Scalable reinforcement learning-based framework for generation of synthetic pathways in synthesis-oriented de novo drug design

Project Notes

Aaron Tian - aztian@wpi.edu
Massachusetts Academy of Math and Science
Worcester, MA

Note-Taking

: Concept
: Follow-Up Citation
: Knowledge Gap
: Vocabulary
: Important Metric

Knowledge Gaps

This list provides a brief overview of the major knowledge gaps for this project, how they were resolved and where to find the information.

Knowledge Gap	Resolved By	Information is located	Date resolved
Reinforcement Learning Math	Aaron Tian	Project Logbook	1/10/22
Vector to Vector Change of Variables, Probability Density	Andrew Lee	Project Logbook -> Professional Communication -> Andrew Lee	9/27/2021
How are chemical reactions encoded for neural networks?	Aaron Tian	Source Code (Data)	1/10/22
How is QED calculated?	Aaron Tian	Source Code	12/20/21

Github Links:

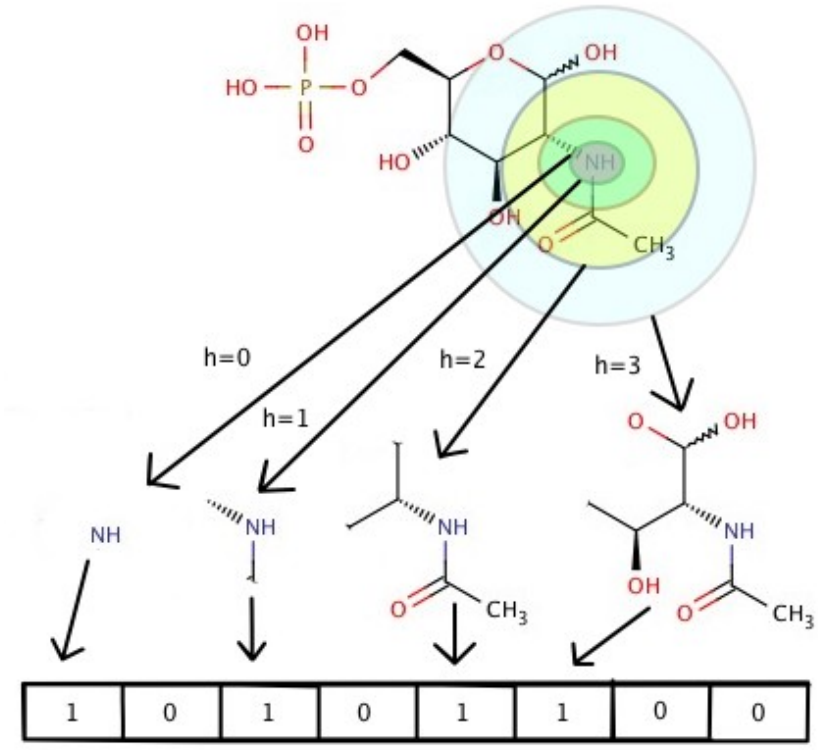
1. TrimNet: https://github.com/yvquanli/trimnet
2. GraphAF: https://github.com/DeepGraphLearning/GraphAF
3. RetroXPert: https://github.com/uta-smile/RetroXpert
4. AiZynthFinder: https://github.com/MolecularAI/aizynthfinder

Literature Search Parameters

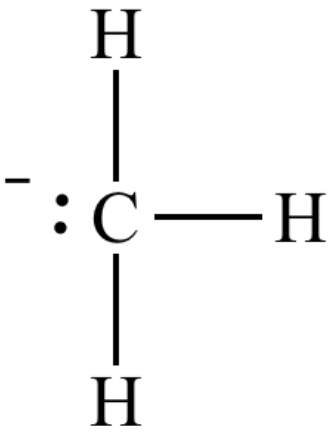
These searches were performed between (Start Date of reading) and XX/XX/2019.
List of keywords and databases used during this project.

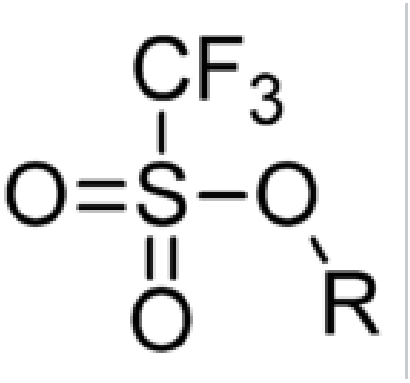
Database/search engine	Keywords	Summary of search
Google	Lead optimization and reinforcement learning	(Zhou et al., 2019) [3] is the only search result that incorporates reinforcement learning.
WPI Library	Molecules AND Graph Neural Networks	Resulted in many relevant sources saved to Zotero.
WPI Library	Graph Neural Networks AND Drug Design	Resulted in some of the same results as the previous search, though [5] and other more drug-design specific papers were found via this search.
Google	MPNN	Yields pioneering work of Gilmer et al. (2018) and related studies.
Google	Connor W. Coley	Coley has published many interesting studies related to my field of research.

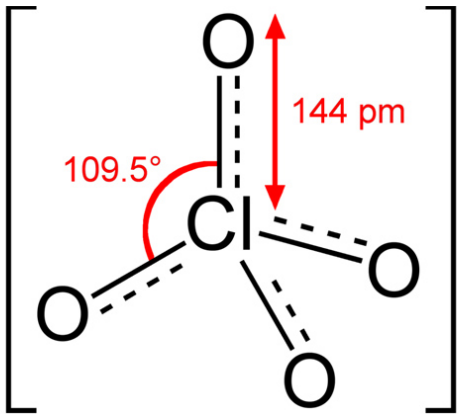
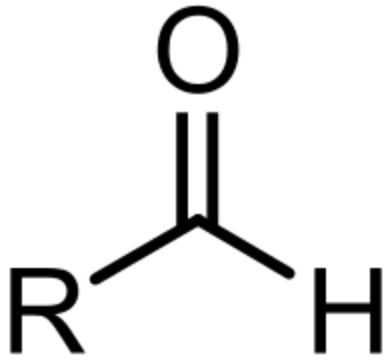
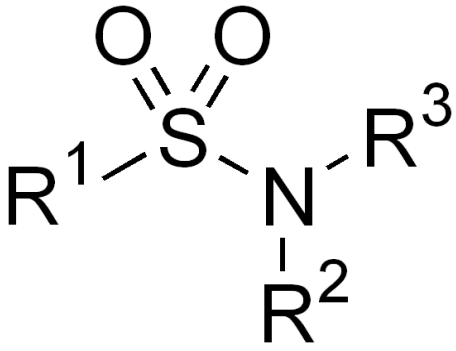
Vocabulary

Term	Definition	Source
Markov Decision Process	A discrete-time mathematical framework to model decision making.	[3]
Morgan Fingerprint	 source (molecular fingerprint ECFP4) a topological fingerprint for molecular characterization. Effective for predicting activity for small biological molecules but aren't perceptive of global features such as size and shape.	[3]
Argmax	Finds the maximum value (typically for a continuous function)	[3]
Exploration vs Exploitation	Choice between trying new actions vs. using currently optimal ones. An obstacle to training reinforcement learning models.	[3]
High Throughput Screening	A tool for running millions of biological/chemical tests at a time by screening molecules in parallel.	[4]
Attention Mechanism	A component of deep learning architecture that manages and quantifies interdependence between data (further research needed).	[4]

Ablation Study	Removal of a specific component of a model to analyze its effect on the network's performance.	[4]
CPI	Compound-Protein Interaction	[4]
Scaffold Splitting	Method of splitting molecular data based on their 2D structural frameworks, implemented in RDKit. More here	[4]
ROC metric	Receiver operating characteristic. Represents the performance of a binary classifier model at all classification thresholds. Includes TPR (true positive rate) and FPR (false positive rate)	[4]
Pharmacological Activity	The effects (beneficial or adverse) of a drug on living matter.	[5]
Gene Ontology	A bioinformatics initiative to unify the representation of gene and gene product attributes across all species.	[5]
Ligand Binding Pocket	Region of a protein that binds to another molecule.	[5]
Binding Pose	The configuration of the protein and substrate when bound together.	[5]
DTI	Drug Target Interaction.	[5]
Molecular Docking	Study of how two or more molecules fit together.	[5]
Putative	Generally assumed.	[5]
Activity Cliffs	Pairs of groups of structurally similar compounds that are active against the same target but have large differences in potency.	[5]
Nonautoregressive	Opposite of autoregressive. Used in one-step generation of graphs.	[5]
Autoregressive	Predicts future values based on past values. Used in stepwise graph generation.	[5]
Reversible Flow Model	Generative deep learning model driven by change of variables (from probability density) More background here: https://medium.com/ai-ml-at-symantec/introduction-to-reversible-generative-models-4f47e566a73	[5]
Kullback-Leibler divergence	(Relative entropy) A method of measuring the difference of two probability distributions.	[5]
Nash Equilibrium	Scenario in which each player in a game has nothing to gain by changing their current strategy.	[5]

Mode Collapse	An issue when a generative model over-optimizes for a particular discriminator. As a result, the generator rotates through a small set of outputs.	[5]
Junction Tree	A modified graph used to extract marginalization in general graphs.	[5]
Adversarial Training	Use of deceptive input data to intentionally fool a machine learning model. Can be used to protect against data inconsistencies when training.	[5]
Turing Test	A test comparing algorithmically generated molecules with ones generated by medicinal chemists.	[5]
Retrosynthesis	Method of synthesis planning that involves deconstructing a target molecule into smaller components.	[5]
Integer Linear Programming	Objective function and constraints are restricted to be integers.'	[5]
Octet Rule	Tendency of atoms to have 8 electrons in the valence shell.	[5]
Alkylation	Transfer of an alkyl group from one molecule to another.	[5]
Carbanion		[5]
Monte Carlo	Computational method that relies on random sampling to obtain numerical data.	[5]
QSAR	Quantitative Structure-Activity Relationship. Computational method of revealing relationships between structural properties of chemical compounds to biological activity.	[6]
Gradient Boosting	A ML strategy that involves breaking a prediction model down into multiple weaker prediction models.	[6]
Regularization	Strategy for fine-tuning a model to certain constraints by adding a penalty term to the loss function.	[6]
AUC	Area under ROC curve (receiver operating characteristic) for evaluating classifier models.	[6]

Reaction Center	Set of bonds that will be disconnected in the retrosynthesis process	[7]
Auxiliary Task	Training a model on an easier task first before tuning it to learn a harder task.	[7]
Reaction Template	Feeding information about chemical reactions to a machine learning model.	[7]
Canonical SMILES	Standard SMILES. Canonical means to map many to one (many molecular configurations to one representation)	[7]
Metabolite	Substances involved in metabolism (typically small molecules)	[8]
Chemoinformatics	Use of computational methods to study chemistry.	[8]
Tautomer	Isomer of a molecule that exists in solution or in cell.	[8]
Salient	Most important.	[8]
Stereoisomer	Isomers that differ in spatial arrangement of atoms.	[8]
Desolvation Energy	Energy required to displace surrounding solvent molecules in a solution to allow an enzyme to bind with its substrate.	[8]
Triflate	Functional group 	[8]
Alkyl Halide	Compound in which one or more hydrogens in an alkane are replaced with halogens.	[8]
Perchlorate	Compound containing the perchlorate ion.	[8]

