Improving Molecular Design by Stochastic Iterative Target Augmentation

Kevin Yang, Wengong Jin, Kyle Swanson,
Regina Barzilay, Tommi Jaakkola

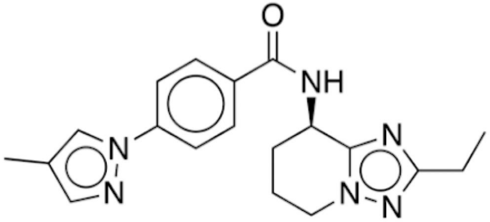
15-Second Overview

Data augmentation approach: improve molecular optimization SOTA by > 10%

Broadly useful for structured generation tasks, e.g. program synthesis (shown later)

Context: Pharmaceutical Drug Discovery

Suppose: have promising drug candidate for e.g., COVID-19



Input Molecule ***X***

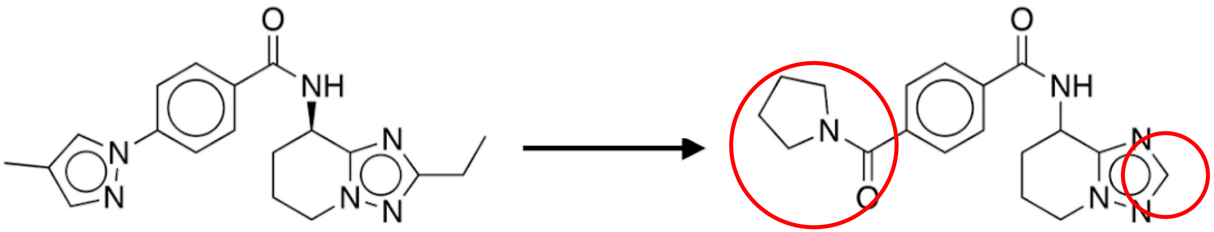
Property(***X***) = 0.78

Context: Pharmaceutical Drug Discovery

Suppose: have promising drug candidate for e.g., COVID-19

Want to make it more potent (higher property score)

Have: **Want:**

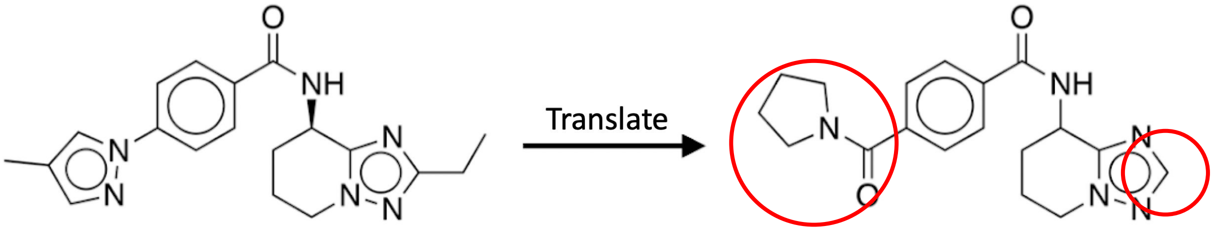


Input Molecule ***X***
Property(***X***) = 0.78

Target Molecule ***Y***
Property(***Y***) = 0.92

Task: Molecular Optimization

"Translate" input molecule to a similar molecule with better property score.

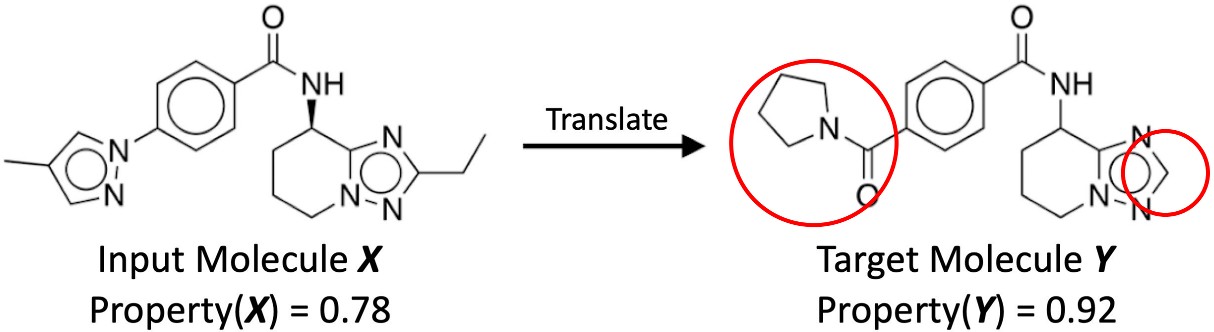


Input Molecule ***X***
Property(***X***) = 0.78

Target Molecule ***Y***
Property(***Y***) = 0.92

Task: Molecular Optimization

“Translate” input molecule to a similar molecule with better property score.



Dataset: collection of input-target pairs

Why is Molecular Optimization Hard?

Why is Molecular Optimization Hard?

Real-world ground truth evaluation: lab assay

Why is Molecular Optimization Hard?

Real-world ground truth evaluation: lab assay

- Slow + expensive!

Why is Molecular Optimization Hard?

