Team name	LambdaZero	
Team member(s) (firstname lastname; …)	Brooks Paige	tbpaige@gmail.com
	Jose Miguel Hernandez Lobato	jmh233@cam.ac.uk
	John Bradshaw	jab255@cam.ac.uk
	Joanna Chen	joanna.w.chen@yale.edu
	Bianca Dumitrascu	biancadumit@gmail.com
	Matt Kusner	matt.kusner@gmail.com
	Marwin Segler	marwin.segler@benevolent.ai
	Jarrid Rector-Brooks	jarridrb@gmail.com
	Paul Bertin	paul.f.bertin@gmail.com
	George Lamb	william.lamb.19@ucl.ac.uk
	Simon Verret	simon.verret@mila.quebec
	Jian Tang	tangjianpku@gmail.com
	Will Hamilton	wlh@cs.mcgill.ca
	Chenghao Liu	liucheng@mila.quebec
	Bruno Rosseau	bruno.rousseau@mila.quebec
	Kostiantyn Lapchevsky	lapchevsky.k@gmail.com
	Aga Slowik	agnieszka.slowik44@gmail.com
	Michael Bronstein	m.bronstein@imperial.ac.uk
	Emmanuel Bengio	bengioe@gmail.com
	Doina Precup	dprecup@cs.mcgill.ca
	Pierre-Luc Bacon	pierre-luc.bacon@mila.quebec
	Yoshua Bengio	yoshua.bengio@mila.quebec
	Scott Fujimoto	scott.fujimoto@mail.mcgill.ca
	Pierre Thodoroff	pierthodo@gmail.com
	Clement Gehring	clement.gehring@gmail.com
	Shivam Patel	patelshi@mila.quebec
	Victor Butoi	vib9@cornell.edu
	Samira Kahou	samira.ebrahimi.kahou@gmail.com
	Riashat Islam	riashat.islam@mail.mcgill.ca
	Howard Huang	howardhuang256@gmail.com
	Evgenii Nikishin	nikishin.evg@gmail.com
	Moksh Jain	mokshmuk@ualberta.ca

	Sumana Basu	sumana.basu@mail.mcgill.ca
Affiliation	Mila - Quebec AI Institute, collaboration with University of Montreal, McGill University, University of Cambridge, Imperial College London	
Contact email	mkkr@mit.edu	
Contact phone number (optional)		
Protein targets (for example: 3CLPro/Nsp5, BoAT1, Fc Receptor, Furin, IL6R, M protein, Nspx, OrfXx, N, E, etc…)	M protein	

Section 1: methods & metrics

Describe what methods you have used, how they are independent from one another, what your workflow was, how you performed the cross-correlation between your methods. If applicable, please report estimated performance metrics of your methods, such as accuracy, sensitivity, false-discovery rate, etc., and how those metrics were obtained (e.g. cross-validation). Please provide key references if available.

Methods:

This project will leverage a previously tested, novel and yet unpublished deep reinforcement learning algorithm derived from the recent successes of DeepMind's AlphaZero and of MIT's approach to represent drug molecules. It also uses a new approach to search molecular space using reinforcement learning algorithms developed by the team at MILA and collaborating universities led by prof. Yoshua Bengio (whole team listed at: https://mila.quebec/en/ai-society/exascale-search-of-molecules/). This new approach is based on the definition of a set of molecular building blocks which can be combined to form molecules and search the space at a more abstract level, as opposed to searching by modifying individual atoms.

The research plan consists of the following main steps. (1) Run docking simulations on 200M randomly chosen molecules to bind on virus targets. (2) Keep the highest score as training examples for the initial value function network which approximates the docking scoring function. (3) Train Lambda-Zero (our novel RL algorithm) using the docking score, drug-likeness, and predicted cost of synthesis as the reward function and select top-scoring molecules. (4) Send these molecules to simulated retrosynthesis collaborators (Molecule.one, Dr. Piotr Byrski) (which will identify which of these molecules can be fabricated) and actual chemical synthesis. (5) Send best molecules designed by RL algorithm to computational biophysics collaborators (DE Shaw Research) to perform (a) long-term molecular dynamics and (b) Free Energy Perturbation (6) Send the best performing molecules in the MD simulations to our experimental collaborators for biological assays (IRIC, Dr. Mike Tyers) (measuring actual biological binding to the target) and biomedical assays (measuring protein binding energy, and effectiveness in cells and then animal models). (7) Feed back the results of these assays as additional data to constrain and improve the search, iterating back to step (3) but with a modified

reward function which incorporates both docking and empirical data from (5) as part of a trained reward function neural network.

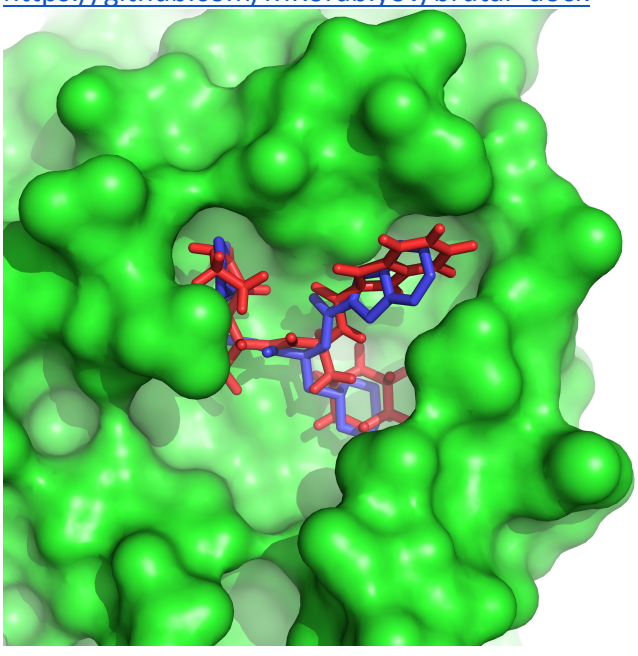
The project will deliver molecular structures deemed good candidates for antiviral drugs. If these molecules have good binding energy to the targets, this should convince biomedical collaborators to pursue the pipeline of evaluation of these molecules, leading eventually to clinical trials as well as compassionate use distribution. The project will also deliver code for efficient implementation and parallelization of the previously developed LambdaZero algorithm honed in for the search of SARS-CoV-2 antivirals.