		
Aldehyde	Functional group. 	[8]
Sulfonamide	Functional group that is the basis of several drugs. 	[8]
Metalloenzyme	Enzymes containing metal ions.	[8]
4N + 2 Rule	Huckel's Rule. Number of pi bonds in an aromatic ring must equal a multiple of 4, +2.	[9]
Marginal Likelihood	Likelihood function in which some parameter variables have been marginalized.	[9]
Annealing	Recombine DNA in double stranded form following separation by heat.	[9]
Antihypertensive	Class of drugs used to treat high blood pressure	[9]

Jensen-Shannon Divergence	Aka Information Radius = total divergence to the average.	[9]
Normalizing Flow	Method for constructing complex complex distributions by transforming a probability density via a series of invertible mappings.	[10]
Affine Transformation	Geometric transformation that preserves lines and parallelism. (Euclidean geometry)	[10]
Invertible Transformation	A map between vector spaces with an inverse map.	[10]
Dequantization	Using real-valued noise to convert discrete data into continuous data.	[10]
Jaccard Similarity	Aka Tanimoto Similarity	[12]
AlogP	Atomic logP	[12]
Symbolic AI	Branch of AI concerned with representing human knowledge/behaviors in a declarative form.	[13]
Double-Blind AB Test	A method of evaluating the differences between two inputs.	[13]
Steric	Referring to the spatial arrangement of atoms in a molecule	[13]
Structural Analog	A compound similar to another compound, but having a different form in respect to a certain component.	[15]

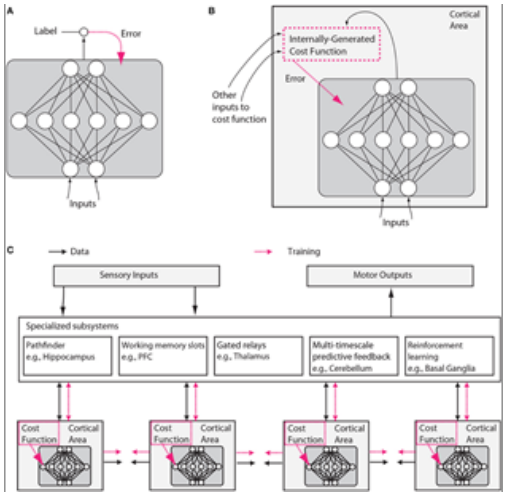
Article Notes

Article #X Notes: Template

Article notes should be on separate sheets

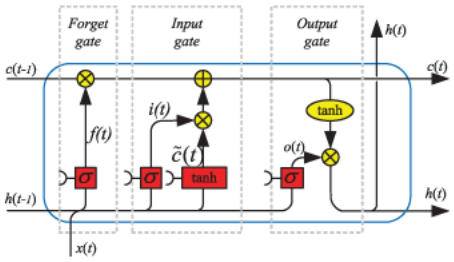
Source Title	**KEEP THIS BLANK AND USE AS A TEMPLATE**
Source citation (APA Format)	
Original URL	
Source type	
Keywords	
Summary of key points (include methodology)	
Research Question/Problem/ Need	
Important Figures	
Notes	
Cited references to follow up on	
Follow up Questions	

Article #1 Notes: Toward an integration of deep learning and neuroscience

Source Title	Toward an Integration of Deep Learning and Neuroscience
Source citation (APA Format)	Marblestone, A. H., Wayne, G., & Kording, K. P. (2016). Toward an Integration of Deep Learning and Neuroscience. *Frontiers in Computational Neuroscience*, 10. https://doi.org/10.3389/fncom.2016.00094
Original URL	https://www.frontiersin.org/articles/10.3389/fncom.2016.00094/full
Source type	Journal Article
Keywords	Neuroscience, Deep Learning
Summary of key points (include methodology)	This article attempts to draw biologically feasible connections between deep learning concepts and neuroscience. It states three key hypotheses about the brain: it optimizes cost functions, dynamically adapts cost functions based on need, and has distinct regions which are specialized to different learning tasks. These hypotheses are then justified from a biological and evolutionary perspective.
Research Question/Problem/ Need	How can the architecture of the biological brain enhance the performance of current deep learning models?
Important Figures	 A. Current model for supervised deep learning optimization.

	B. Proposed model for biologically feasible learning optimization. C. Proposed learning framework inspired by biological brain model.
Notes	- PREFRONTAL CORTEX for working memory. - Biological neurons have significantly more computational power than artificial neurons. - Internal bootstrapping process used to select cost functions.
Cited references to follow up on	Formal Theory of Creativity, Fun, and Intrinsic Motivation (1990–2010) - Selective learning in Reinforcement Learning A Biological Gradient Descent for Prediction Through a Combination of STDP and Homeostatic Plasticity - An implementation of a biologically inspired optimization algorithm.
Follow up Questions	1. How do neurochemicals affect the brain's function, and how do they fit in this deep learning-inspired scheme? 2. How do electrical signals in biological brains correspond to the numerical information passed through deep networks? 3. How does the brain synthesize outputs from each of the proposed regions into a continuous train of thought?

Article #2 Notes: Review of LSTMs and recurrent networks

Source Title	A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures
Source citation (APA Format)	Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Computation, 31*(7), 1235–1270. https://doi.org/10.1162/neco_a_01199
Original URL	https://direct.mit.edu/neco/article/31/7/1235/8500/A-Review-of-Recurrent-Neural-Networks-LSTM-Cells
Source type	Journal Article
Keywords	Recurrent, LSTM, Architecture
Summary of key points (include methodology)	This article highlights the technical details of the architecture of the LSTM, a neural network highly representative of the human brain's ability to commit information to long-term and short-term memory. The driving force behind the LSTM's memorization power comes from a "cell state", which is essentially a single vector passed throughout the training process representing the network's memory. Specific gates in the network control the flow of information into the cell state, enabling the model to decide what information is important to remember.
Research Question/Problem/ Need	How can a neural network store relevant information in memory like a biological brain?
Important Figures	 $f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f),$ $i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i),$ $\tilde{c}_t = \tanh(W_{\tilde{c}h}h_{t-1} + W_{\tilde{c}x}x_t + b_{\tilde{c}}),$ $c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t,$ $o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o),$ $h_t = o_t \cdot \tanh(c_t).$

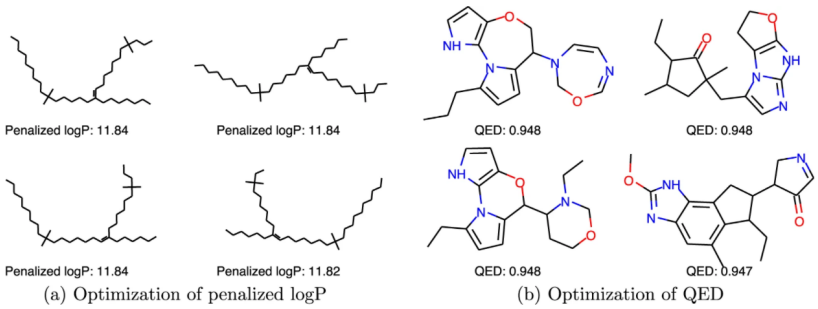
Notes	- c = cell state = vector - forget gate -> remove unnecessary information - input gate -> update cell state - output gate -> generate output based on input and cell state
Cited references to follow up on	An Empirical Exploration of Recurrent Network Architectures - Increasing bias has a positive effect on network performance. Training recurrent networks by evolino. - Evolutionary optimizers can improve model performance in some cases.
Follow up Questions	1. What is the reasoning behind specifically choosing sigmoid as an activation function for some parameters, and tanh as an activation for others? 2. Can a more complex data structure be used to hold the cell state-like, for example, a graph or MLP? 3. Can the LSTM deliberately remove information from the cell state once already committed to memory?

Article #3 Notes: Optimization of molecules via deep reinforcement learning

Source Title	Optimization of Molecules via Deep Reinforcement Learning
Source citation (APA Format)	Zhou, Z., Kearnes, S., Li, L., Zare, R. N., & Riley, P. (2019). Optimization of Molecules via Deep Reinforcement Learning. *Scientific Reports, 9*(1), 10752. https://doi.org/10.1038/s41598-019-47148-x
Original URL	https://www.nature.com/articles/s41598-019-47148-x#Sec9
Source type	Journal Article
Keywords	Lead Optimization, Reinforcement Learning
Summary of key points (include methodology)	The article details a deep reinforcement learning framework, MolDQN, which combines chemistry knowledge with state-of-the-art reinforcement learning techniques. The authors identify dataset bias during pre-training and chemical validity as two setbacks of molecule generation and introduce MolDQN as a solution. The framework formulates molecule modification as a Markov decision process, filtering out chemically infeasible changes to achieve 100% validity on generated data. The network also deliberately avoids methods that require pre-training. Additionally, the algorithm addresses the exploration vs. exploitation problem by using a bootstrapped ε-greedy policy, which selects the best action with probability (1-ε) and a random action with probability ε. ε is annealed linearly from 1 to 0.01 over the course of the training, allowing the model to essentially "focus" its learning as it becomes more knowledgeable. The model was tested on two targets--penalized logP and quantitative estimate of druglikeness, QED--and results found improved performance on multi-target generation compared to state of the art models, likely due to the lack of pre-training bias. However, the authors state that the current methods of evaluating performance of molecule optimizers are fundamentally flawed; they aren't representative of the experimental nature of desirable molecular properties in drug design.
Research Question/Problem/ Need	Problem: Traditional methods of lead optimization in drug discovery are time-consuming and costly.
Important Figures	

	Penalized logP				QED			
	1st	2nd	3rd	Validity	1st	2nd	3rd	Validity
random walk[a]	−3.99	−4.31	−4.37	100%	0.64	0.56	0.56	100%
greedy[b]	11.41	—	—	100%	0.39	—	—	100%
ϵ-greedy, $\epsilon = 0.1$[b]	11.64	11.40	11.40	100%	0.914	0.910	0.906	100%
JT-VAE[c]	5.30	4.93	4.49	100%	0.925	0.911	0.910	100%
ORGAN[c]	3.63	3.49	3.44	0.4%	0.896	0.824	0.820	2.2%
GCPN[c]	7.98	7.85	7.80	100%	0.948	0.947	0.946	100%
MolDQN-naïve	11.51	11.51	11.50	100%	0.934	0.931	0.930	100%
MolDQN-bootstrap	11.84	11.84	11.82	100%	0.948	0.944	0.943	100%
MolDQN-twosteps	—	—	—	—	0.948	0.948	0.948	100%

Table of performance measures of different deep learning approaches to molecular optimization.



(a) Optimization of penalized logP (b) Optimization of QED

Resulting model outputs when optimizing for logP (a) and QED (b), respectively.

Notes

Abstract

- Proposed model: MolDQN, based on reinforcement learning.
- Pre-training = possible dataset bias. Should be avoided (?)
- Equal or better performance on benchmark molecule optimization tasks
- Extended for multi-objective learning to maximize drug-likeness

Introduction

- Previous approaches
  - Autoencoder-based
    - molecule to latent space (optimization) to molecule
    - Generation of SMILES strings
      - Poor chemical validity (solved with grammar constraints)
    - GANs
  - Reinforcement Learning
    - Used with a string generator on SMILES
    - GCPN achieved 100% chemical validity but requires pre-training (You et al.)
- MolDQN
  - Driving forces
    - Deep Q-Learning
    - Markov Decision Process
  - Different from previous work:
    - Value function learning instead of policy gradient method.
    - No pre-training
    - Designed for multi-objective learning

Methods

- Modification of a molecule in 3 steps:
  - Atom addition
  - Bond addition
  - Bond removal
- Molecules generated only depend on current state and the modification made = MDP.
- MDP
  - Explicit limit to the number of steps allowed.
  - Remove chemically invalid actions from the decision pool.