Real-world ground truth evaluation: lab assay

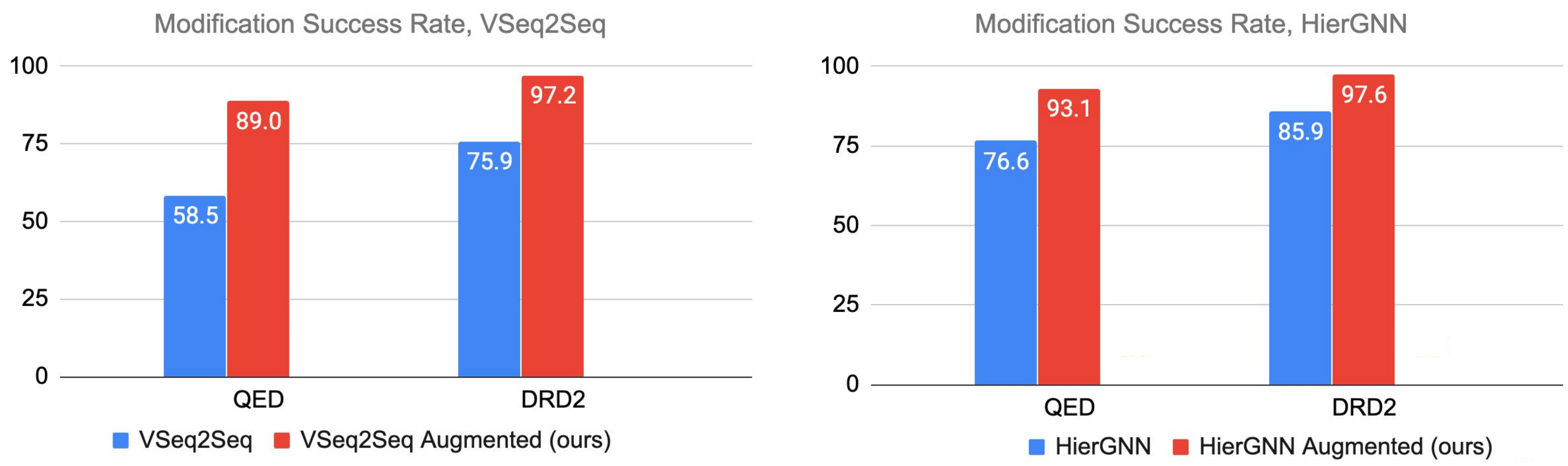
- Slow + expensive!

Key Problem: Small Datasets

Stochastic Iterative Target Augmentation

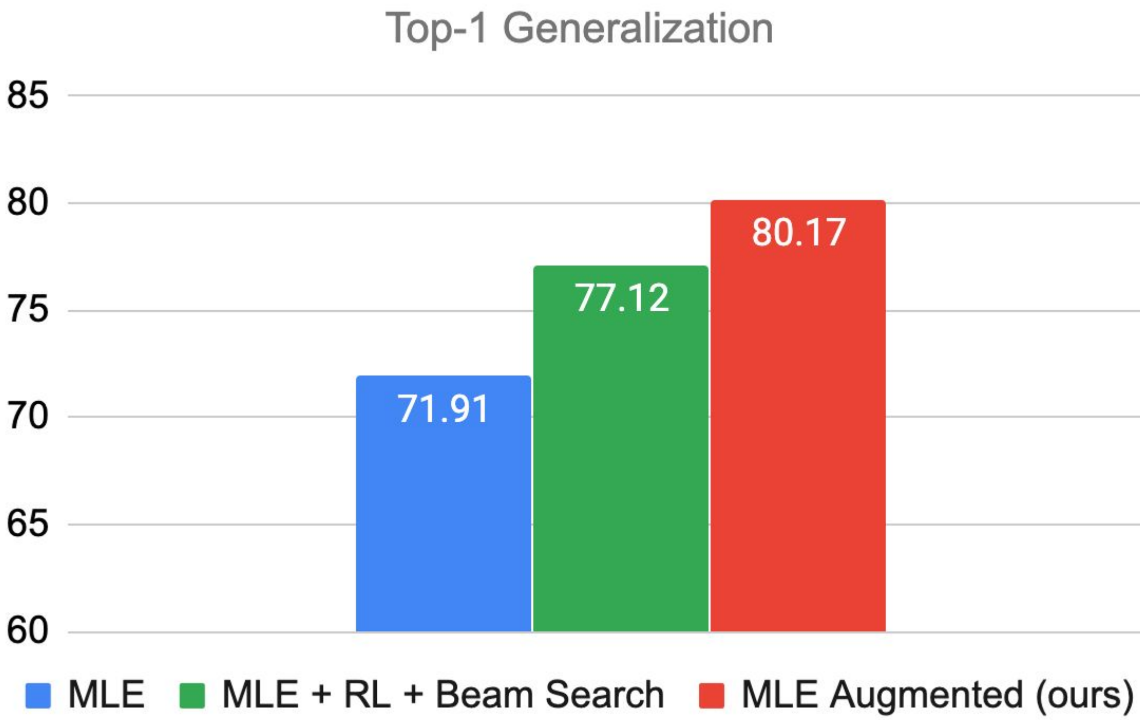
Data augmentation meta-algorithm on top of existing model

Results: Molecular Optimization



- Over 10% absolute gain over SOTA on both datasets

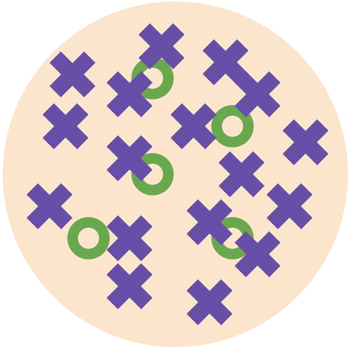
Results: Program Synthesis



Stochastic Iterative Target Augmentation

Data augmentation meta-algorithm on top of existing model

- Sample input-output pairs
  from generator

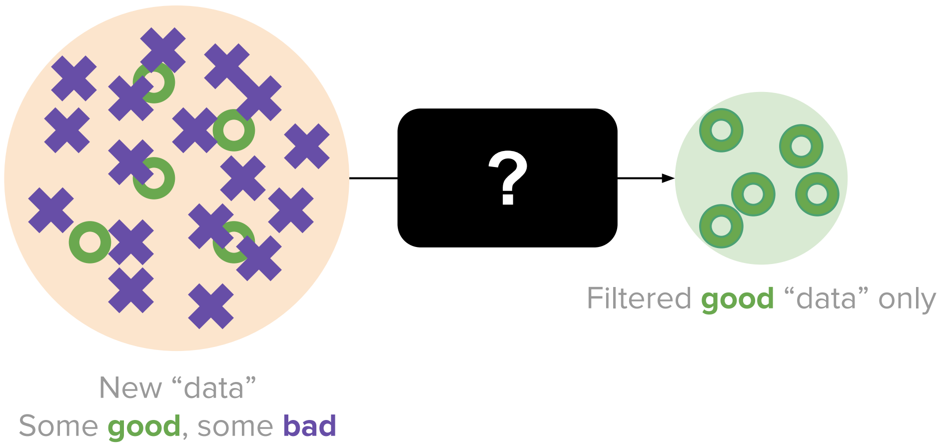


New “data”
Some **good**, some **bad**

Stochastic Iterative Target Augmentation

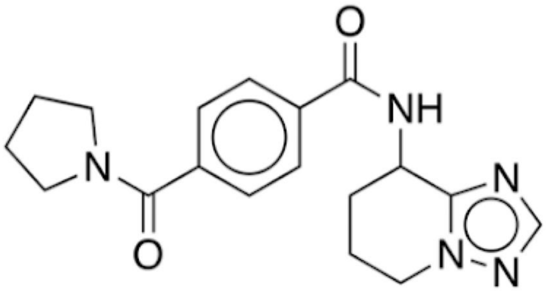
Data augmentation meta-algorithm on top of existing model

- Sample input-output pairs from generator



How to filter for only the good pairs?