Section 2: targets

Describe for each protein target: why you chose it, from which source you obtained it (e.g., insidecorona.net / covid.molssi.org / rcsb.org) and why this is the best quality structure, if any pre-processing (e.g., energy minimization, residue correction, alternative folding, …) was performed.

The SARS-CoV-2 main protease (M pro) crystal structure was obtained from rcsb.org, which was first reported in *Nature*, **2020**, *582*, 289-293. The M protein was chosen because it is a key enzyme in coronaviruses that mediates replication and transcription. The crystal structure used in our studies has a high resolution of 2.1 Å. It also already contains specific interaction to a ligand (N3, *PLoS Biol*. **2005**, *3*, e324), which helps to identify new drug molecules. We evaluated several docking algorithms including AutoDock, internally developed version of AtomNet (Heifets at al), Dock 3.6, and Dock6, and various ligand 3D structure generation methods and over 100 crystal structures.

While we could have opted for a more powerful docking/scoring pipeline, our key objective here is to prove the power of RL algorithm as a method for the search in the space of small molecules. The construction of the docking pipeline is a hyperparameter of the RL algorithm that could easily be substituted later. Therefore in this experiment, for 3D conformer generation we have chosen rdkit ETKDG, and 20 poses were generated and further minimized with MDFF. A single best conformer was chosen for docking with Dock6. The docking setup and results are available under MIT license: https://github.com/MKorablyov/brutal_dock



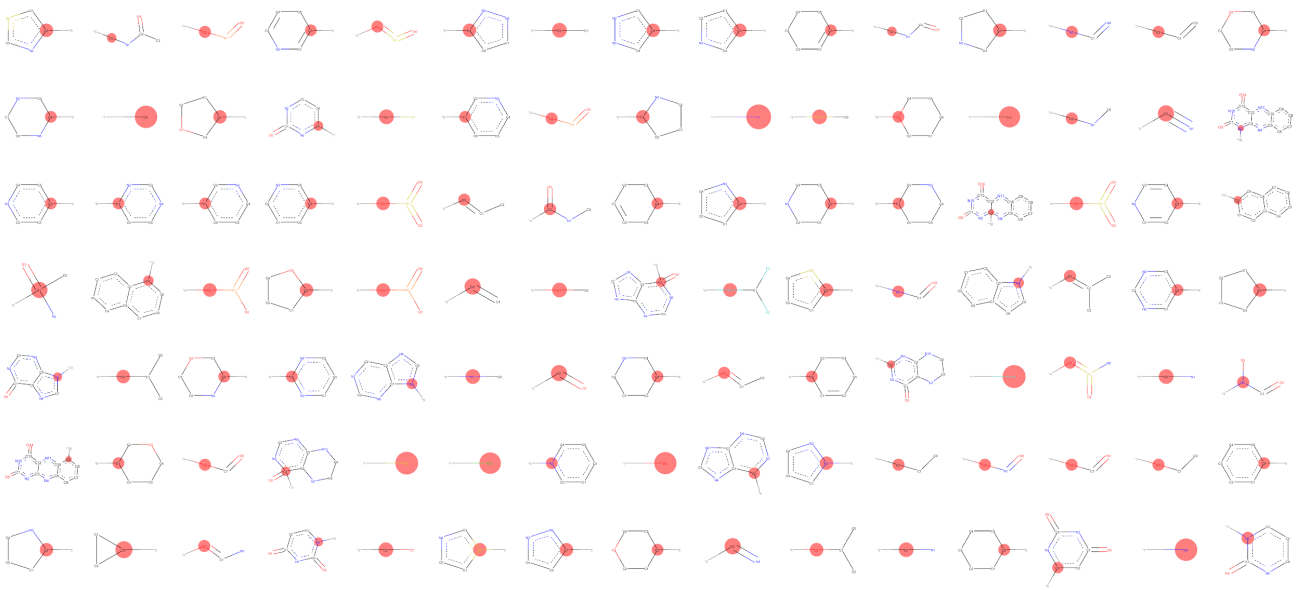
From the SMILES string of a strong (100nmol IC50 inhibitor [reference]), we could well recover the existing crystal structure also with favorable predicted binding energy of -55 kCal/mol.