	- Atoms/bonds can be removed/added.
	- Value function is approximated using only the expected reward at the current time step due to being a MDP.
	- Math
	  - m: molecule
	  - t: current time step
	  - s: state space (m, t)
	  - a: action
	  - Q(s, a): predicts the reward of taking an action *a* on the state *s*
	  - Deep Q-learning used to approximate Q function.
	  - π: policy (not used in value function method)
	- Scalarized reward framework used for multi-objective learning
	- Exploitation vs Exploration
	  - Exploitation: always choosing the best reward = model learns nothing about other states (overfitting?)
	  - Exploration: random action selection = no reward.
	  - ε-greedy modification (bootstrapped-DQN) to solve issue
	- Huber loss.
	- Adam optimizer.
	Results
	- Tested on penalized logP and QED.
	- Optimizing for properties decreases diversity of generated molecules.
	- MolDQN is more effective at multi-objective learning.
	
	Conclusion
	- Future work:
	  - Hyperparameter optimization
	  - Using different Q-function approximators.
	- Need for a better method of evaluating generated molecules;
	  - Experimentally verifiable properties (therapeutically relevant) are better than computed properties.

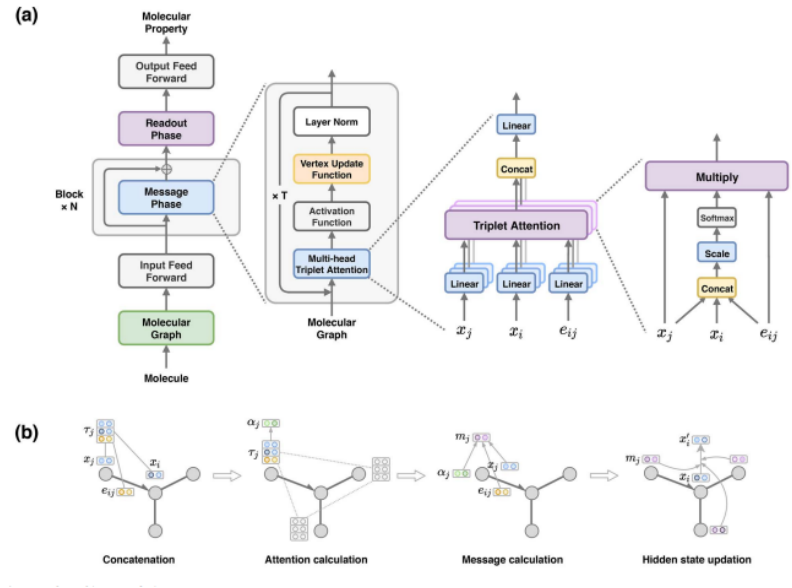
Cited references to follow up on	Deep Q-Networks: Mnih, V. et al. Human-level control through deep reinforcement learning. Nature 518, 529 (2015). Double Q-learning: Van Hasselt, H., Guez, A. & Silver, D. Deep reinforcement learning with double Q-Learning. In AAAI 2, 5 (2016). MPNN Q-function approximator: Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O. & Dahl, G. E. Neural message passing for quantum chemistry. arXiv preprint arXiv:1704.01212 (2017). Evaluating Molecule Generators in Chemistry: Benhenda, M. Chemgan challenge for drug discovery: can AI reproduce natural chemical diversity? arXiv preprint arXiv:1708.08227 (2017).

	Graph Convolutional Policy Network with 100% Validity: You, J., Liu, B., Ying, R., Pande, V. & Leskovec, J. Graph convolutional policy network for goal-directed molecular graph generation. arXiv preprint arXiv:1806.02473 (2018).
Follow up Questions	- Would a Graph Convolutional Network work better for Q-function approximation, as opposed to a deep fully connected layer? - Can the network **ensure** the molecule's binding affinity to a target is preserved? - Can the network be taught optimization rules explicitly? - Can the model be stopped before the terminal time-step if the molecule satisfies some criteria? - Are some complex modifications not considered by the network due to its restriction of one valid modification at a time? E.g. a series of modifications that result in a valid molecule but is invalid in transitionary states.

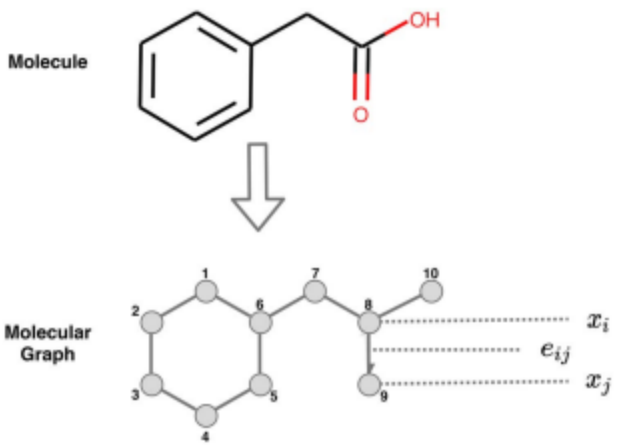
Article #4 Notes: [✧] TrimNet

Source Title	TrimNet: learning molecular representation from triplet messages for biomedicine.
Source citation (APA Format)	Li, P., Li, Y., Hsieh, C.-Y., Zhang, S., Liu, X., Liu, H., Song, S., & Yao, X. (2021). TrimNet: Learning molecular representation from triplet messages for biomedicine. *Briefings in Bioinformatics, 22*(4), bbaa266. https://doi.org/10.1093/bib/bbaa266
Original URL	https://academic-oup-com.ezpv7-web-p-u01.wpi.edu/bib/article/22/4/bbaa266/5955940
Source type	Journal Article
Keywords	Molecules AND Graph Neural Networks
Summary of key points (include methodology)	This paper details a novel method of processing molecular graphs that allows for significant reduction in the number of model parameters while maintaining state-of-the-art performance. Standard graph neural networks transform all nodes of the graph by its adjacency matrix and edge attributes, resulting in the creation of many irrelevant parameters. In addition, messages can only be passed to a node's adjacent neighbors, which is not representative of how molecular bonds work. The proposed network uses a triplet attention mechanism to process the nodes and edges of the graph. This method circumvents the matrix mapping of edge information, which removes the irrelevant parameters. The model was tested on various quantum chemical and molecular datasets, and it was found that the model could achieve state of the art performance on a variety of different tasks. Additionally, its use of an attention mechanism increases model interpretability, connecting its behavior to known chemical rules. TrimNet reduced the number of parameters used by up to 1/30 of the previous state of the art, but it was unable to significantly improve computational complexity (training time) despite this.
Research Question/Problem/ Need	How can we efficiently represent and process molecular information?

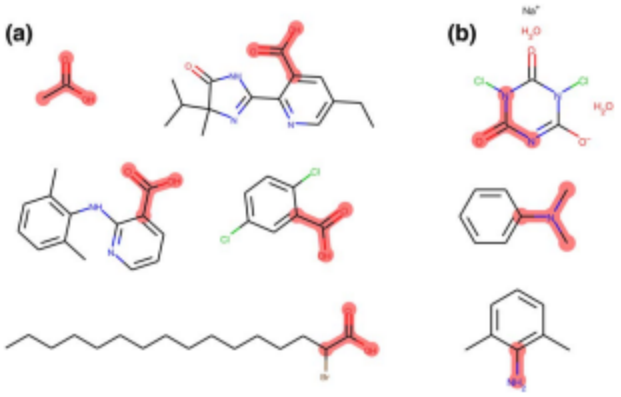
Important Figures	



A representation of the framework used in the paper.



Good diagram of graph representation of a molecule.



	Graph attention in TrimNet focuses on toxic functional groups.

Notes	Abstract
	- Limitation of current graph-based methods:
	  - Large-scale parameters
	  - Insufficient bond information extraction
	Methodology
	- Evaluated on:
	  - Quantum Properties
	  - Physiology
	  - Bioactivity
	  - CPI
	- Benchmark datasets: QM9, MUV, HIV, BACE, BBBP, Tox21, ToxCast, SIDER, ClinTox, Human & C. elegan.
	- Message Phase:
	(a) $\tau_{i,j}^{t+1} = \text{LeakyReLU}(\boldsymbol{u}^T[\boldsymbol{W}_h \boldsymbol{h}_i^t \| \boldsymbol{W}_e \boldsymbol{e}_{ij}^t \| \boldsymbol{W}_h \boldsymbol{h}_j^t])$,
	(b) $\alpha_{i,j}^{t+1} = \text{softmax}(\tau_{i,j}^{t+1}) = \frac{e^{\tau_{i,j}^{t+1}}}{\sum_{j \in \mathcal{N}_i} e^{\tau_{i,j}^{t+1}}}$,
	(c) $\boldsymbol{m}_i^{t+1} = \sum_{j \in \mathcal{N}_i} \alpha_{i,j}^{t+1} \odot \boldsymbol{W}_h h_j^t \odot \boldsymbol{W}_e \boldsymbol{e}_{ij}^t$,
	(d) $\boldsymbol{m}_i^{t+1} = \|_k^K \sum_{j \in \mathcal{N}_i} \alpha_{i,j}^{t+1,k} \odot \boldsymbol{W}_h^k h_j^t \odot \boldsymbol{W}_e^k \boldsymbol{e}_{ij}^t$,
	(e) $h_i^{t+1} = LN(\textbf{GRU}(h_i^t, m_i^{t+1}))$.
	- Readout Phase:
	  - Implements Set2Set networks for a graph-level embedding.

$$q_t = \text{LSTM}(q^*_{t-1}),$$

$$\alpha_{i,t} = \text{softmax}(h_i^T q_t),$$

$$r_t = \sum_{i=1}^{N} \alpha_{i,t} h_i^T,$$

$$q^*_t = q_t || r_t.$$

- q* is fed into feedforward NN for final prediction.

- Training:
  - Varying loss based on task
  - Gradient descent & error backpropagation using Adam optimizer
  - Grid search for hyperparameter tuning (!?!?!)

Results

- QM9
  - New state of the art results on 12 of 12 tested quantum mechanical properties
  - Decrease in MAE by more than 80% on 6 QM properties.
  - Incorporating edge information (distances, angles, and electronic features) further increased performance on 9 of 12 tasks.
- Bioactivity and Physiology
  - Increased performance on 6 of 8 tasks.
- CPI
  - Replaced the GNN portion of Tsubaki et al. GPI model with TrimNet.
  - Result: “more accurate.”
- TrimNet uses 1/30 of attentive FP parameters in quantum property prediction.
  - However, training time remained relatively constant with previous models = no improvement in computational complexity.
- Better interpretability;
  - Greater attention to atomic groups likely to cause toxicity in toxicity prediction tasks via the attention mechanism.
- Layer normalization is a key addition to MPNN architecture.
- Greater scalability with model depth
  - Positive trend of improvement observed on half of tasks when depth is increased to 10.