Idea: Filter with Property Predictor



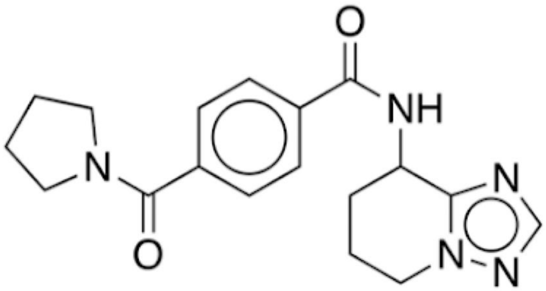
Molecule ***Y***

Predict →

Property(***Y***) = 0.92

Idea: Filter with Property Predictor

This is easier than generation!

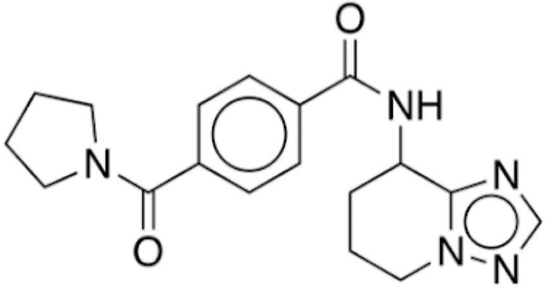


Molecule ***Y***

Predict → Property(***Y***) = 0.92

Idea: Filter with Property Predictor

This is easier than generation!



Molecule ***Y***

$\xrightarrow{\textbf{\textit{Predict}}}$ Property(***Y***) = 0.92

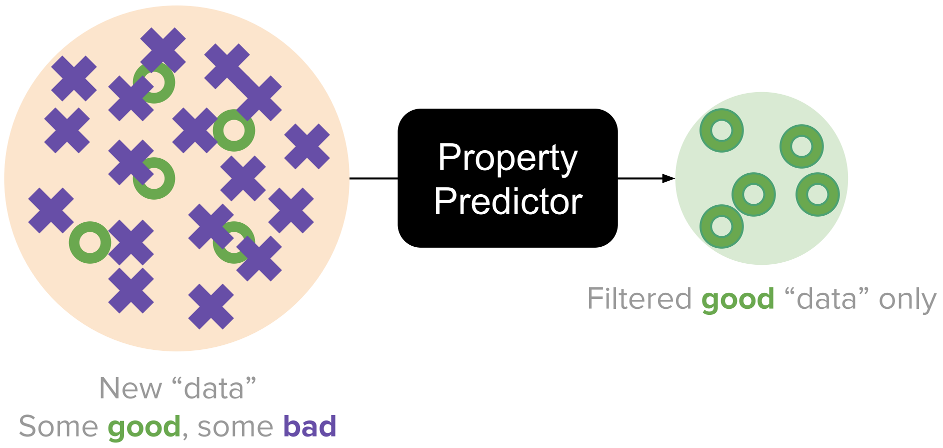
Program synthesis analogue: hard to write program, easier to run test cases

Stochastic Iterative Target Augmentation

Data augmentation meta-algorithm on top of existing model

- Sample input-output pairs from generator

- Filter with property predictor, add good pairs to training data



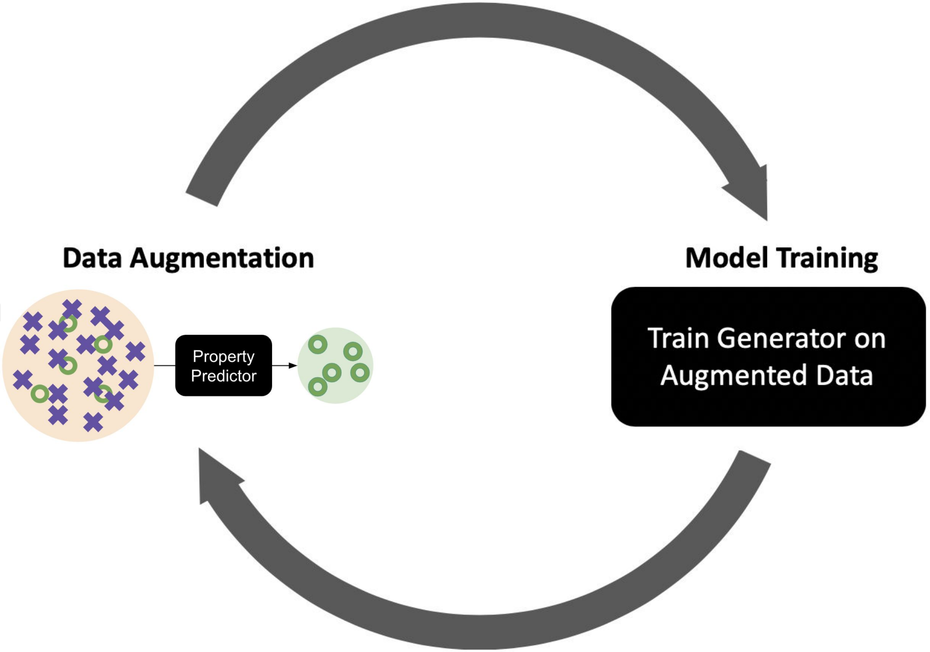
Stochastic Iterative Target Augmentation

Data augmentation meta-algorithm on top of existing model

- Sample input-output pairs
  from generator

- Filter with property predictor,
  add good pairs to training data

- Train generator, repeat



Outline

Setup + Evaluation

Detailed Method

More Empirical Analysis

Program Synthesis Experiments + Results

Outline

Setup + Evaluation

Detailed Method

More Empirical Analysis

Program Synthesis Experiments + Results

Real World Molecular Optimization

Real-world ground truth evaluation: lab assay

- Slow + expensive! (➜ small datasets)