Section 3: libraries

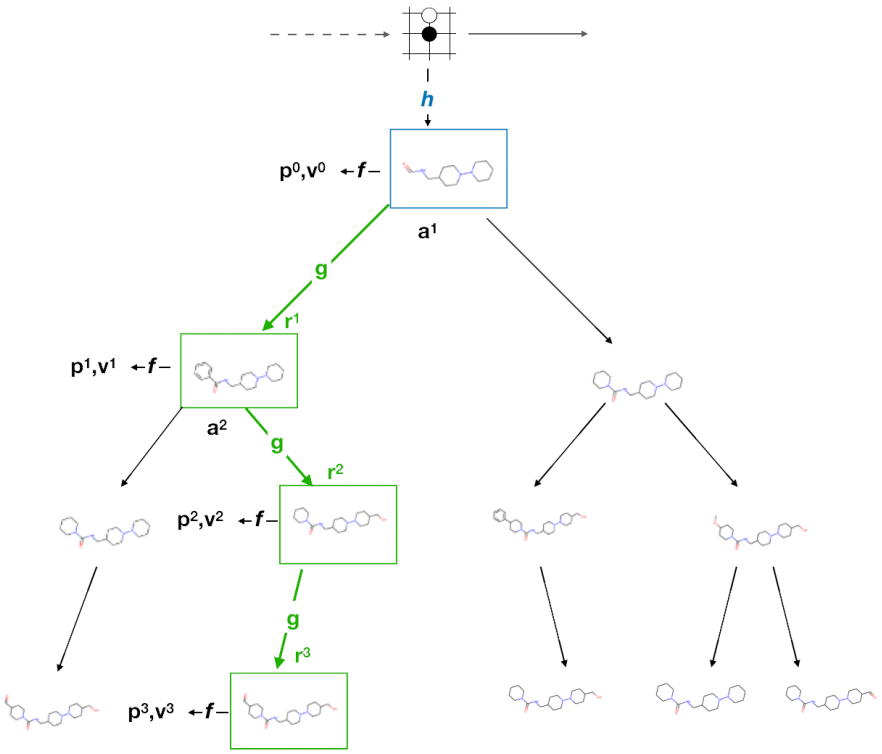
Describe which libraries you have used, how they were combined, if any compounds were removed / added, why additions are relevant, any unique features of your library, etc. Please provide the sources you obtained the libraries from (if publicly available). Describe the procedure of data preparation (removal of duplicates, standardization, etc). Indicate if different libraries were used for different targets, and why. If possible, provide a download link to your version of the library.

Library:

Instead of relying on a standard library of ready to buy molecules such as Zinc ready, or a library of molecules that would be easy to generate such as Enamine Real, here we teach an RL algorithm to act in a space of primitives, or building blocks, that it can join together in various ways.



For an action space, we have chosen 105 blocks listed here. It could be noted that some of the blocks, like benzene rings, could be generated from other blocks - carbon atoms - both present here.



A typical Monte Carlo Search Tree in LambdaZero algorithm. Starting from a root, random molecule, the algorithm subsequently expands the molecule by adding and removing blocks.

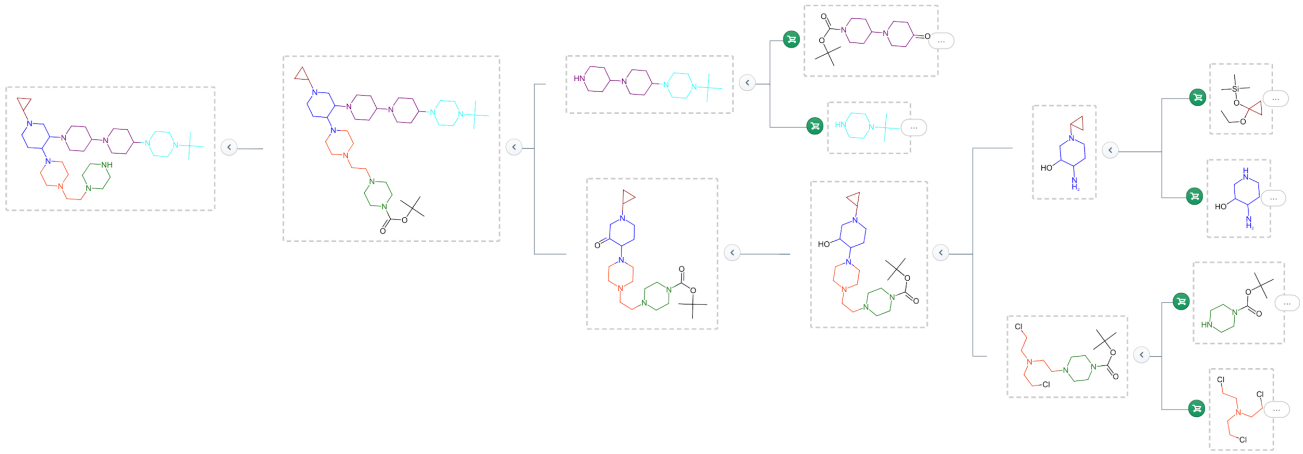
Chemical feasibility of the library

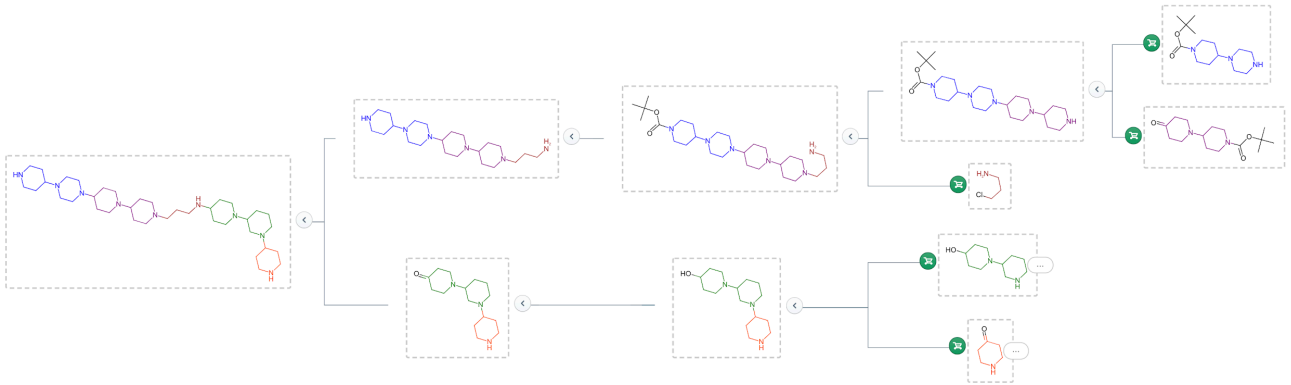
While choosing a good vocabulary of the building blocks already adds a strong prior on the sampled space of molecules (for example we found 30% of random walks to be synthesizable) we opted out for an additional synthesis loss in the core of the RL algorithm.