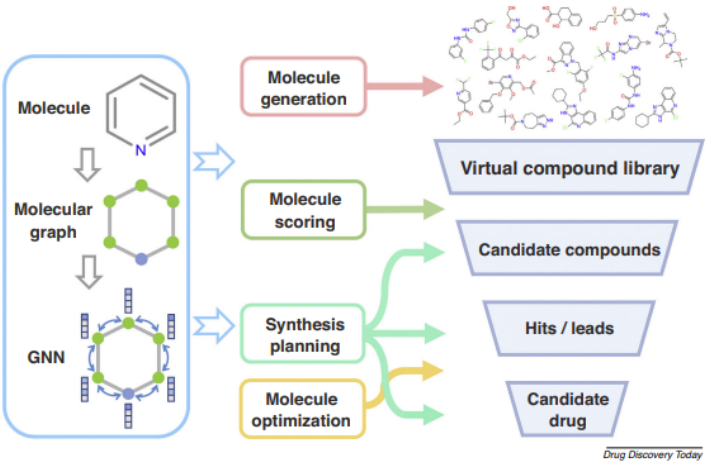
Cited references to follow up on	Over-Smoothing of Graph Networks: Chen D, Lin Y, Li W, Et al. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. In: AAAI Conference on Artificial Intelligence, New York, Palo Alto, CA, USA: AAAI Press, 2020, 3438–3445.

	Father Paper of Graph Networks for Drug Design: Gilmer J, Schoenholz SS, Riley P, et al. Neural message passing for quantum chemistry. In :International Conference on Machine Learning, Sydney, Australia, 2017, 1263–1272. ICML Press. Graph Attention and Attentive FP: Xiong Z, Wang D, Liu X, et al. Pushing the boundaries of molecular representation for drug discovery with graph attention mechanism. J Med Chem 2019. page acs.jmedchem.9b00959. Human and C. elegans Benchmarks: Liu H, Sun J, Guan J, et al. Improving compound-protein interaction prediction by building up highly credible negative samples. Bioinformatics 2015; 31(12): i221–9. CPI Prediction Framework (Modified by TrimNet): Tsubaki M, Tomii K, Sese J. Compound-protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. Bioinformatics 2019; 35(2): 309–18. Set2Set for graph-level embedding: Vinyals O, Bengio S, Kudlur M et al. Order matters: sequence to sequence for sets. In: International Conference on Learning Representations, San Juan, Puerto Rico, 2016. ICLR Press.
Follow up Questions	- Why is the parameter gap significantly higher in Tox21-Toxcast compared to BACE-HIV? - Are the reported results amplified by the TrimNet's use of grid search for hyperparameter tuning? - Why is it necessary for edge and node features to be embedded to the same dimensionality in expression a)? - What is the rationale behind using recurrent networks in expression e) and the readout layer? - Layer normalization vs batch norm vs instance norm?

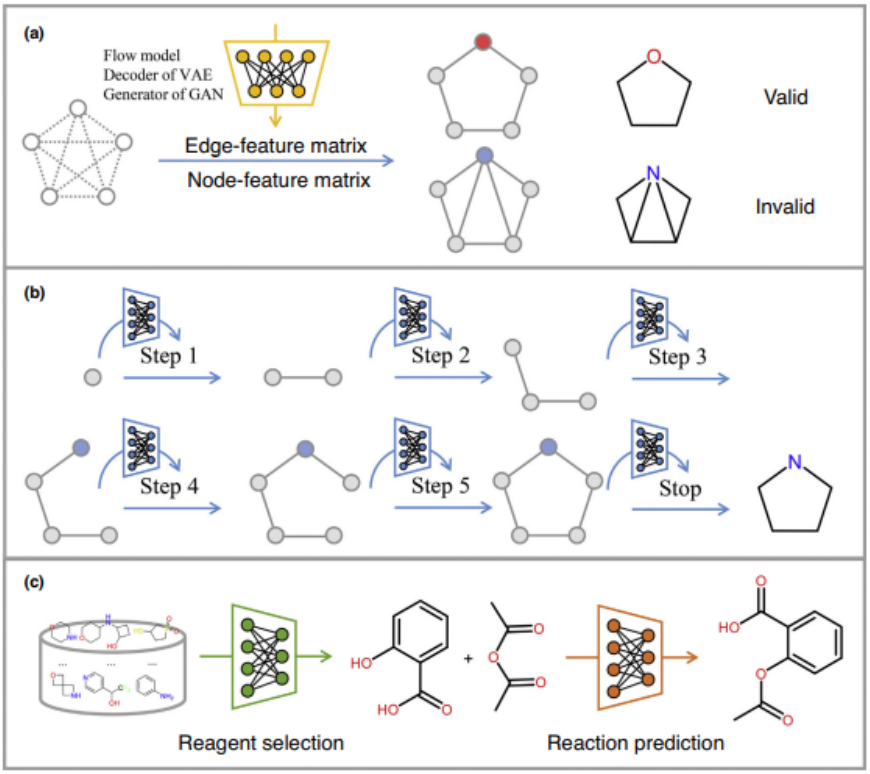
Article #5 Notes: [REVIEW] Graph neural networks for automated *de novo* drug design

Source Title	Graph neural networks for automated de novo drug design.
Source citation (APA Format)	Xiong, J., Xiong, Z., Chen, K., Jiang, H., & Zheng, M. (2021). Graph neural networks for automated de novo drug design. *Drug Discovery Today, 26*(6), 1382–1393. https://doi.org/10.1016/j.drudis.2021.02.011
Original URL	https://www-sciencedirect-com.ezpv7-web-p-u01.wpi.edu/science/article/pii/S1359644621000787?via%3Dihub
Source type	Secondary Literature (Review)
Keywords	Graph Neural Networks AND Drug Design
Summary of key points (include methodology)	This article gives a broad review of recent research in the field of deep learning for drug discovery. It outlines key terminology, pioneering research, important benchmarks, and current knowledge gaps. More specifically, the article defines molecule generation, molecule scoring, and molecule optimization as the three key elements of drug discovery that can be addressed computationally via deep learning. In the field of molecule generation, generated molecules are assessed based on their validity, uniqueness, and novelty. However, these criteria do not account for molecular synthesis. Molecule generators have not been able to produce remarkable results due to the lack of a synthesis route for the generated molecule. Notably, the paper highlights the current knowledge gap in computational retrosynthesis as an important research direction. Current deep learning models for retrosynthesis are single-step and do not perform well enough to be of use in any sort of practical application. Improving their performance and expanding them to cover full-route retrosynthesis would be a valuable next step.
Research Question/Problem/ Need	What are the current gaps in the application of deep learning in drug discovery?

Important Figures	



The role of GNNs in automated drug discovery.



Non-autoregressive (a) vs autoregressive (b) vs virtual reaction ©

Summary of models for reaction and retrosynthesis prediction that have been tested correspondingly on the USPTO and USPTO-50K data sets

Class	Model name	Sequence based	Graph based	Template based	Top accuracy (%)	Year	Refs
Reaction prediction	WLDN		√		79.6	2017	[69]
	S2S	√			80.3	2018	[87]
	GTPN		√		82.4	2019	[72]
	WLDN5		√		85.6	2019	[70]
	Schwaller et al	√			90.4	2019	[88]
	Qian et al.		√		>90	2020	[71]
Retrosynthesis prediction	Seq2Seq	√			37.4	2017	[89]
	Retrosim			√	52.9	2017	[90]
	Karpov et al.	√			42.7	2019	[91]
	Lee et al.	√			43.8	2019	[92]
	SCROP	√			59.0	2019	[93]
	GLN		√	√	64.2	2019	[74]
	RetroXpert	√	√		70.4	2020	[77]
	Lin et al.	√			54.6	2020	[94]
	GET-LT1	√	√		57.4	2020	[95]
	G2Gs		√		61.0	2020	[78]
	GRAPHRETRO		√		67.8	2020	[79]

Table of previous models and performance. For future reference.

Notes

Introduction

- Deep learning mainly applied to three aspects of drug design:
  - Molecule generation
  - Molecule scoring
  - Molecule optimization
- GNNs for molecular learning classified in 4 categories:
  - Graph-level prediction
    - For molecular property prediction
  - Node/edge-level prediction
    - For reaction and retrosynthesis
  - Graph implicit representation
    - For one-step molecule generation
  - Transforming rules of graph
    - For iterative molecular generation/optimization and reaction/retrosynthesis prediction.

Molecule scoring

- Scoring based on drug-likeness and pharmacological activity
- Ligand-based scoring with GNN
  - Doesn't need receptor information
  - Creates differentiable fingerprints
  - Better results than Morgan fingerprints on solubility prediction
  - Virtual supernode sometimes used in graph-level embeddings
  - Attention mechanisms applied to GNN increases performance and interpretability
- Receptor-based scoring with GNN
  - Receptor protein encoding methods:
    - Protein sequence
    - Protein pocket structure
    - Protein-ligand complex structure
  - Attention mechanisms enable the network to learn noncovalent interactions between protein and compound.
- Lack of high quality datasets, resulting in dataset bias.

Molecule generation

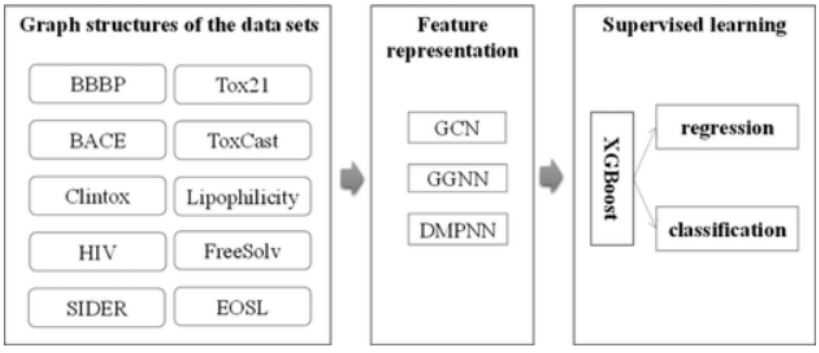
- Autoregressive, non-autoregressive, and reversible flow models.

	- Evaluated with:
	  - Validity
	  - Novelty
	  - Uniqueness
	- Tradeoff between evaluated metrics
	- Synthesis issue: generated molecules often can't be synthesized
	  - Combining generative models with synthesis planning models is a valuable next step
	- Need for new methods of evaluating the quality of generated molecules.
	Synthesis planning
	- ILP allows for manual programming of chemistry constraints.
	- New state of the art: RetroXpert
	  - Bond disconnection prediction
	  - Reactant prediction
	- Current state of the art is not strong enough to be helpful independently.
	- Need for extending single-step retrosynthesis to full route planning
	Conclusion
	- Need for deeper GNNs that solve the over-smoothing problem (deeper stacks)

Cited references to follow up on	Preservation of Spatial Information in Graph Convolutions: Coley, C.W. et al. (2017) Convolutional embedding of attributed molecular graphs for physical property prediction. J. Chem. Inf. Model. 57, 1757–1772 Self-Attention Mechanism: Tang, B. et al. (2020) A self-attention based message passing neural network for predicting molecular lipophilicity and aqueous solubility. J. Cheminf. 12, 1–9 Directional Message Passing in Molecular Graphs: Yang, K. et al. (2019) Analysing learned molecular representations for property prediction. J. Chem. Inf. Model. 59, 3370–3388 Multitask Learning for CPI: Li, S. et al. (2020) MONN: a multi-objective neural network for predicting compound-protein interactions and affinities. Cell Syst. 10, 308–322 DTI Prediction via Protein-Ligand Binding Poses: Lim, J. et al. (2019) Predicting drug–target interaction using a novel graph neural network with 3D structure-embedded graph representation. J. Chem. Inf. Model. 59, 3981–3988 MoleculeNet benchmark: Wu, Z. et al. (2018) MoleculeNet: a benchmark for molecular machine learning. Chem. Sci. 9, 513–530 Federated Learning: Li, T. et al. (2020) Federated learning: challenges, methods, and future directions. IEEE Signal Proc Mag 37, 50–60