Real World Molecular Optimization

Real-world ground truth evaluation: lab assay

- Slow + expensive! (➔ small datasets)
- Only use at final test time

Real World Molecular Optimization

Real-world ground truth evaluation: lab assay

- Slow + expensive! (→ small datasets)
- Only use at final test time

Use fast + cheap *in silico* (i.e., computational) predictor for model validation

Test time only

Lab Assay

Data used to train →

Can use anytime

in silico

Evaluation Setup

(Lab assay, *in silico* predictor) become (*in silico* predictor, proxy predictor)

Evaluation Setup

(Lab assay, *in silico* predictor) become (*in silico* predictor, proxy predictor)

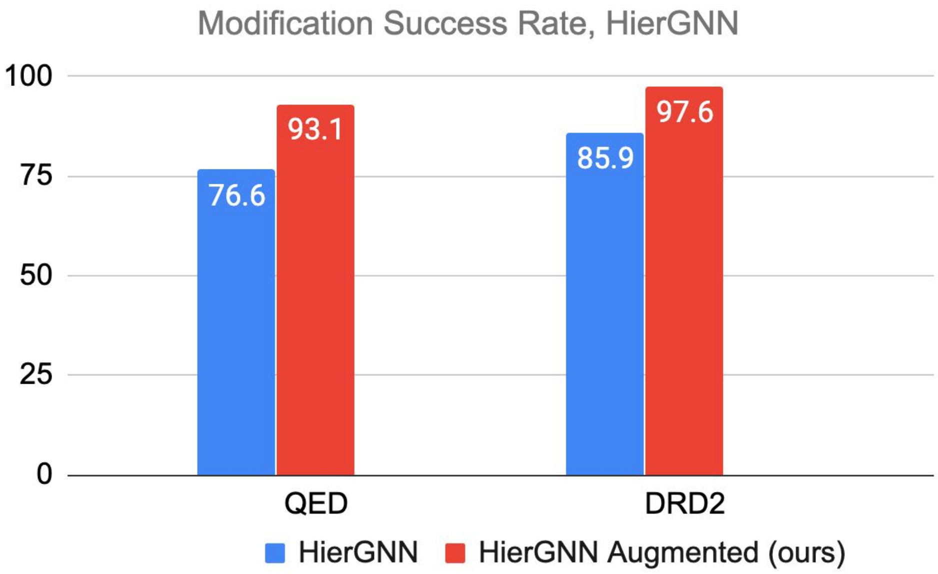
- Just train proxy on property values of molecular optimization training pairs

Metric

“Success” if even 1/20 tries passes ground truth evaluator

Metric

“Success” if even 1/20 tries passes ground truth evaluator



Molecular optimization is hard…

Outline

Setup + Evaluation

Detailed Method

More Empirical Analysis

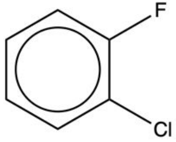
Program Synthesis Experiments + Results

Stochastic Iterative Target Augmentation

Goal:

Input Molecule

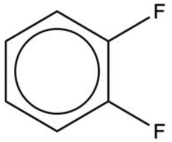
Clc1ccccc1F

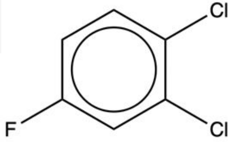


Somehow →

New Correct Targets

Fc1ccccc1F
Clc1ccc(F)cc1Cl

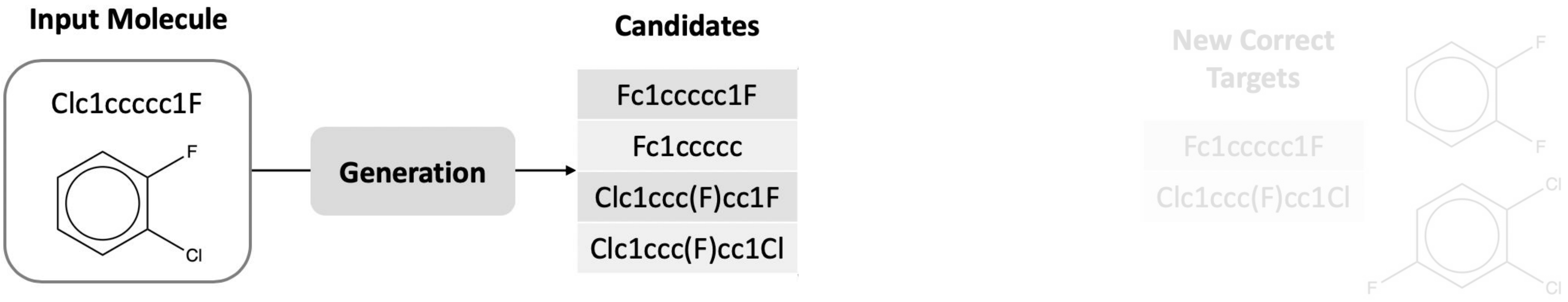




Target augmentation: Augment the set of correct targets for a given input.

Stochastic Iterative Target Augmentation

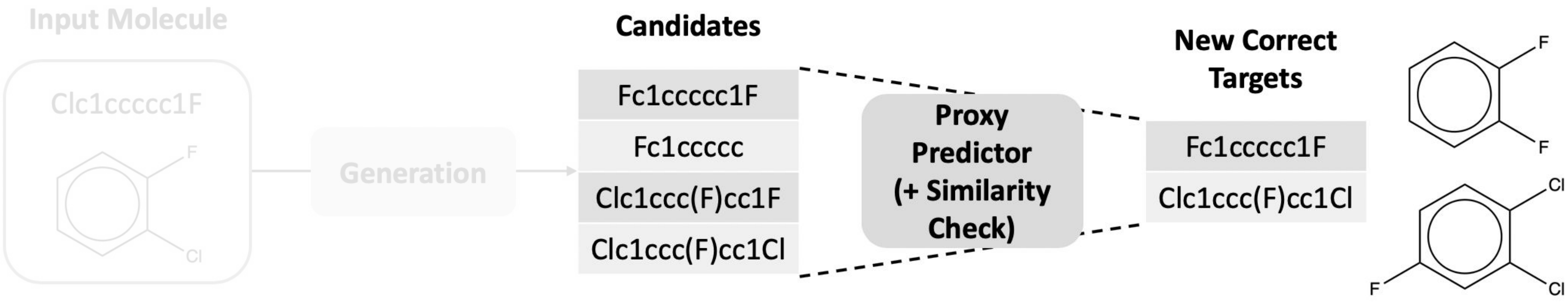
1. Given inputs, sample input-target pairs from current generative model



Target augmentation: Augment the set of correct targets for a given input.