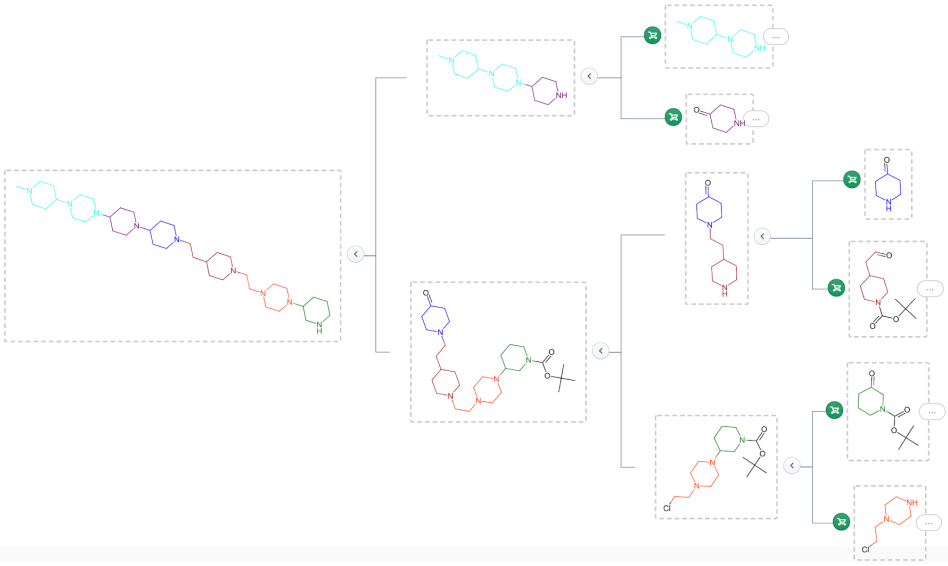
Our collaborators recently published state-of-the-art algorithms to predict chemical synthesis path [molecule one].

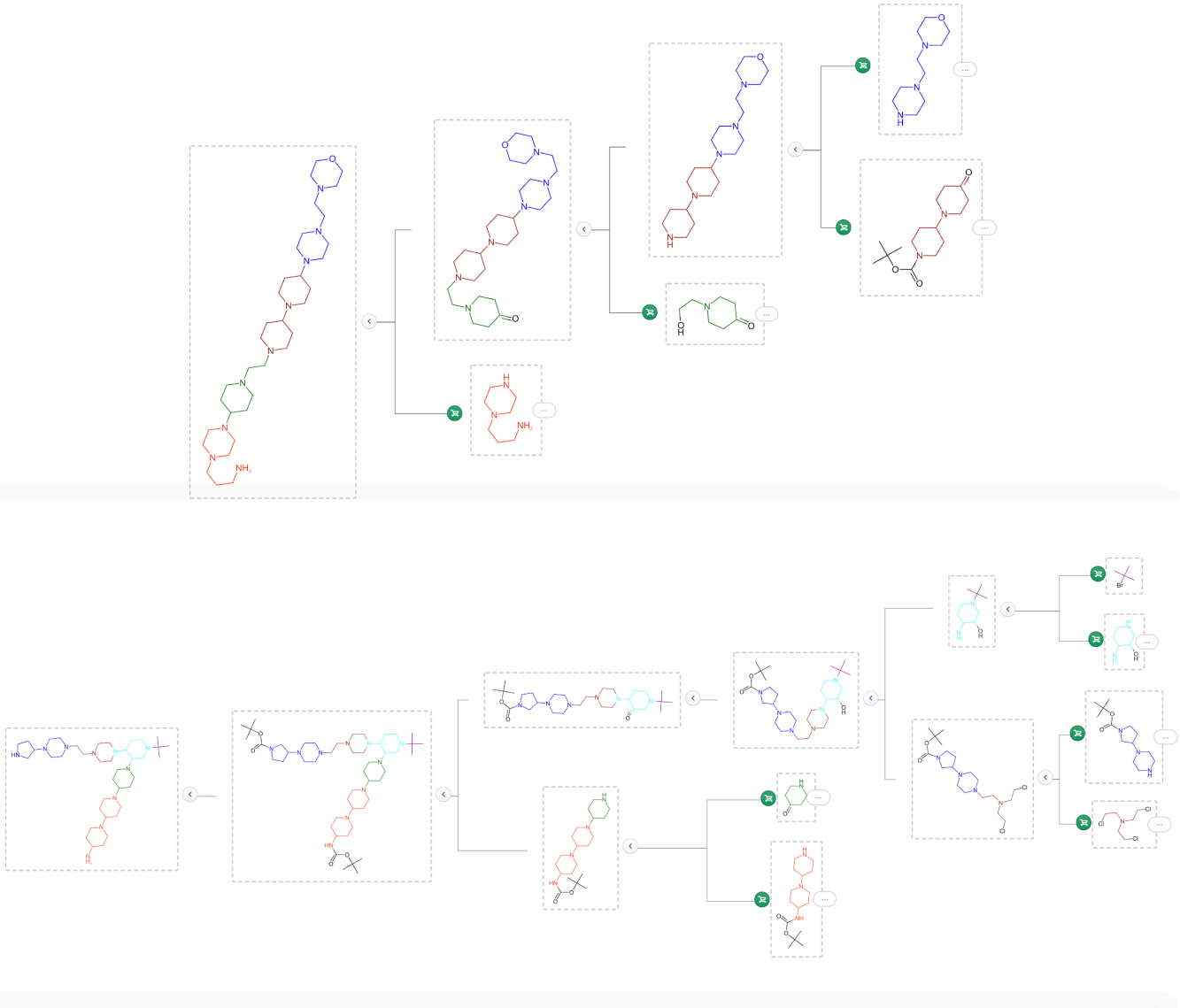
Algorithm	Collection	Sample Size	Synthesizable	Route coverage
Eli Lilly & Company	DrugBank, approved & investigational	6059	2477	40.1%
Molecule.one	Zinc15, investigational	2848	1995	**70.0%**
Molecule.one	Zinc15, FDA approved	1609	1354	**84.2%**
Molecule.one	Random walk	10000	2166	21.7%