	Flow-based Generative Model (100% validity): Madhawa, K. et al. (2019) GraphNVP: an invertible flow model for generating molecular graphs. arXiv 2019, 1905.11600 MolMP for Autoregressive Generation: Li, Y. et al. (2018) Multi-objective de novo drug design with conditional graph generative model. J. Cheminf. 10, 33 Autoregressive Flow-based Molecule Generation: Shi, C. et al. (2020) GraphAF: a flow-based autoregressive model for molecular graph generation. arXiv 2020, 2001.09382 JT-VAE: Jin, W. et al. (2018) Learning multimodal graph-to-graph translation for molecular optimization. arXiv 1812.01070 Molecule Chef: Bradshaw, J. et al. (2019) A model to search for synthesizable molecules. Adv. Neural Inf. Process. Syst. 33, 7937–7949 Turing Test: Bush, J.T. et al. (2020) A Turing test for molecular generators. J. Med. Chem. 63, 11964–11971 Reaction Prediction within GNN and ILP: Qian, W.W. et al. (2020) Integrating deep neural networks and symbolic inference for organic reactivity prediction. ChemRxiv 2020, 11659563 Template-based Retrosynthesis with GNN: Dai, H. et al. (2019) Retrosynthesis prediction with conditional graph logic network. Adv. Neural Inf. Process. Syst. 33, 8872–8882 Deeper GNNs with dynamically changing edges?: Li, G. et al. (2019) DeepGCNs: can GNNs go as deep as CNNs? In Proceedings of IEEE/ CVF International Conference on Computer Vision, IEEE. pp. 9267–9276
Follow up Questions	- Do autoregressive or non-autoregressive generative models offer more scalability? - How is retrosynthesis performance evaluated? - Can dataset bias be addressed without having to generate new data? - What properties of Li et al. (2019)'s model enable the GNN to stack much deeper layers?

Article #6 Notes: XGraphBoost

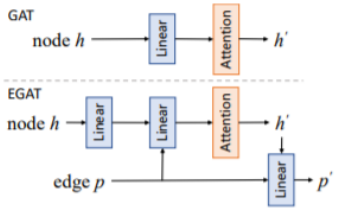
Source Title	XGraphBoost: Extracting Graph Neural Network-Based Feature for a Better Prediction of Molecular Properties
Source citation (APA Format)	Deng, D., Chen, X., Zhang, R., Lei, Z., Wang, X., & Zhou, F. (2021). XGraphBoost: Extracting Graph Neural Network-Based Features for a Better Prediction of Molecular Properties. *Journal of Chemical Information and Modeling, 61*(6), 2697–2705. https://doi.org/10.1021/acs.jcim.0c01489
Original URL	https://pubs.acs.org/doi/10.1021/acs.jcim.0c01489
Source type	Research Paper
Keywords	Graph Neural Network, Molecules
Summary of key points (include methodology)	In the field of deep learning drug discovery, an efficient way of processing molecules is necessary. Previous machine learning methods for molecule processing require the input of hand-crafted features which must be manually calculated by researchers. The article identifies graph neural networks as a promising solution to molecular representation without the use of these calculated features, and it predicts that integrating XGBoost as the prediction layer for the network will enhance its performance. To test this hypothesis, the authors took three graph networks, GCN, GGNN, and DMPNN, and replaced their output layers with XGBoost. The modified models were tested on 10 datasets part of MoleculeNet, and the study found that XGBoost improved the performance of all 3 models. DMPNN outperformed
Research Question/Problem/ Need	Can Graph Neural Network variants be improved by replacing their prediction layers with a more specialized algorithm?
Important Figures	 The procedure outlined in the paper. 10 datasets were tested in the addition of XGBoost to graph processing models.
Notes	Abstract:

	- Machine Learning (not DL) approaches to molecular property prediction are heavily reliant on hand-crafted features
	- GNNs can automatically create embeddings for molecules as graphs.
	- XGraphBoost uses XGBoost classifier
	- Github: https://github.com/chenxiaowei-vincent/XGraphBoost.git
	Introduction:
	- Paper defines a node as a vector containing one-hot encodings for the following information:
	  - Atomic type, atomic element, additional H atoms, number of valence, aromatic properties.
	- XGraphBoost: seems to be an ensemble learning method using GCN, GGNN, and DMPNN.
	
	Performance
	- Evaluated on classification and regression problems.
	  - Binary classification: Specificity, Sensitivity, Accuracy, and AUC
	  - Regression: RMSE error.
	- Datasets: (Similar to TrimNet)
	  - ESOL
	  - FreeSolv
	  - Lipophilicity
	  - BACE
	  - BBBP
	  - Clintox
	  - HIV
	  - Tox21
	  - ToxCast
	  - SIDER
	- DMPNN achieved best performance on 9 of 10 datasets
	- DMPNN outperforms Morgan fingerprints on 10 of 10 datasets.
	- XGBoost increased performance of 3 of 3 models.
	  - GCN performance increased the most.
	- Performance increased with hyperparameter tuning.

Cited references to follow up on	GRU-based Graph Neural Network: Ruiz, L., Gama, F., & Ribeiro, A. (2019). Gated graph convolutional recurrent neural networks. 2019 27th European Signal Processing Conference (EUSIPCO). https://doi.org/10.23919/eusipco.2019.8902995 ChEMBL dataset: Mayr, A., Klambauer, G., Unterthiner, T., Steijaert, M., Wegner, J. K., Ceulemans, H., Clevert, D.-A., & Hochreiter, S. (2018). Large-scale comparison of machine learning methods for drug target prediction ON CHEMBL. Chemical Science, 9(24), 5441–5451. https://doi.org/10.1039/c8sc00148k Self-attention SMILES:

	Zheng, S., Yan, X., Yang, Y., & Xu, J. (2019). Identifying structure–property relationships through smiles syntax analysis with self-attention mechanism. Journal of Chemical Information and Modeling, 59(2), 914–923. https://doi.org/10.1021/acs.jcim.8b00803
Follow up Questions	- What property of DMPNN makes it perform consistently better than GCN? - Why did GCN have the highest overall performance increase with XGBoost? - Are Morgan fingerprints what the paper referred to as “hand-crafted features”? If so, this is misleading. - Can XGBoost be extended to replace the output layers of other graph-processing neural networks? (Try TrimNet, RetroXPert)

Article #7 Notes: RetroXpert

Source Title	RetroXpert: Decompose Retrosynthesis Prediction Like A Chemist
Source citation (APA Format)	Yan, C., Ding, Q., Zhao, P., Zheng, S., Yang, J., Yu, Y., & Huang, J. (2020). RetroXpert: Decompose Retrosynthesis Prediction like a Chemist. *ArXiv:2011.02893 [Cs, q-Bio]*. http://arxiv.org/abs/2011.02893
Original URL	https://arxiv.org/pdf/2011.02893.pdf
Source type	Research Paper
Keywords	Retrosynthesis, Deep Learning
Summary of key points (include methodology)	Current methods of computational synthesis planning are inefficient, inaccurate, and uninterpretable. This paper attempts to address the issue by developing a deep learning approach. It formulates a retrosynthesis stage as a two step process: prediction of the reaction center/synthons, and prediction of the reactants based on the synthons. On both tasks, providing the network with chemical reaction information significantly increased performance. Unfortunately, this is not practical in a broader application. Without reaction type data, the researchers found that using edge-enhanced attention with the model could boost the performance, though not to the same extent as reaction data. For future research, the researchers state that a better method of evaluating retrosynthesis model performance is in need, since retrosynthesis can have multiple pathways that are optimal in different situations.
Research Question/Problem/ Need	How can computational retrosynthesis be improved through the use of deep learning?
Important Figures	$$z_i = \mathbf{W}h_i,$$ $$c_{i,j} = \mathrm{LeakyReLU}(\mathbf{a}^T[z_i \| z_j \| p_{i,j}]),$$ $$\alpha_{i,j} = \frac{\exp(c_{i,j})}{\sum_{k\in\mathcal{N}_i}\exp(c_{i,k})},$$ $$h'_i = \sigma\Big(\sum_{j\in\mathcal{N}_i}\alpha_{i,j}\mathbf{U}[z_j \| p_{i,j}]\Big),$$ $$p'_{i,j} = \mathbf{V}[h'_i \| h'_j \| p_{i,j}],$$ (1)  Figure 2: Embedding computation flows of GAT and the proposed EGAT. Model architecture of RetroXPert.

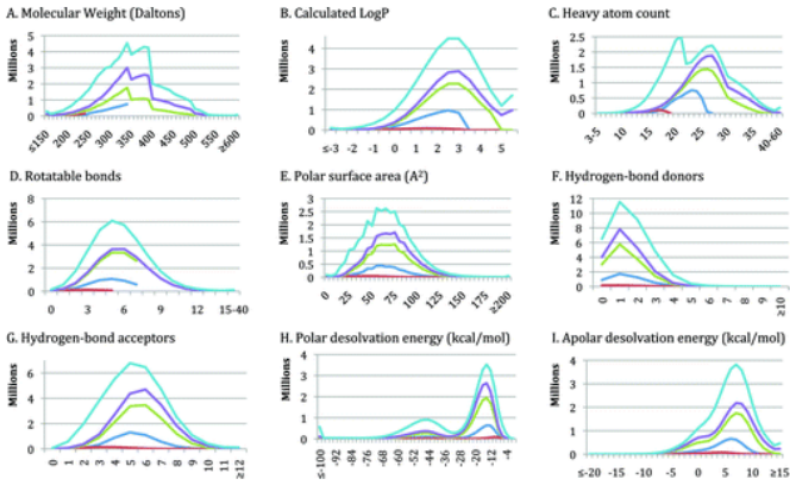
Notes	Abstract
	- Automated retrosynthesis is cumbersome and lacks interpretability
	- 2 step process:
	  - Identify potential reaction center and generate synthons
	  - Generate reactants associated with synthons via reactant generation

	model
	- Model provides interpretability and outperforms state of the art.
	
	Introduction
	- Huge size of search space for retro: 10^7 reactions and compounds.
	- Molecules have multiple synthetic pathways, feasibility of a route is determined by multiple factors.
	- Template based methods
	  - Synthia relies on hand-coded rules, and cannot generate novel reactions.
	- Template-free methods:
	  - Lack interpretability
	  - Don't consider chemistry domain knowledge
	  - Slow model convergence.
	- Use of edge-enhanced graph attention networks.
	- Use of the reactant generation network to predict reactants given synthon.
	- Use of data augmentation.
	Methodology
	- Trained on USPTO 50k and USPTO full datasets
	- EGAT trained first on an auxiliary task, increasing model performance.
	  - Predict # of disconnected bonds. 99% acc
	- EGAT 86% acc w/ reactions, 64.9% w/o.
	- Data AUG is effective.
	Conclusion
	- Need for reasonable evaluation metrics for retrosynthesis prediction.