Stochastic Iterative Target Augmentation

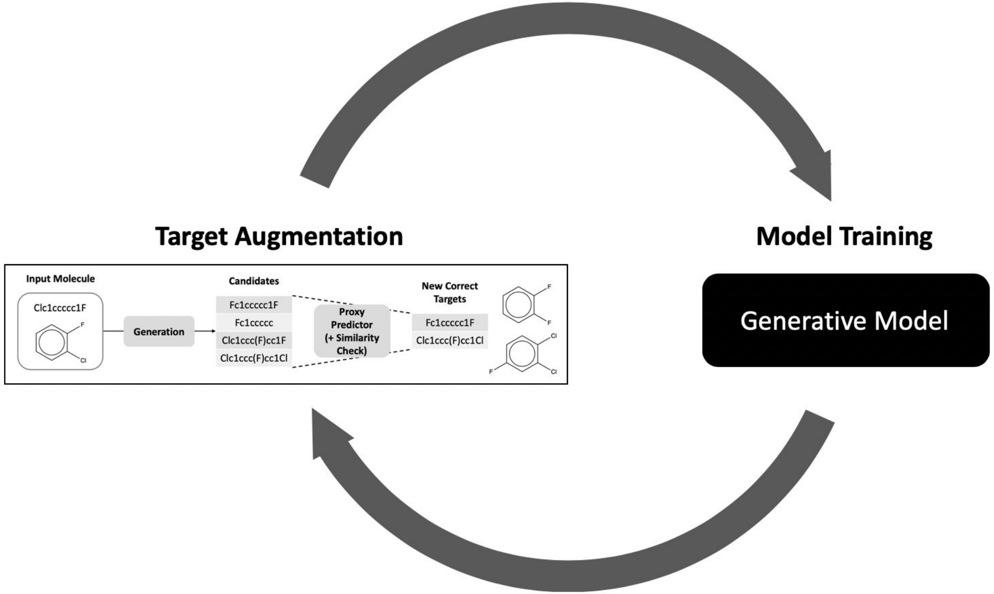
1. Given inputs, sample input-target pairs from current generative model
2. Filter candidate input-output pairs using property predictor



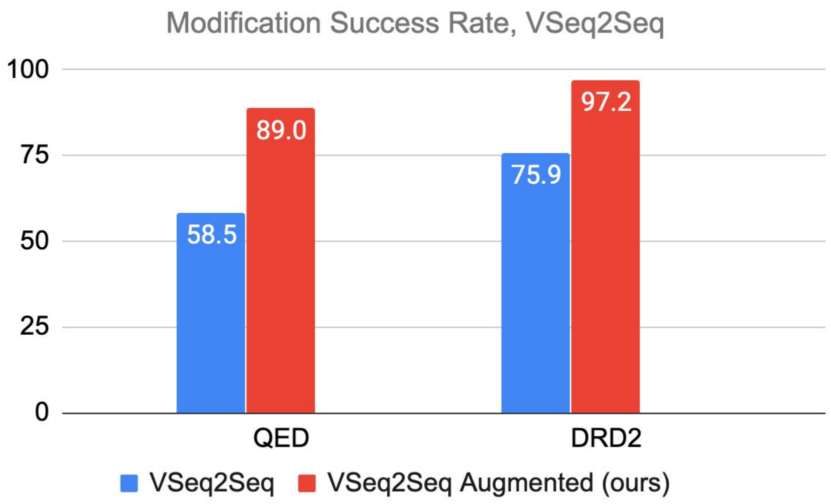
Target augmentation: Augment the set of correct targets for a given input.

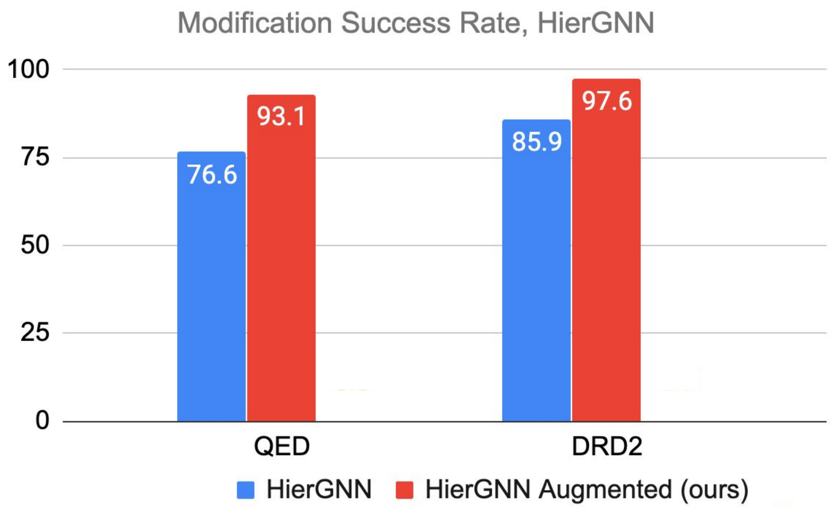
Stochastic Iterative Target Augmentation

1. Given inputs, sample input-target pairs from current generative model
2. Filter candidate input-output pairs using property predictor
3. Add good pairs to training data, train model, repeat