We independently Benchmarked Molecule One's algorithm on several known molecular datasets, and found it to be quite comparable to the state of the art.



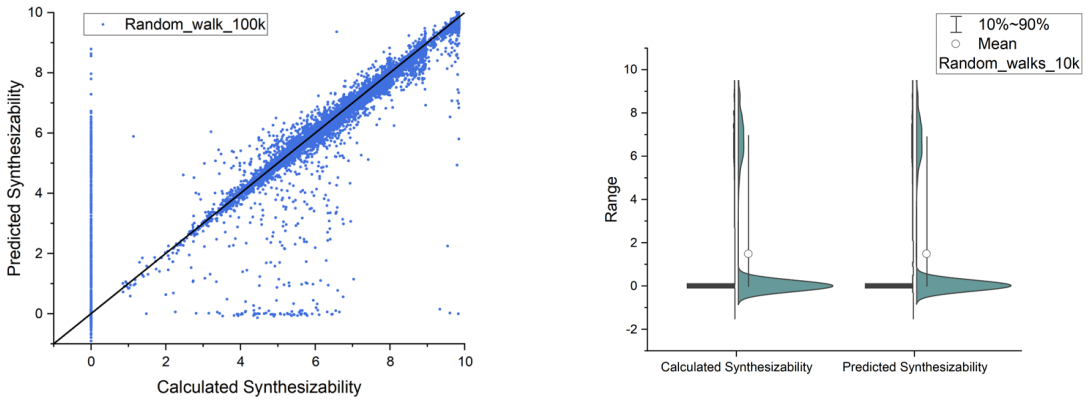






Synthesis path for two among the best binders predicted by the LambdaZero RL algorithm to bind SARS-CoV-2 Mpro. Manual review of the synthesis path by our computational chemists (Marwin Segler, Chenghao Liu) was very favorable.

At peak, our algorithm evaluates up to 1.6B molecular structures/day, and molecule.one API takes 50-60/molecule on average. The speed of the synthesis prediction was not fast enough to us. Therefore, based on ChemProp Message Passing Neural Network [https://github.com/chemprop/chemprop] recently developed by a group at MIT, we developed a proxy that only took ~25 GPU milliseconds to evaluate, which has only a small error compared to the initial algorithm.



Correlation plot and distribution plot of calculated synthesizability (0 is non-synthesizable, 10 is commercially available) from Molecule.one and that predicted by MPNN, in 100k randomly generated molecules. Note there are only 287 false positives and 74 false negatives.

In our approach, we have decided that molecules predicted to be not synthesizable would receive the reward of 0 regardless of any other qualities of the molecule. Hardly synthesizable molecules with predicted synthesizability score between 0 and 5 would only receive a fraction of the original reward.

Drug-likeness:
We have chosen a QED coefficient available in RdKit as an additional reward. Molecules with good QED of > 0.5 (average in Zinc-drug-like) were positively rewarded.

