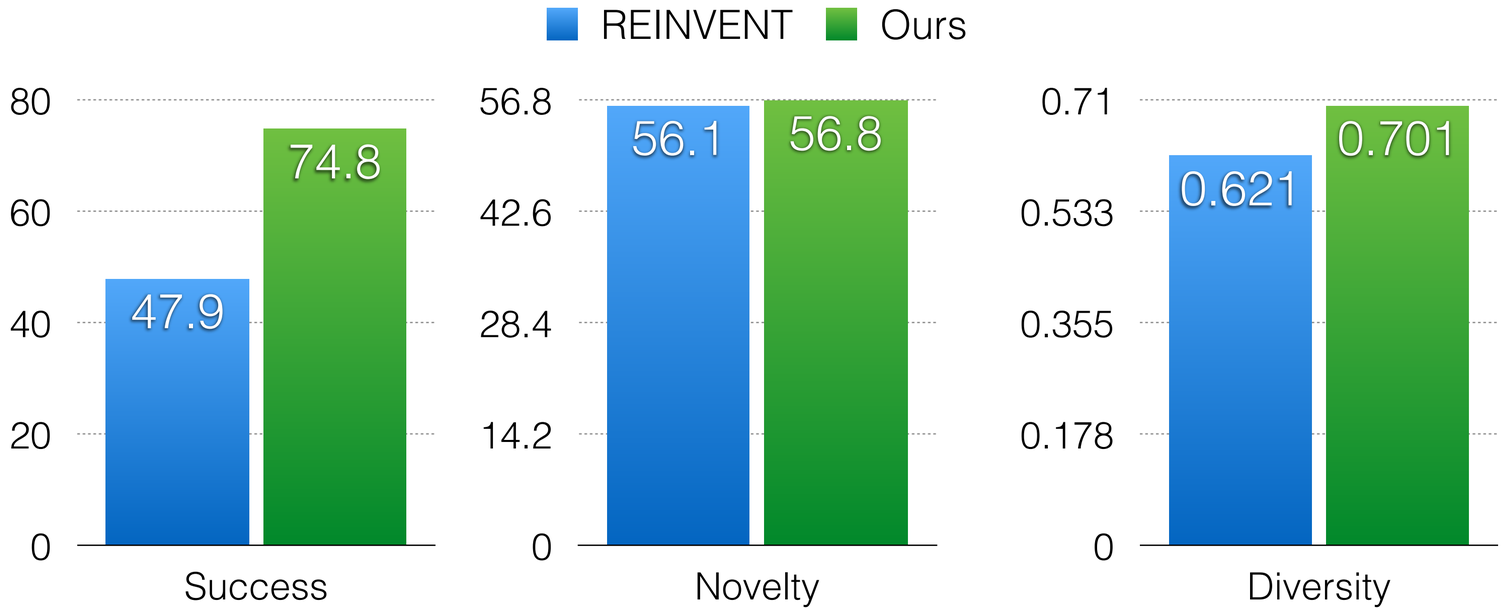
- Jointly inhibiting JNK3 and GSK3β can be beneficial for treating Alzheimers disease [1]
- Property predictors are trained over the dataset from Li et al., 2018 [1]
- Our model achieves the best result across all the three metrics.

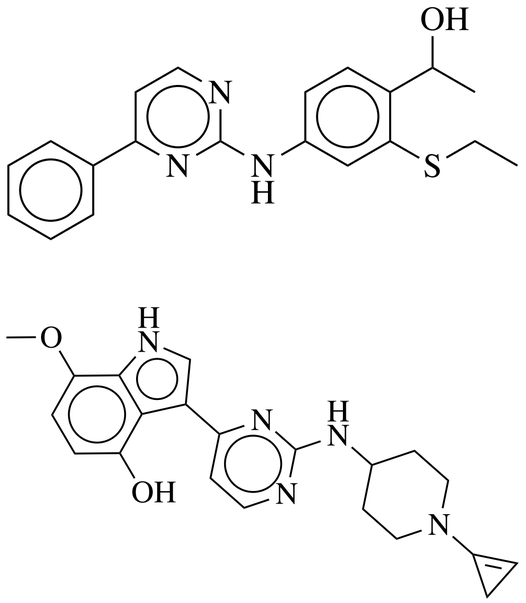


[1] Li et al., Multi-objective de novo drug design with conditional graph generative model. *Journal of Cheminformatics*, 2018.