Cited references to follow up on	USPTO Dataset: Schneider, N., Stiefl, N., & Landrum, G. A. (2016). What's what: The (nearly) definitive guide to reaction role assignment. Journal of Chemical Information and Modeling, 56(12), 2336–2346. https://doi.org/10.1021/acs.jcim.6b00564 Template-Free Retrosynthesis Planning: Lin, K., Pei, J., Lai, L., & Xu, Y. (2019). Automatic retrosynthetic Pathway planning Using Template-free Models. https://doi.org/10.26434/chemrxiv.8168354.v1 Retrosynthesis Reaction Prediction: Zheng, S., Rao, J., Zhang, Z., Xu, J., & Yang, Y. (2019). Predicting retrosynthetic reaction using Self-corrected Transformer neural networks. https://doi.org/10.26434/chemrxiv.8427776.v1 Beam Search: Tillmann, C., & Ney, H. (2003). Word reordering and a dynamic programming beam search algorithm for statistical machine translation. Computational Linguistics, 29(1), 97–133. https://doi.org/10.1162/089120103321337458
Follow up Questions	- Can TrimNet be used to replace the graph attention mechanism in their model architecture?

	- How are the labels represented in USPTO datasets? - Why doesn't data augmentation work for the prediction of reactants? - How will retrosynthesis prediction fit in the broader scheme of generating more feasible drug molecules via deep learning?

Article #8 Notes: ZINC Dataset

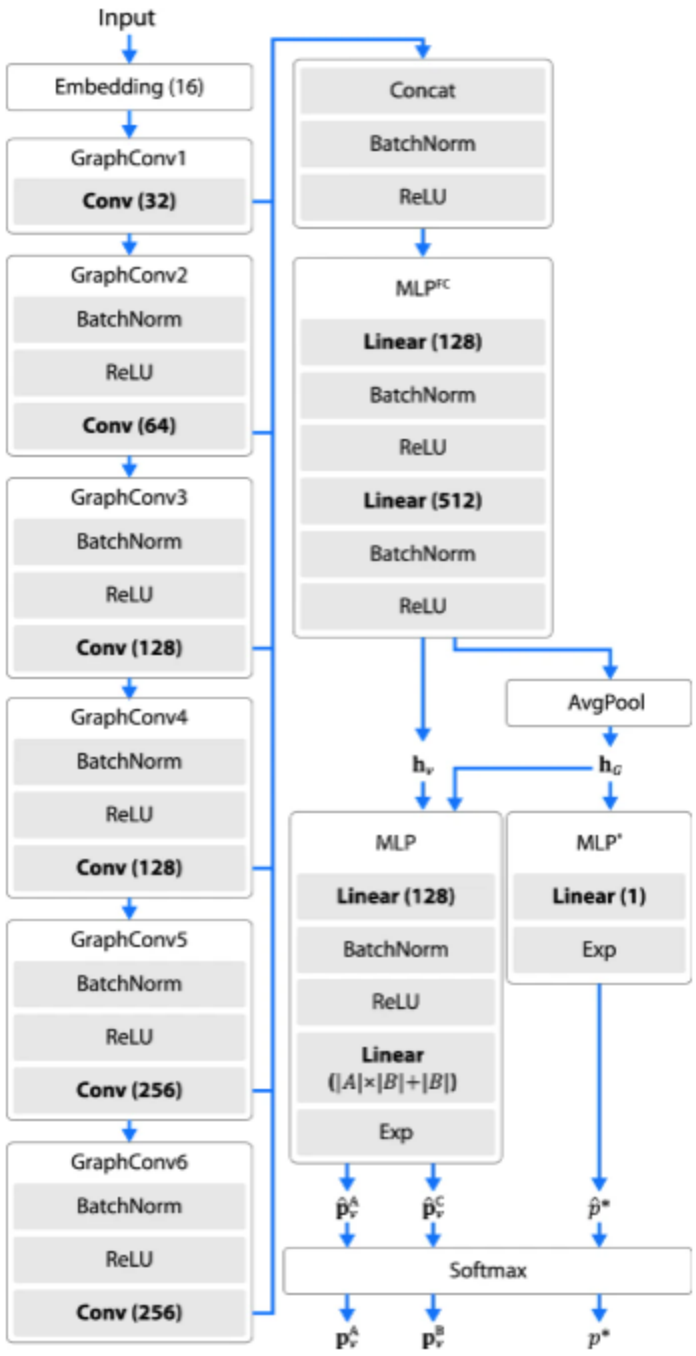
Source Title	ZINC: A Free Tool to Discover Chemistry for Biology
Source citation (APA Format)	Irwin, J. J., Sterling, T., Mysinger, M. M., Bolstad, E. S., & Coleman, R. G. (2012). ZINC: A free tool to discover chemistry for biology. *Journal of Chemical Information and Modeling, 52*(7), 1757-1768. https://doi.org/10.1021/ci3001277
Original URL	https://pubs.acs.org/doi/10.1021/ci3001277
Source type	Journal Article
Keywords	Machine Learning, Chemistry Dataset
Summary of key points (include methodology)	Previous molecule datasets had varying drawbacks that limited their application to a diverse pool of users. For example, some datasets contain relevant information but fail to account for 'metadata' such as the purchasability of the given molecules. The ZINC dataset attempts to address these drawbacks with a new application that is frequently updated with new molecules and is able to create subsets of the chemical space in order to create datasets relevant to a specific task. The dataset utilizes modern algorithms to index and group data
Research Question/Problem/ Need	The dataset described in the paper was created in response to a need for a chemistry tool that stores information about a large quantity of molecules and is capable of subclassing the molecules based on this information.
Important Figures	 The distribution of molecules with the indicated property, with different colors representing different standard subsets of the ZINC dataset.
Notes	- ZINC: 20M commercially available molecules - Search by structure, biological activity, physical property, vendor, catalog number,

	name, and CAS number. - zinc.docking.org - ZINC: intended to be a research tool to find chemical metter for biological targets. - Contains 100+ vendors and 20+ databases - Using relevant protonated an tautomeric forms for molecular docking is important - pH dependent representations of the screening library is important. - ChEMBL, PubChem, DrugBank, BindingDB, and TCM@Taiwan databases lack focus on docking and purchasability - ChemSpider has no support for organization into discovery-oriented subsets. - E-commerce sites for compounds do not have focus on relevant 3D structures and subsets for docking. Methods - Takes catalogs and converts to isometric SMILES - Generate up to 4 stereoisomers for stereochemically ambiguous molecules. - Trial 3D structure generated with Corina for a canonical conformation. - Generated in 4 pH ranges with Epik (Schrodinger). Calculation of Physical Properties - Mib for logP and polar surface area. - AMSOL for polar and apolar desolvation energies. Clustering - Sort ligands by increasing molecular weight. - SUBSET 1.0 algorithm to select compounds that differ by > Tanimoto cutoff w/ axonpath fingerprints. - Representatives can be said to cover the chemical space of the subset. Results - Molecules represented in multiple 3D forms and organized into 4 pH ranges. - Molecules in each subset have physical properties suitable for drug-discovery. Rest of the paper gives examples of how to use the dataset.
Cited references to follow up on	N/A
Follow up Questions	- Can the Zinc dataset be connected to a deep learning model without slowing down performance? - How does the dataset measure purchasability? - Can the purchasability metric be used with retrosynthesis prediction to score the efficacy of a given set of reactants? - Can custom subsets be made based on a collection of multiple desired properties?

Article #9 Notes: Multi-objective Conditional Graph Generative Model

Source Title	Multi-objective de novo drug design with conditional graph generative model
Source citation (APA Format)	Li, Y., Zhang, L., & Liu, Z. (2018). Multi-objective de novo drug design with conditional graph generative model. *Journal Of Cheminformatics, 10*(1). https://doi.org/10.1186/s13321-018-0287-6
Original URL	https://jcheminf.biomedcentral.com/articles/10.1186/s13321-018-0287-6
Source type	Journal Article
Keywords	Multi Objective Optimization, Deep Learning, De novo drug design
Summary of key points (include methodology)	Deep graph generative models have recently shown promising results when applied to drug design, but current models are too general. This paper develops a new framework for graph generation that is tailored toward molecules. It does so by using a conditional graph model which constructs molecules in an iterative manner. The action taken at each step is determined by a probability model, which factors the current state of the molecule, as well as any constraints defined by the user. This enables the model to both account for chemical validity and optimize for chemical properties simultaneously.
Research Question/Problem/ Need	Current deep learning approaches to de novo drug design are designed too generally and lack support for multi-objective optimization.

Important Figures	



Architecture of the proposed model.

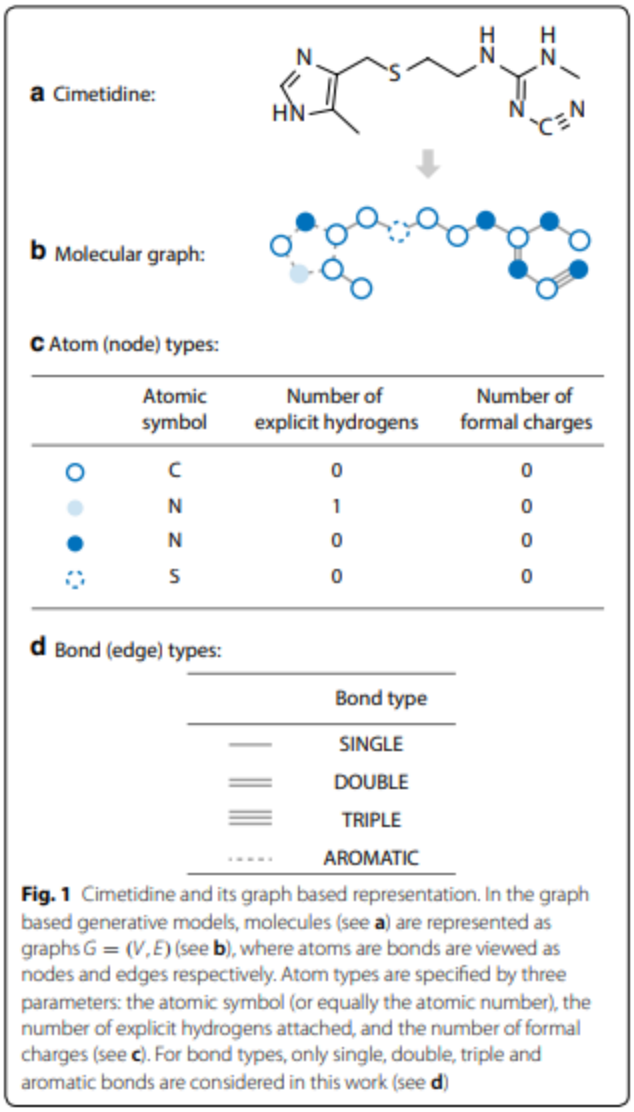


Fig. 1 Cimetidine and its graph based representation. In the graph based generative models, molecules (see **a**) are represented as graphs $G = (V, E)$ (see **b**), where atoms are bonds are viewed as nodes and edges respectively. Atom types are specified by three parameters: the atomic symbol (or equally the atomic number), the number of explicit hydrogens attached, and the number of formal charges (see **c**). For bond types, only single, double, triple and aromatic bonds are considered in this work (see **d**)

Enumeration of a chemical structure into data to pass into the neural network.