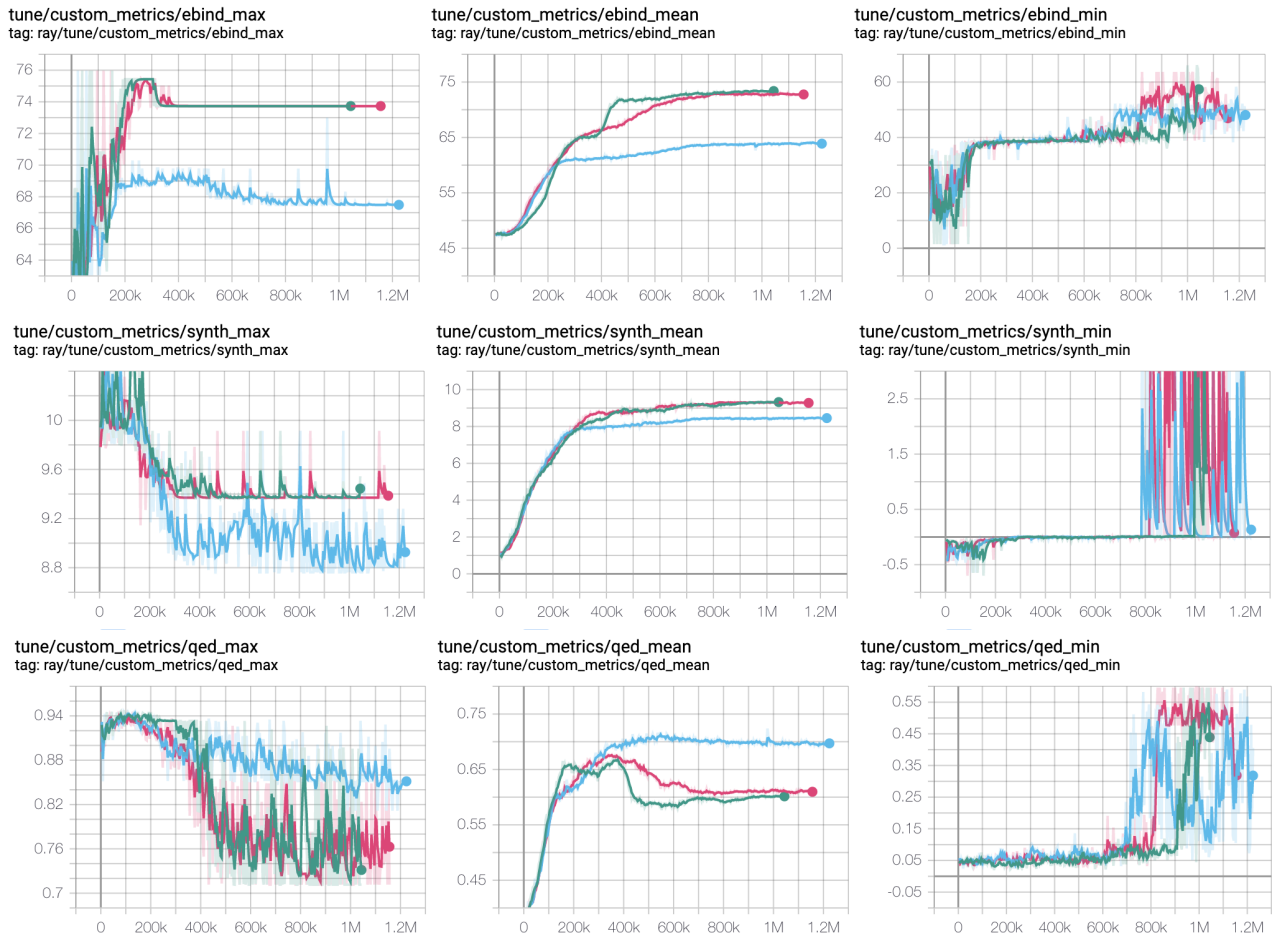
Section 4: results
Briefly describe you key findings, any interesting trends in your data, a description of your top 5 compounds for each target. If possible, provide a link to a code and/or data repository. Please do not submit randomly selected compounds!

Results:

Baseline:
We initially started our development with google's MOL DQN algorithm [https://arxiv.org/abs/1810.08678]. Initial code had very poor parallelization properties (using a blocking CPU thread for reward computation) and could only produce 5000 molecules/day on a machine with 1 GTX 1080 GPU and 8 CPUs. A simple rewrite of the code unblocking the reward thread resulted in ~8x speedup ~40K/molecules/day. Changing the environment to building blocks rather than atoms resulted in an additional 5 x speedup due to fewer steps required to complete the molecule ~200K molecules/day. Moving all of the agent and reward computations to GPU and partial GPU allocation per agent (0.3 V100 GPU/s per agent) resulted in an additional ~8 x speedup ~ 1.6M molecules/GPU/day.
Finally, when we added a restart to our environment (i.e. every starting state is a previously found good molecule which can technically be counted as a viable molecule), if we count every molecule on the path evaluated by RL agent as a screened molecules, we can say it's 19.5 M molecules/GPU/day.

After optimization, we did not find MolDQN as a reasonable baseline but have chosen (a) a very large random search (b) default implementation in ray.rallib of Proximal Policy Optimization, and Ape-X as a baseline to compare against.

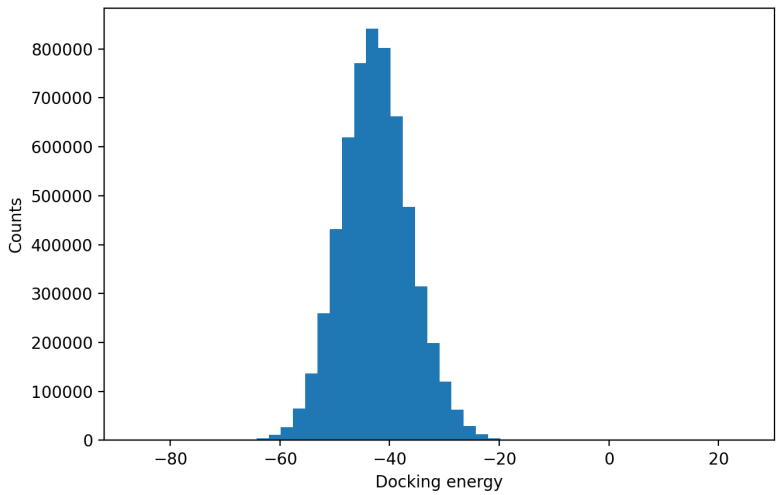
Proximal Policy Optimization Results:
We performed about 250 end-to-end training of PPO models with various hyperparameters. The best PPO result is shown below.



We observe that predicted binding energy improves in training. Generated molecules are predicted to be easily synthesizable (synthesizability of 10, higher is better) and reasonably drug-like with the QED coefficient to be > 0.65 for our molecules (average QED for Zinc Drug-like is 0.5).

LambdaZero:
We found a molecule design game to be notably similar to GO, in particular, discrete action space, environment is fully-observable, very high branching factor of ~100/move in GO, and ~800/move per molecule. We take full advantage of the tree-structure of the search problem and are using MCTS/UCT earlier used in AlphaZero, but not particularly popular in RL.