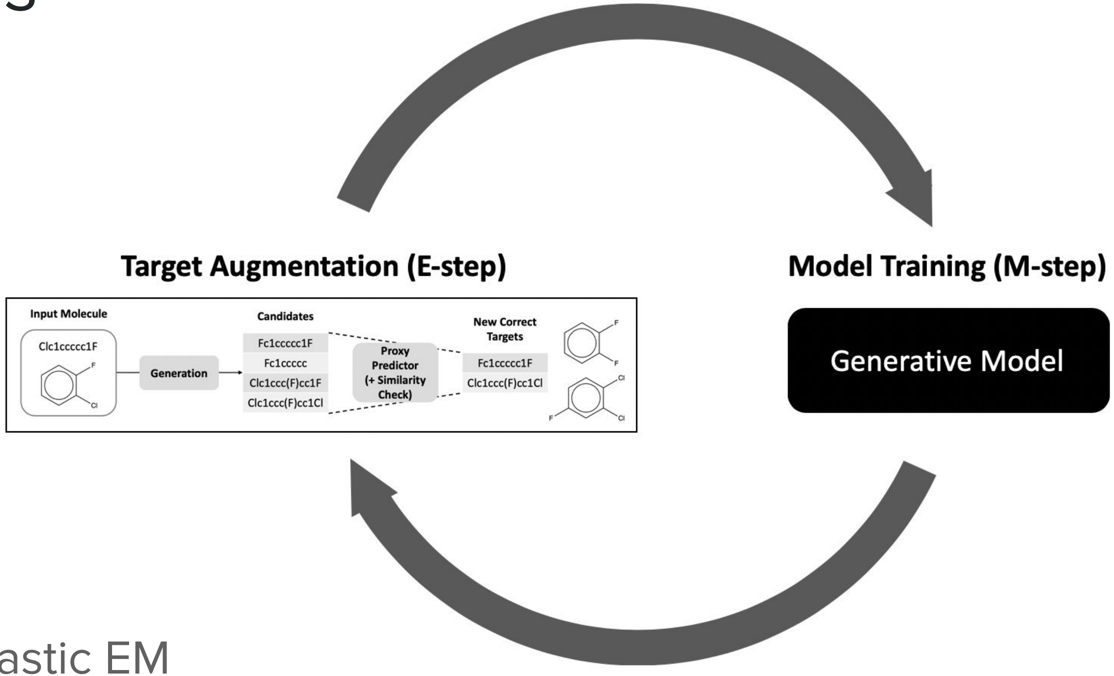
Results: Molecular Optimization





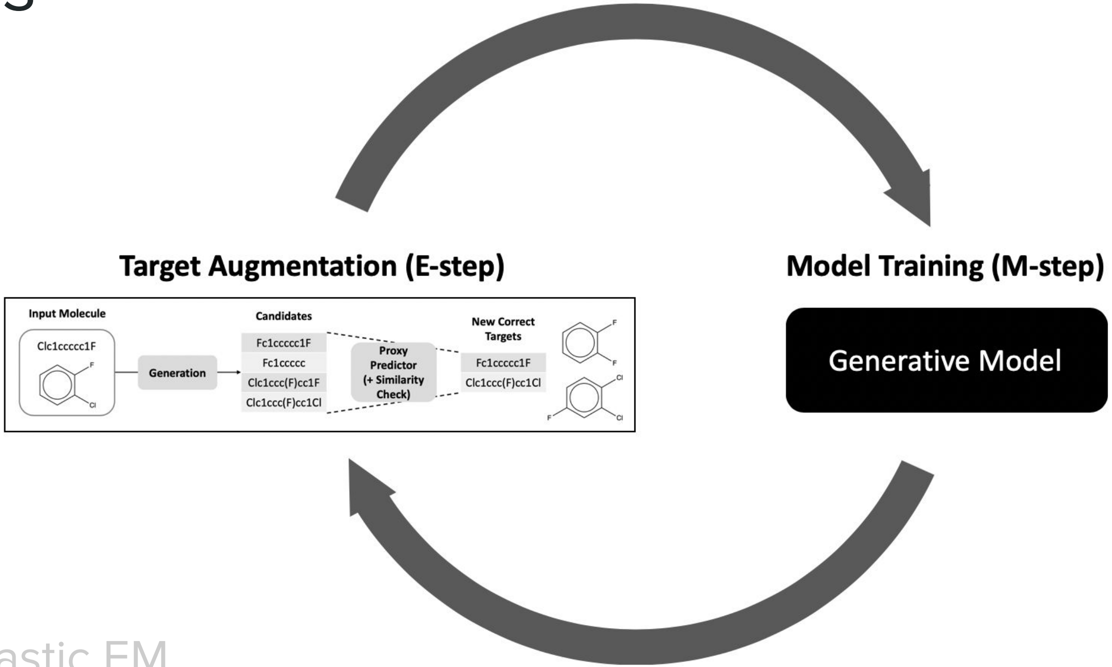
- Over 10% absolute gain over SOTA on both datasets

Observations



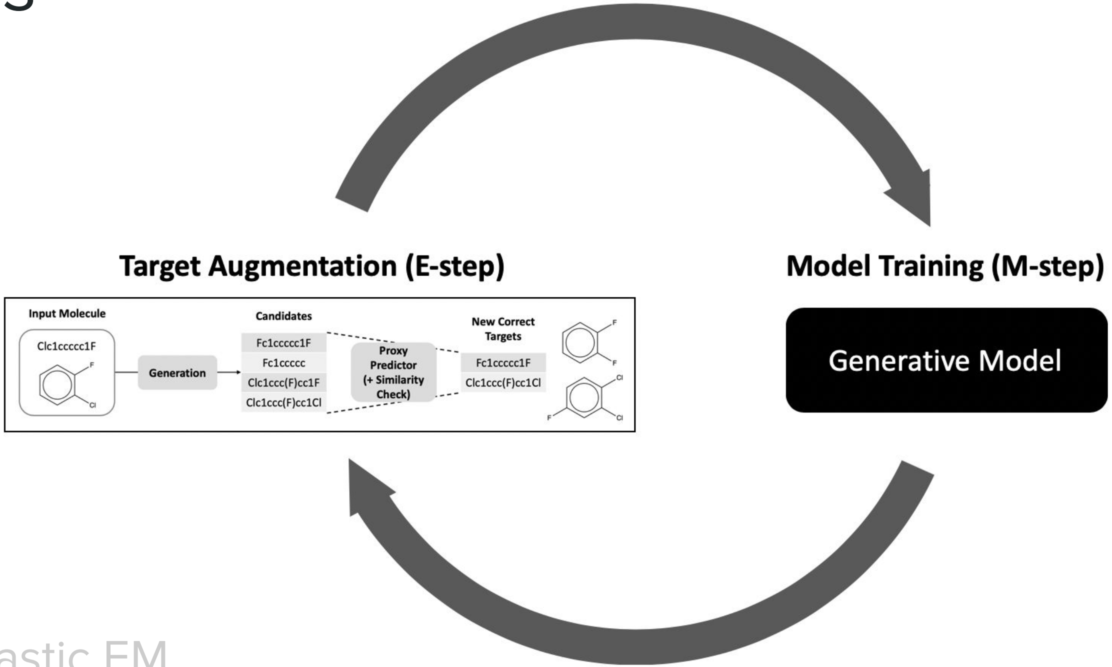
- View as Stochastic EM

Observations



- View as Stochastic EM
- Why iterative? Better generator ➔ easier to find new correct targets