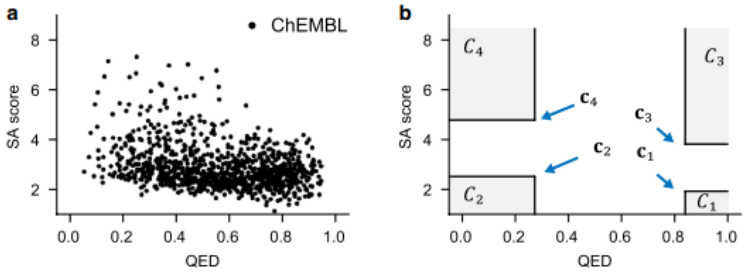


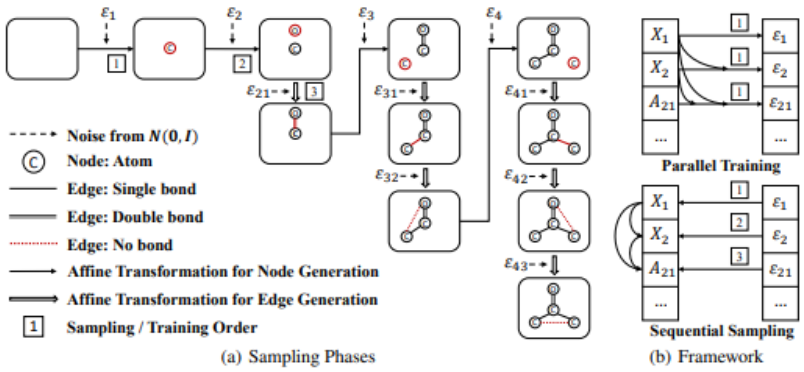
Fig. 10 Location of $C_1 \sim C_4$ and $c_1 \sim c_4$. **a** Distribution of QED and SAscore in the ChEMBL dataset; **b** Location of input conditions ($C_1 \sim C_4$ and $c_1 \sim c_4$)

Distribution of QED and SA metrics in the ChEMBL dataset.

Notes	Background
	- Estimated 10^60 to 10^100 synthetically available molecules
	- Discontinuous space = difficult to search
	
	Methods
	- Map to latent space with graph convolutional model
	- Autoregressive molecule generation (step by step)
	- Use of decoding scheme w/ 4 choices
	  - Initialize
	  - Append
	  - Connect
	  - Terminate
	- Decoding policy needs probability value of each transition

Cited references to follow up on	None
Follow up Questions	- If the model can account for chemical validity rules, why doesn't it create 100% chemically valid molecules? - How are constraints coded into the probability model? - Can constraints be discrete, or are they required to yield a continuous range of values/probability distribution? - Can the graph convolutions in the model be replaced with TrimNet in order to improve performance?

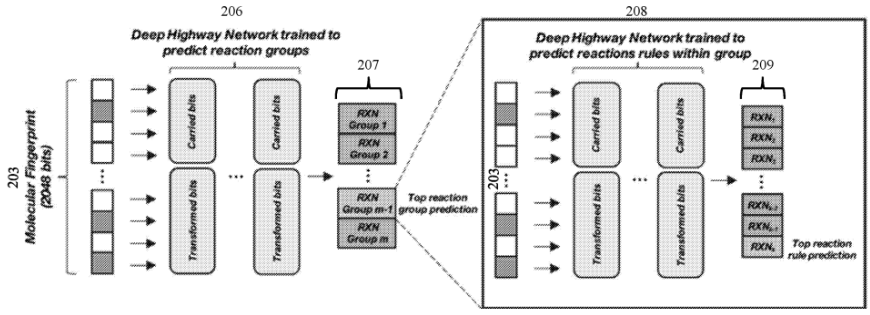
Article #10 Notes: Autoregressive Flow with GraphAF

Source Title	GraphAF: A Flow-Based Autoregressive Model for Molecular Graph Generation
Source citation (APA Format)	Shi, C., Xu, M., Zhu, Z., Zhang, W., Zhang, M., & Tang, J. (2020). GraphAF: A Flow-based Autoregressive Model for Molecular Graph Generation. *ArXiv:2001.09382 [Cs, Stat]*. http://arxiv.org/abs/2001.09382
Original URL	https://arxiv.org/pdf/2001.09382.pdf
Source type	Journal Article
Keywords	From [5]
Summary of key points (include methodology)	In the field of deep learning de novo drug design, generating novel molecules from scratch is difficult because the generated molecules must conform to chemical rules and be optimized for certain properties. To remedy this issue, the authors of this paper propose a framework called GraphAF, which is based on a normalizing flow model. In a flow model, the objective is to define a function (deep learning network) that maps from a graph input to latent space. Using the change of variables formula, the function can be reversed. The framework generates molecules through an iterative process via reinforcement learning, which enables it to check for chemical validity at each step. When testing the model, the researchers found that it was capable of outperforming previous state of the art models. It was also 2x faster than the state of the art, demonstrating that normalizing flow is a faster alternative to previous generation methods. As a future direction, the authors indicate that they will test the scalability of the model to larger datasets.
Research Question/Problem/ Need	It is difficult to generate molecular graphs with deep learning because of the requirement to create chemically valid molecules and optimize for certain properties.
Important Figures	 (a) Sampling Phases (b) Framework A diagram of the GraphAF framework.
Notes	Abstract - Chemical validity and property optimization issue - Flow-based autoregressive model proposed

	- Model benefits:
	  - High model flexibility for data density estimation
	  - Efficient parallel computation for training
	  - Iterative sampling = ability to incorporate chemistry rules
	  - 100% validity w/ chemical rules
	  - 2x faster than the existing state of the art. GCPN
	Introduction
	- Significant progress recently in graph generation and normalizing flow models
	- Existing autoregressive techniques too slow
	Methodology
	- Defines an invertible transformation from base distribution (multivariate Gaussian) to molecular graph structure, G=(A, X)
	- Start with an empty graph and iteratively make modifications based on previous sub-graph
	- Node type and edge type are discrete, and need a dequantization technique to convert into continuous distribution. (add noise)
	- Parallel training & masking
	- Use conditional logic to check for valency
	- Reinforcement learning to do molecular property optimization
	  - Penalized LogP and QED used as metrics for evaluation

Cited references to follow up on	N/A
Follow up Questions	- Can a flow-based model be applied to retrosynthesis prediction? - Has the model been tested with metrics other than logP and QED? - Are there other chemical rules that can be added to the model other than valency in order to improve performance? - Why are the properties of a molecular structure preserved when its graph is converted to BFS form during parallel training?

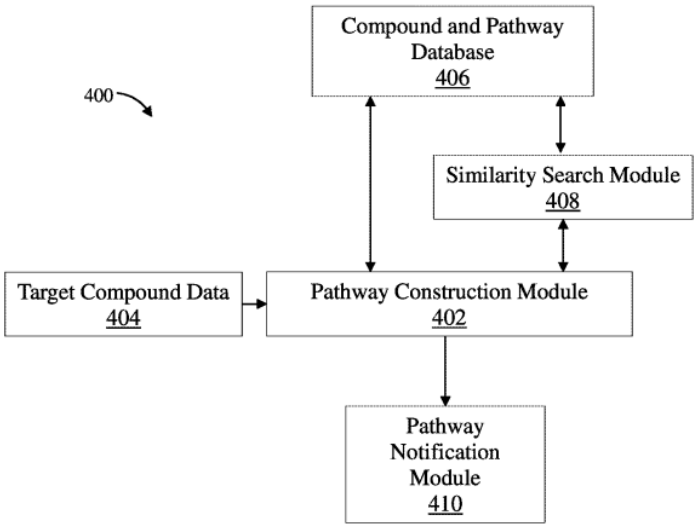
Article #11 Notes: [PATENT] Deep Highway Networks for Retrosynthesis Prediction

Source Title	Retrosynthesis Prediction Using Deep Highway Networks and Multiscale Reaction Classification
Source citation (APA Format)	Baylon, J., Cilfone, N., Gulcher, J., & Chittenden, T. (2020). *Retrosynthesis Prediction Using Deep Highway Networks and Multiscale Reaction Classification.*
Original URL	https://patentimages.storage.googleapis.com/fb/f2/35/242fcd21ebb48c/WO2020023650A1.pdf
Source type	Patent
Keywords	Deep Learning, Retrosynthesis
Summary of key points (include methodology)	Synthesis planning is highly important in the field of drug discovery, where it is necessary to know the optimal method of synthesizing a molecule. However, current template-based models are unable to generalize to new cases because they rely solely on reaction rules given to them manually or by a dataset. This patent proposes a deep-learning driven approach to computational retrosynthesis prediction which is able to generalize better to novel reactions. The framework achieves this by integrating deep highway networks in a multi-step process to predict the starting reactants of a given molecule., which includes the use of molecular fingerprints to determine a reaction rule, then reversing the reaction to obtain the products. The use of deep learning allows it to better learn chemical rules without it being taught explicitly, thus allowing for better generalization.
Research Question/Problem/ Need	Template-based models for retrosynthesis prediction are unable to account for novel reactions, and are therefore ineffective at generalizing to new inputs.
Important Figures	 Diagram of proposed framework
Notes	- Written in Tensorflow - Deep Highway Network - Use of multiscale templates extracted from smaller datasets instead of alternative

	template method - Learns patterns over molecules that were obtained via similar reactions. - Multi-step framework: - Determine molecular fingerprint - Pass fingerprint to 1st classifier - Determine candidate group - Select a second classifier based on candidate group - Pass fingerprint to 2nd classifier - Determine candidate reaction rule - Determine reactants based on candidate rule Majority of the 2nd half of the document details possible implementations of the framework, including a web application.
Cited references to follow up on	
Follow up Questions	- Can the process be repeated iteratively for multi-step retrosynthesis? - How would such a model respond to a case where the predicted reactants are chemically invalid/unobtainable? - Can the model be modified to output a probability that the given molecule is synthesizable? - How does the use of reaction templates in this framework differ from the use of templates in previous computational methods?

Article #12 Notes: [PATENT] Efficient Retrosynthesis Analysis

Source Title	Efficient Retrosynthesis Analysis
Source citation (APA Format)	Botea, A., Buesser, B., Chen, B., Kishimoto, A., & Savage, J. (2020). *Efficient Retrosynthesis Analysis.*
Original URL	https://patentimages.storage.googleapis.com/90/1b/a8/4d60e016e332d2/US10679733.pdf
Source type	Patent
Keywords	Retrosynthesis
Summary of key points (include methodology)	An efficient method of identifying synthesis routes for target molecules is needed in the pharmaceutical, chemical, material, and food industries. Traditional methods of identifying synthesis pathways (namely, manual retrosynthetic analysis) are prone to error and often result in suboptimal routes. The patent proposes a workflow that is capable of predicting more efficient synthetic routes computationally. It involves an iterative process, which begins with identifying similar molecules with known synthetic routes in a database to a given target compound via Jaccard similarity. Using these pathways, the model predicts whether the target molecule can be synthesized in a similar way using known reaction rules. If so, the model uses these reaction rules to generate precursors which are reacted to form the target. This process is repeated, with the precursors becoming the new targets, until a set of precursors which are purchasable or have known synthesis routes is generated. The patent concludes by proposing a computing environment in which this method can be executed.
Research Question/Problem/ Need	The patent was proposed in response to a need for an efficient retrosynthetic system to determine whether a molecule can be synthesized.