Four Constraints: GSK3 + JNK3 + QED + SA

- Jointly inhibiting JNK3 and GSK3β can be beneficial for treating Alzheimers
- We further require generated dual inhibitors to be drug like (QED > 0.6) and synthetically accessible (SA < 4.0)
- Our model significantly outperforms REINVENT (esp. success rate)





Generated compounds

Are the rationales chemically meaningful?

Are the rationales chemically meaningful?

- The extracted rationales should accurately explain the property of interest

Are the rationales chemically meaningful?

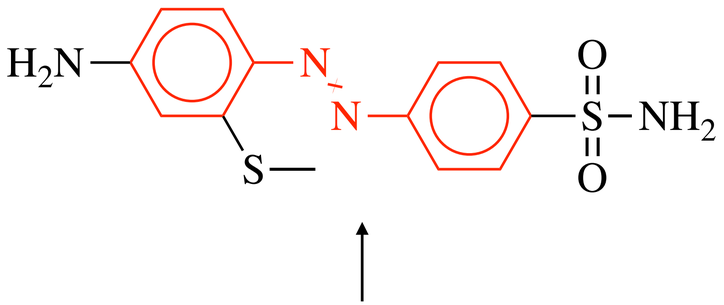
- The extracted rationales should accurately explain the property of interest
- As there is no ground truth rationales for GSK3, JNK3, we construct a toxicity dataset to evaluate rationale extraction module.

Are the rationales chemically meaningful?

- The extracted rationales should accurately explain the property of interest
- As there is no ground truth rationales for GSK3, JNK3, we construct a toxicity dataset to evaluate rationale extraction module.
- We construct a toxicity dataset where a molecule is labeled as toxic if it contains structural alerts (Sushko et al., 2012)

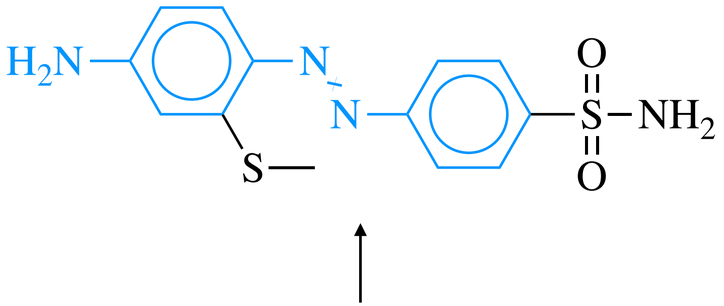
Are the rationales chemically meaningful?

- The extracted rationales should accurately explain the property of interest
- As there is no ground truth rationales for GSK3, JNK3, we construct a toxicity dataset to evaluate rationale extraction module.
- We construct a toxicity dataset where a molecule is labeled as toxic if it contains structural alerts (Sushko et al., 2012)



Structural alert

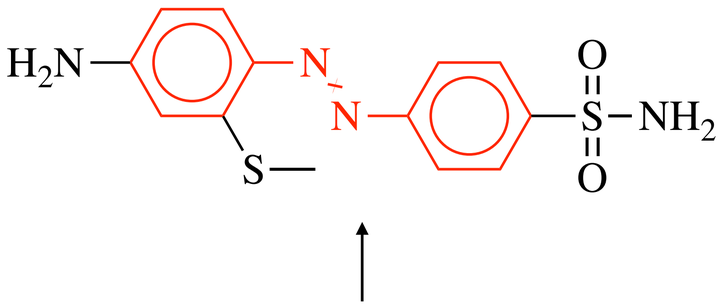
Toxic compound



Extracted rationale

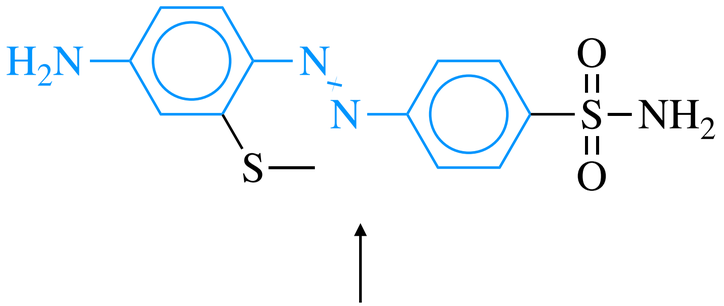
Are the rationales chemically meaningful?