Observations



- View as Stochastic EM
- Why iterative? Better generator ➔ easier to find new correct targets
- May as well use proxy to filter samples at test time too

Outline

Setup + Evaluation

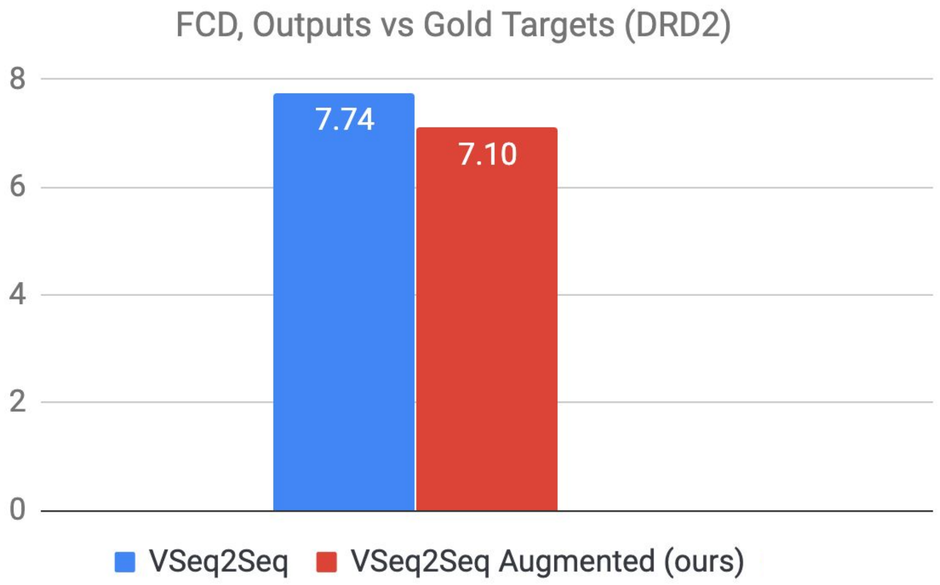
Detailed Method

More Empirical Analysis

Program Synthesis Experiments + Results

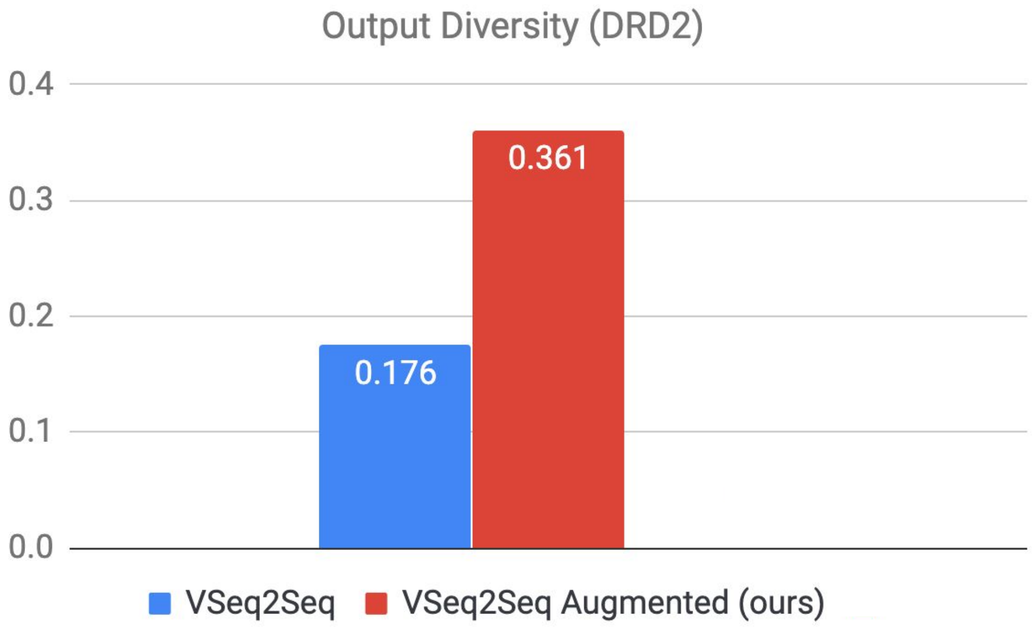
Frechet Chemnet Distance Analysis

FCD (embedding distance) is the molecular analogue to Inception distance in images. Lower is better.

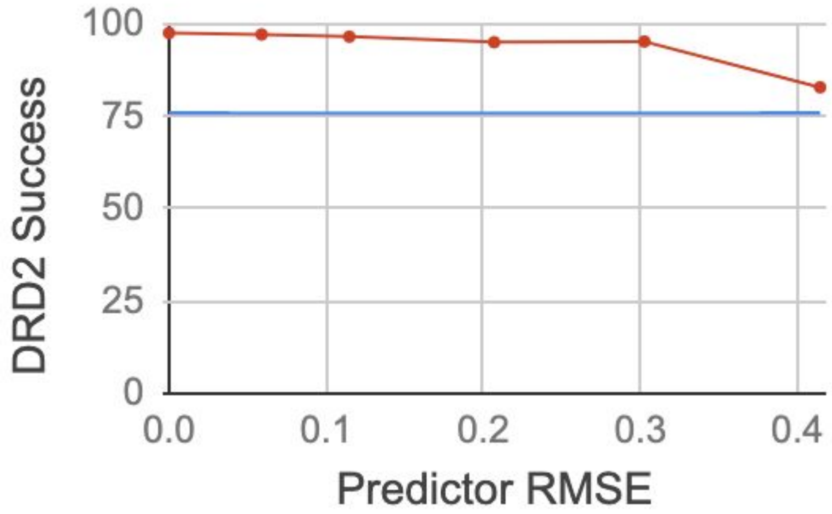


Improved Diversity

Diversity: average distance between different correct outputs for the same input



Robustness to Predictor Quality



Far left point is oracle (ground truth); second-from left is learned proxy predictor.
Blue line indicates baseline performance.