Pretraining:
As a starting point for the RL, we selected 7 million diverse unique drug-like molecules from Zinc. We trained agents based on a Message Passing Neural network on this initial dataset - this allowed us to initialize hyperparameters at a meaningful point.



Histogram.

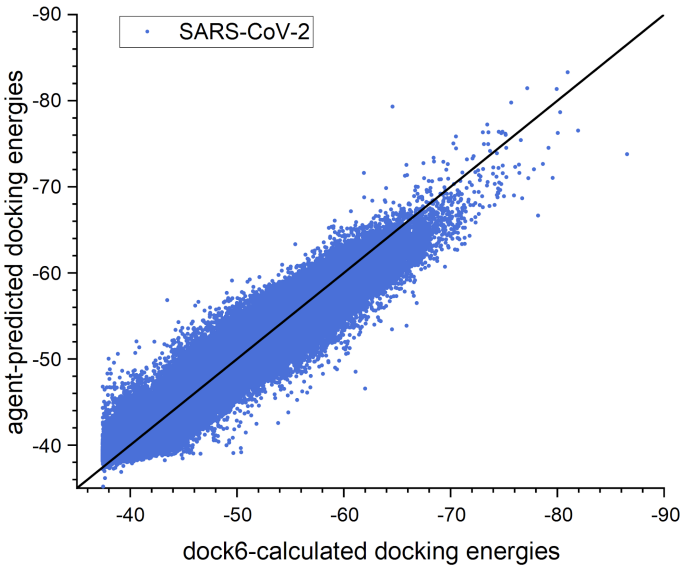
Histogram of docking energies of ~7 million Zinc15 molecules to M pro generated by dock6. The average docking energies in this dataset is -42.54, and the standard deviation is 6.32. The highest value is -81.94

Training:

We have chosen a setup of 40 V100 GPUs, and 160 CPUs on Beluga server of Compute Canada. On the speed benchmark, LambdaZero could enumerate 800M molecules/day by pure brute force. But incorporation of RL makes the algorithm much more powerful compared to random search as any of the generated molecules are already biased to bind.

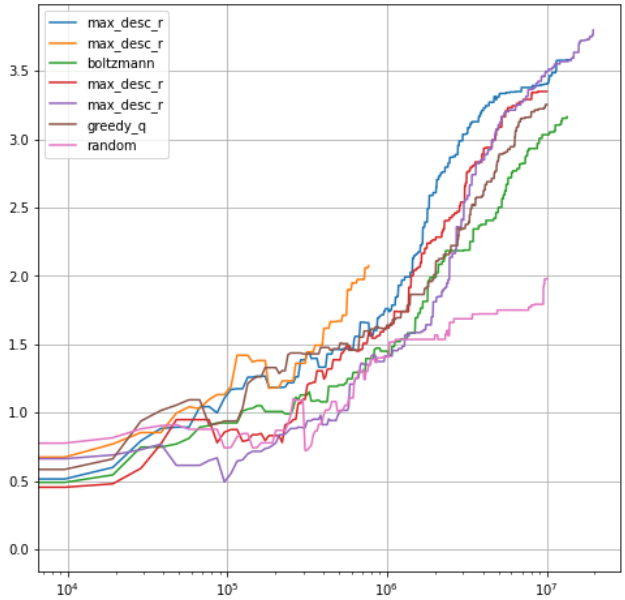
LambdaZero learned an neural estimator of the docking energy:

It is important to clarify that LamdaZero only has to physically dock a small fraction of the molecules it samples. Most of the time, the agent relies on a learned Message-Passing Neural Network to estimate the quality of the molecule without actually performing the time-consuming process of docking. We found the agent to be capable of learning the docking function almost perfectly.



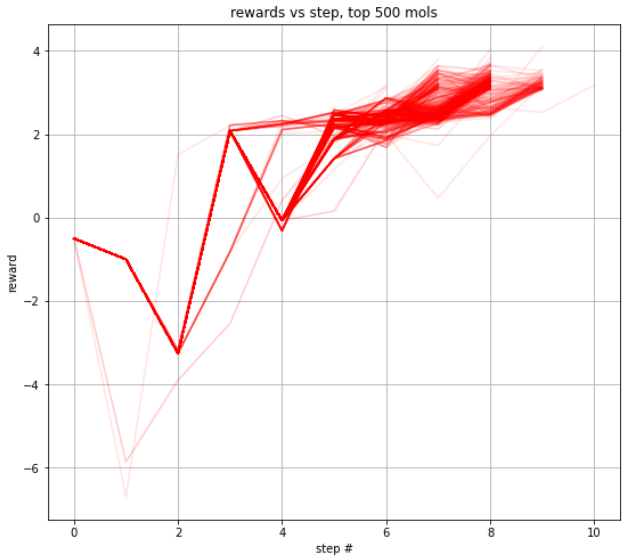
Predicted by the agent docking energy plotted against real docking dock6 energy shows that agent learned a representation that approximates docking energy extremely well.

LambdaZero strongly outperforms random search.



Reward plotted against the number of molecules sampled. “random” is a random search. All other curves are various versions of LambdaZero. While with the small number of molecules < 1M, LamdaZero could not considerably outperform enumeration of random molecules, with bigger sample size LambdaZero starts to find patterns in the data and considerably outperform random search. Version of LambdaZero are “max_desc_r” trained to predict value of the max descendent of a molecule, boltzmann sampling in “boltzmann” is boltzmann sampling in MCTS with respect to the value function, “greedyQ” picks greedily according to the value function.

LambdaZero learns to avoid immediate rewards for future reward.