- The extracted rationales should accurately explain the property of interest
- As there is no ground truth rationales for GSK3, JNK3, we construct a toxicity dataset to evaluate rationale extraction module.
- We construct a toxicity dataset where a molecule is labeled as toxic if it contains structural alerts (Sushko et al., 2012)
- We train a graph convolutional network and use MCTS to extract rationales



Structural alert

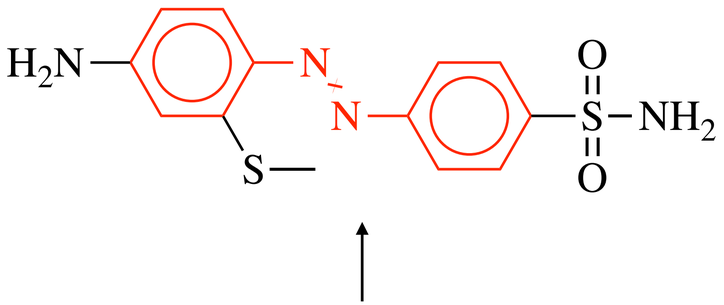
Toxic compound



Extracted rationale

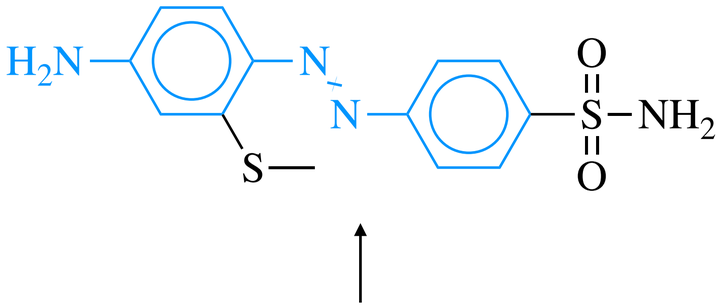
Are the rationales chemically meaningful?

- The extracted rationales should accurately explain the property of interest
- As there is no ground truth rationales for GSK3, JNK3, we construct a toxicity dataset to evaluate rationale extraction module.
- We construct a toxicity dataset where a molecule is labeled as toxic if it contains structural alerts (Sushko et al., 2012)
- We train a graph convolutional network and use MCTS to extract rationales
- The extracted rationales should match the structural alerts