Outline

Setup + Evaluation

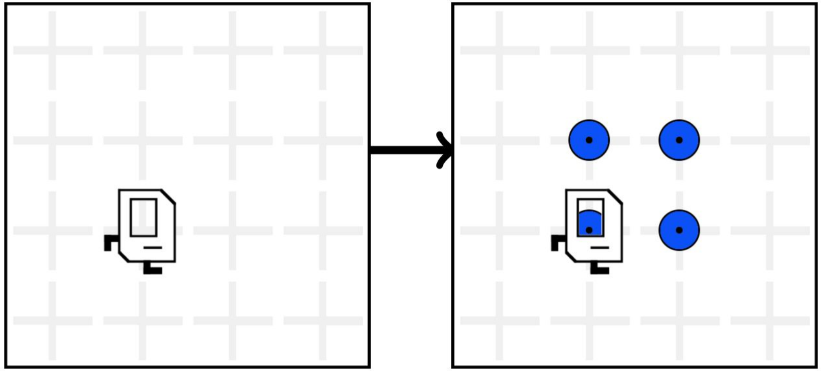
Detailed Method

More Empirical Analysis

Program Synthesis Experiments + Results

Program Synthesis Task: Karel Dataset

Inputs: Test Cases



Outputs: Programs

Program A

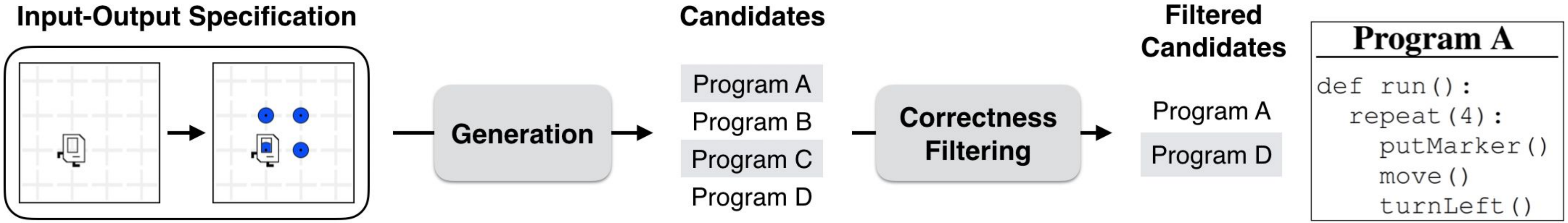
```
def run():
  repeat(4):
    putMarker()
    move()
    turnLeft()
```

Program B

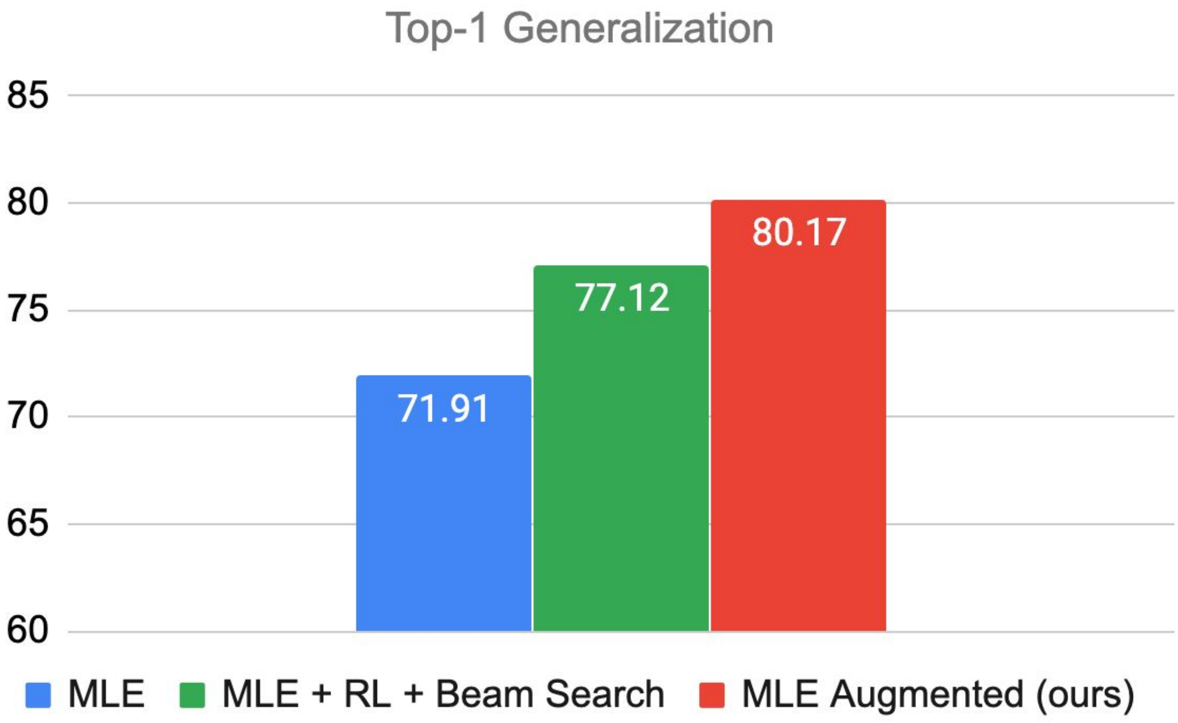
```
def run():
  while(noMarkersPresent):
    putMarker()
    move()
    turnLeft()
```

Evaluate correctness using held-out test cases

Program Synthesis Target Augmentation



Results: Program Synthesis



Summary

Data augmentation meta-algorithm for improving performance on structured generation tasks

Summary

Data augmentation meta-algorithm for improving performance on structured generation tasks

Significantly improves over SOTA in molecular optimization: > 10%

Summary

Data augmentation meta-algorithm for improving performance on structured generation tasks

Significantly improves over SOTA in molecular optimization: > 10%

Applicable to other domains: program synthesis

Summary

Data augmentation meta-algorithm for improving performance on structured generation tasks

Significantly improves over SOTA in molecular optimization: > 10%

Applicable to other domains: program synthesis

Thanks for Watching!