LambdaZero learned to take risk in molecule optimization. Some intermediate steps could decrease the combined reward (binding energy, synthesizability, QED) of the molecule, but the final

molecule would still be much better compared to the initial properties. The trajectories for only 500 best found molecules are shown.

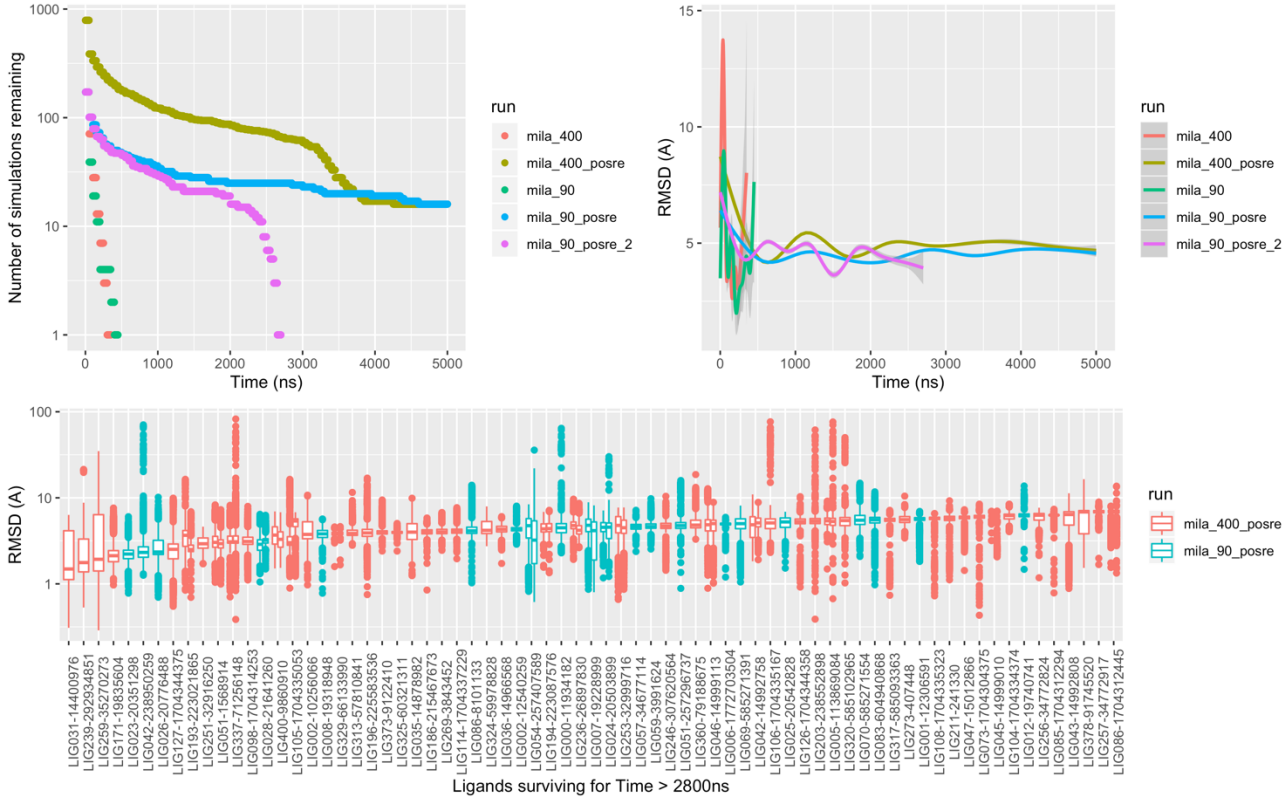
LambdaZero finds radically better molecules compared to any initially present in the 7M diversity subset from Zinc.

Compound SMILES	Dock6 energy
CC(C)(C)N1CCN(C2CCN(C3CCN(C4CN(C5CC5)CCC4N4CCN(CCN5CCNCC5)CC4)CC3)CC2)CC1	-95.818619
C(CNC1CCN(C2CCCN(C3CCNCC3)C2)CC1)CN1CCC(N2CCC(N3CCN(C4CCNCC4)CC3)CC2)CC1	-95.094940
CN1CCC(N2CCN(C3CCN(C4CCN(CCC5CCN(CCN6CCN(C7CCCNC7)CC6)CC5)CC4)CC3)CC2)CC1	-94.968063
NCCCN1CCN(C2CCN(CCN3CCC(N4CCC(N5CCN(CCN6CCOCC6)CC5)CC4)CC3)CC2)CC1	-92.938660
CC(C)(C)N1CCC(N2CCN(CCN3CCN(C4CCNC4)CC3)CC2)C(N2CCC(N3CCC(N4CCC(N)CC4)CC3)CC2)C1	-90.472382

For better reproducibility, we post dock6 setups used to train LambdaZero algorithm with MIT license online: https://github.com/MKorablyov/brutal_dock/tree/master/mpro_6lze/docksetup And dock6 could be found here: https://github.com/MKorablyov/dock6 but additional license from dock6 creators may be required.

Summary table of the best molecules found. We note the Dock6 energies of the top generated compounds are 2.20 standard deviations higher than the highest in Zinc15 dataset (-81.94), and 8.43 standard deviations away from the average in Zinc15 dataset.

Molecules found in Zinc dataset remain bound in long-term MD simulation.



Our collaborator performed MD simulations of molecules found in Zinc to get some additional evaluation of the docking algorithm. “Mile_90_posre” are the molecules from the docking algorithm we ended up using that show good retention in the binding site even after 5000ns of the simulation.

Experimental evaluation:
We ordered best performing molecules from Zinc and also the simulation to be delivered and tested using fluorescence assays and other experimental techniques on June 6th.