Structural alert

Toxic compound



Extracted rationale

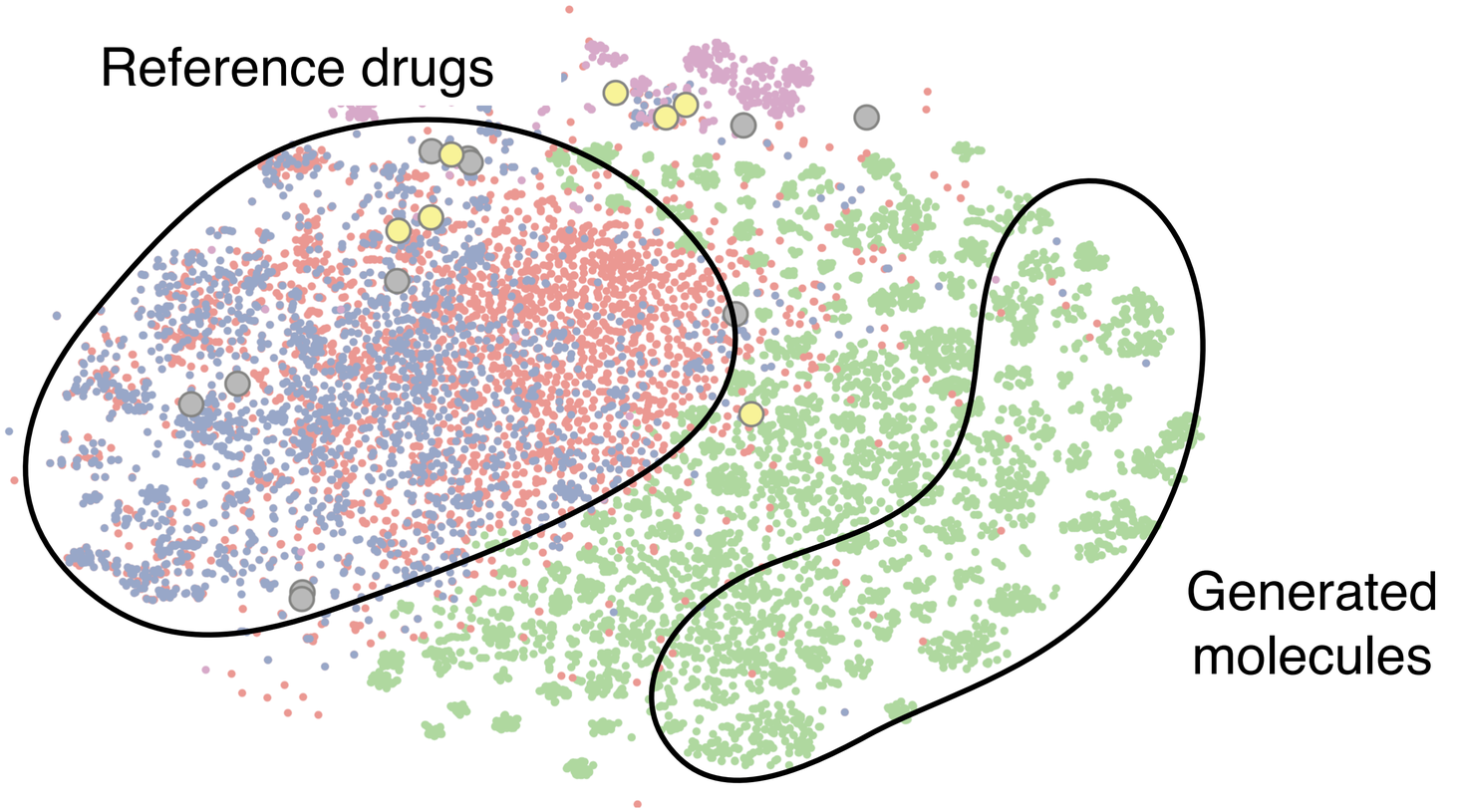
Are the rationales chemically meaningful?

- The extracted rationales should accurately explain the property of interest
- As there is no ground truth rationales for GSK3, JNK3, we construct a toxicity dataset to evaluate rationale extraction module.
- We construct a toxicity dataset where a molecule is labeled as toxic if it contains structural alerts (Sushko et al., 2012)
- We train a graph convolutional network and use MCTS to extract rationales
- The extracted rationales should match the structural alerts

Method	Partial Match	Exact Match
Integrated Gradient	0.857	39.4%
MCTS Rationale	**0.861**	**46.0%**

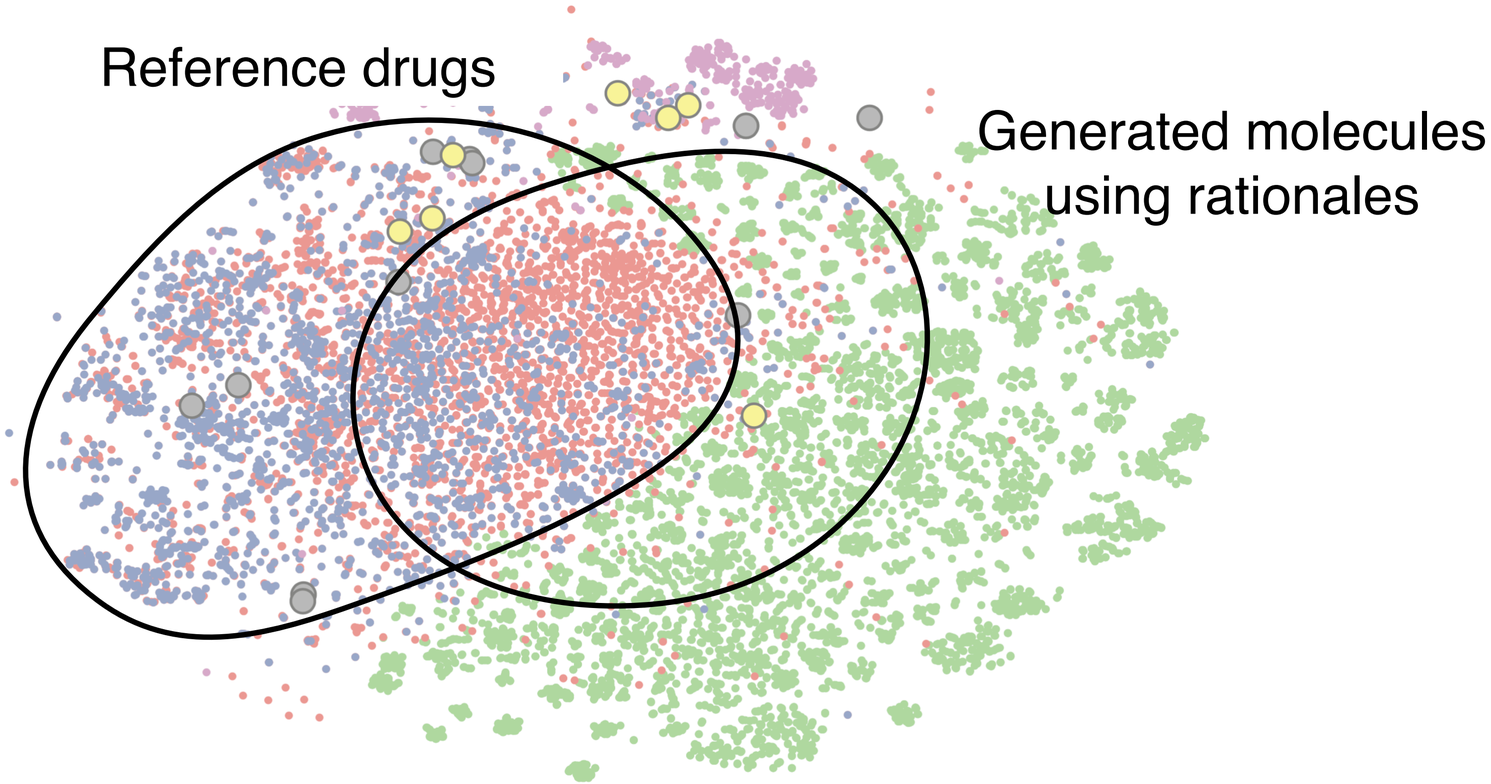
Are the property predictors reliable?

- We use a property predictor to evaluate the generated compounds.
- However, generated compounds can be far away from the drugs used to train the property predictor — predicted properties may not be reliable!



Rationale encourage reliability

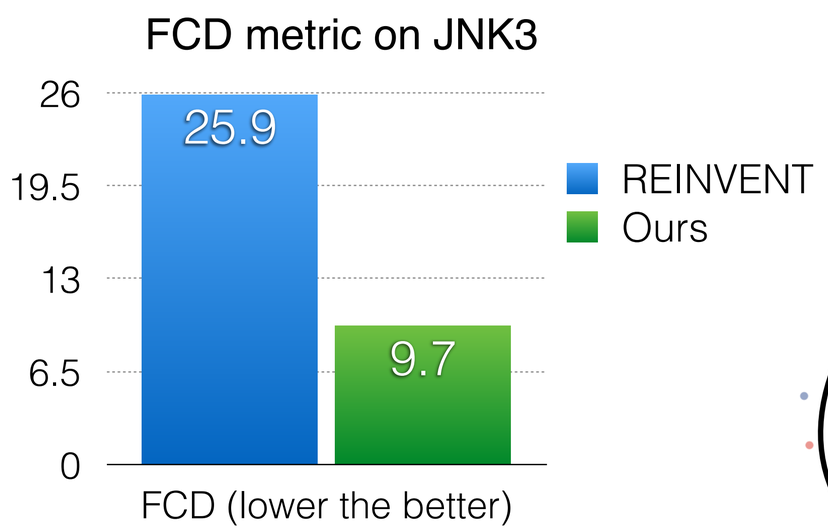
- We use a property predictor to evaluate the property of generated compounds
- However, generated compounds can be far away from the drugs used to train the property predictor — that's why rationales are useful!
- Molecules generated from rationales are closer to reference drugs



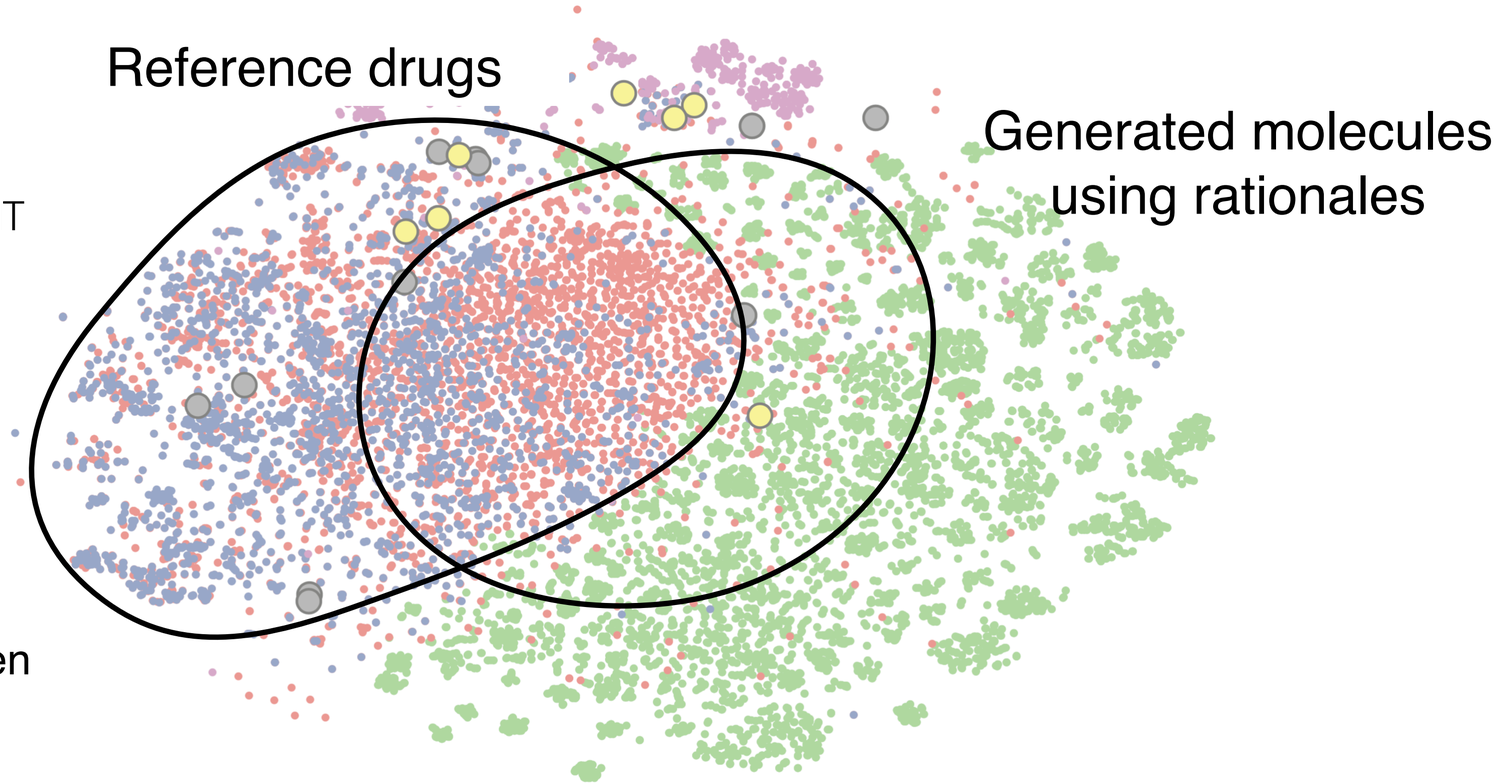
[1] Preuer et al. Frechet ChemNet distance: a metric for generative models for molecules in drug discovery

Rationale encourage reliability

- We use a property predictor to evaluate the property of generated compounds
- However, generated compounds can be far away from the drugs used to train the property predictor — that's why rationales are useful!
- Molecules generated from rationales are closer to reference drugs



FCD measures distributional discrepancy between generated molecules and training set [1]



[1] Preuer et al. Frechet ChemNet distance: a metric for generative models for molecules in drug discovery

Summary

- Molecular graph generation is particularly challenging due to multiple constraints
- In this paper, we propose hierarchical RL based on rationales
- Our model works better than previous state-of-the-art RL methods
- Methods can be further enhanced using advanced generative architectures
  - Instead of atom-by-atom generation, we can generate molecules based on substructures
  - Jin et al., Hierarchical Generation of Molecular Graph using Structural Motifs. ICML 2020
  - (poster ID 2743)