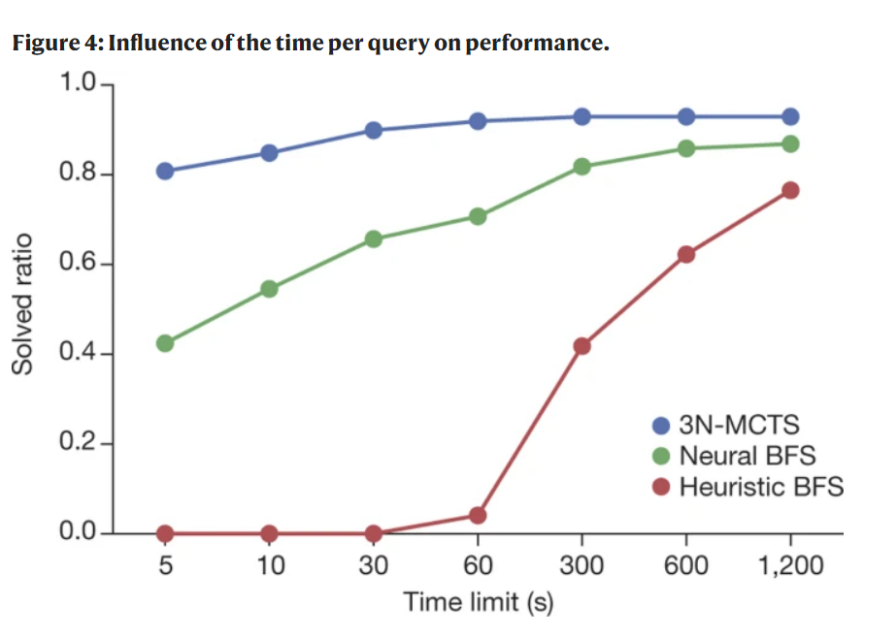
Important Figures	 A component diagram for the framework proposed by the patent.

Notes	- Manual retrosynthetic analysis prone to error
	  - Results in suboptimal pathways
	- Smaller compounds with simpler structures are desired
	- NCBI PubChem database has 30 million molecules
	- Increasing quantity of known compounds = slower computation speed for brute-force searching.
	- NEW METHOD IS STILL PURELY COMPUTATIONAL
	- Proposes a system that delivers a product to multiple users simultaneously.
	- Use of Jaccard coefficient to determine compound similarity.
	- Iterative process:
	  - Identify similar compounds to target with known synthesis pathway
	  - Check if the target can be synthesized using similar reaction rules.
	  - Use reaction rules to break targets down into precursors.
	  - Repeat the process until a set of precursors with known synthesis routes is located.

Cited references to follow up on	N/A
Follow up Questions	- Can this framework be integrated into a molecule optimization model? - How can the framework be modified to account for purchasability of starting reactants? - Can the purchasability of starting reactants be determined without first performing a full retrosynthetic analysis of the molecule?

	- How effective is Jaccard similarity at determining similar pathways, given that similar molecules may have drastically different synthetic routes?

Article #13 Notes: Chemical synthesis planning with deep learning and symbolic AI

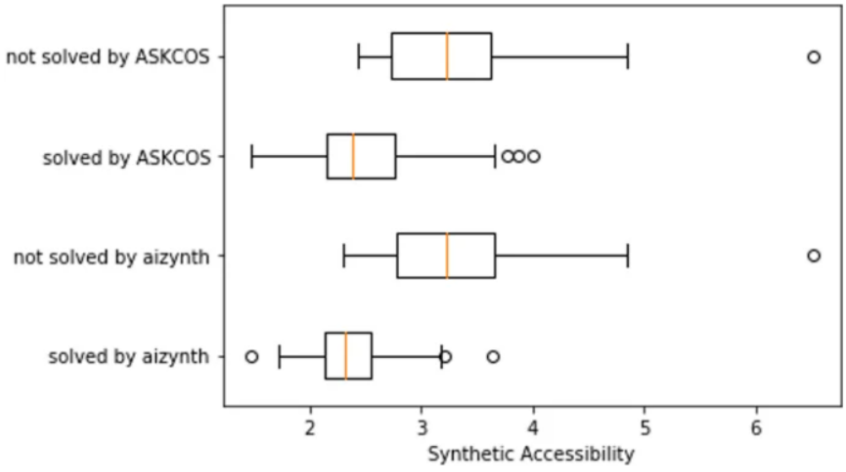
Source Title	Planning chemical syntheses with deep neural networks and symbolic AI
Source citation (APA Format)	Segler, M. H. S., Preuss, M., & Waller, M. P. (2018). Planning chemical syntheses with deep neural networks and symbolic AI. *Nature, 555*(7698), 604–610. https://doi.org/10.1038/nature25978
Original URL	https://www.nature.com/articles/nature25978
Source type	Journal Article
Keywords	Deep Learning, Chemical Synthesis
Summary of key points (include methodology)	In response to the need for more accurate methods of computational synthesis route prediction, the authors of this paper propose a new method of computationally deriving a molecule's synthesis pathway with deep learning. The framework is built on Monte Carlo tree search, and uses 3 neural networks to aid in the prediction of synthesis routes. These 3 neural networks are responsible for identifying plausible chemical reactions from a dataset, assessing the feasibility of said reactions, and predicting the outcome of the reactions. Compared to previous computational methods, the proposed model generates more feasible synthesis pathways, scoring as well as in-literature pathways when evaluated by professionals in a double-blind test.
Research Question/Problem/ Need	Traditional methods of computational retrosynthetic analysis have a lack of chemical intelligence and have high noise.
Important Figures	 Comparison of different methods' accuracy over time curves. As can be seen, 3N-MCTS is

	accurate even under a short time constraint and maintains its top position even as time limit increases.

Notes	Introduction
	- Out of scope reaction = does not proceed as predicted due to incomplete understanding, steric/electronic effects, or conflicting reactivity.
	- Introduction has good references for why computational retrosynthesis methods are ineffective
	  - High noise and lack of chemical intelligence
	- Deep learning methods can extract symbolic transformations, mimicking experts' intuitive decision making.
	- Heuristic best first search vs Monte Carlo tree search
	- 3 neural networks used alongside MCTS
	  - Expansion policy = finds most promising transformations from dataset
	  - 2nd network predicts if transformation is feasible
	  - Rollout policy = predict position value.
	- Trained on virtually all published reactions in organic chemistry.
	Expansion and Rollout Policies
	- 12.4 million transformation rules extracted from Reaxys chemistry database.
	- Rollout design:
	  - rules that contain atoms/bonds changed during reaction and first degree neighbors.
	- Expansion design:
	  - General rule definition, only reaction center extracted.
	  - Deep highway network with exponential linear unit nonlinearities.
	- Expansion policy 31% accuracy (decent performance, since multiple synthetic routes available. Network evaluates as correct when most optimal route is predicted)

Cited references to follow up on	N/A
Follow up Questions	- Can heuristic approaches be improved by using neural models instead of hard-coded rules? - What is the purpose of the intermediate neural network classifier? - How exactly does Monte Carlo tree search play a role in improving the model? - How were hyperparameters tuned for each model?

Article #14 Notes: AiZynthFinder

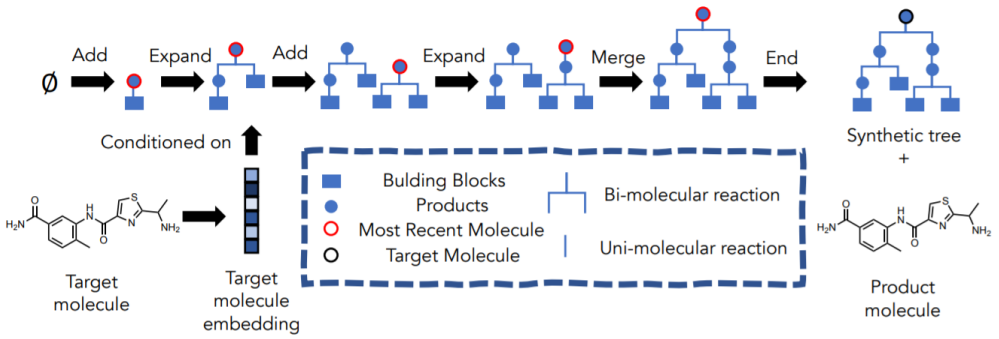
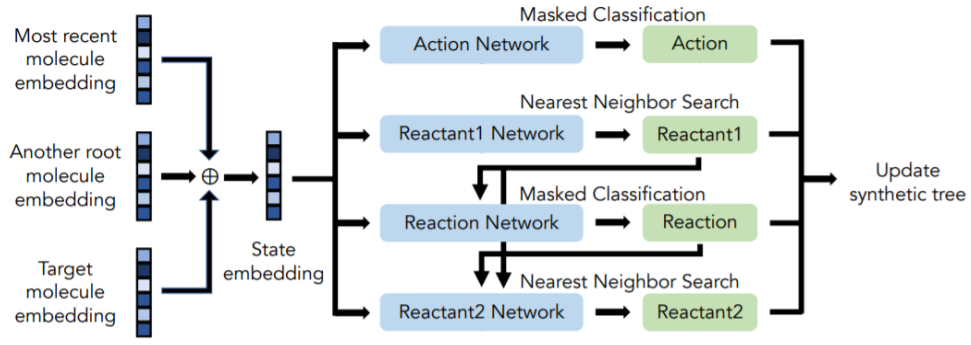
Source Title	AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning
Source citation (APA Format)	Genheden, S., Thakkar, A., Chadimová, V., Reymond, J.-L., Engkvist, O., & Bjerrum, E. (2020). AiZynthFinder: A fast, robust and flexible open-source software for retrosynthetic planning. *Journal of Cheminformatics, 12*(1), 70. https://doi.org/10.1186/s13321-020-00472-1
Original URL	https://jcheminf.biomedcentral.com/track/pdf/10.1186/s13321-020-00472-1.pdf
Source type	Journal Article
Keywords	Machine Learning, Chemical Synthesis
Summary of key points (include methodology)	The authors of this paper identify the need for more transparent, user-friendly interfaces for synthetic route prediction. The system described in the article provides the implementation details for an open source software developed for this purpose. Using Monte Carlo tree search, the algorithm performs retrosynthetic analysis on a target molecule, typically achieving results in under 10 seconds. The algorithm is implemented as both a graphical user interface (GUI) and a command-line interface (CLI).
Research Question/Problem/ Need	The application described in the article was developed in response to a need for algorithmic transparency and user-friendliness in open source retrosynthesis software.
Important Figures	 AiZynthFinder vs ASKCOS, in terms of whether the application was able to find a synthesis route for a molecule of the given SA score.

Notes	Introduction
	- Retrosynthesis planning: template-based and template-free
	  - Template-based: reaction templates from a database of known reactions are used.
	  - Template-free: no use of reaction templates - treat as an NLP problem,

	where reactants -> products.
	- Some open source software for retrosynthesis prediction exist
	  - Need for more transparent code (published in GitHub)
	  - Need to be more maintainable - written with better software engineering techniques
	- Include extensive documentation to be user-friendly.
	Implementation
	- Central algorithm not described in this literature
	- Input: target compound to be broken into precursors
	- Outcome: list of purchasable compounds / molecules that can't be solved by the algorithm
	- Based on monte carlo tree search
	  - Each leaf node: set of molecules that can/cannot be broken down further.
	  - Use upper confidence bound statistics to determine the most promising node.
	  - Neural Network Policy used to shortlist reaction templates to prioritize which child to create.
	  - Create precursor with chosen reaction template, then repeat until a purchasable precursor is found (or max depth is reached)
	  - Backpropagate to tree root and repeat.
	- Does not have a filter to quickly remove unfeasible reactions
	- No use of different policies for expansion and rollout phases.

Cited references to follow up on	N/A
Follow up Questions	- Can the synthetic accessibility score be used as a heuristic parameter to optimize in deep learning models? - How does the network policy prioritize a certain reaction template? - Why can't the additional features of the CLI be added to the Graphical Interface? - How are molecules represented numerically in this system?

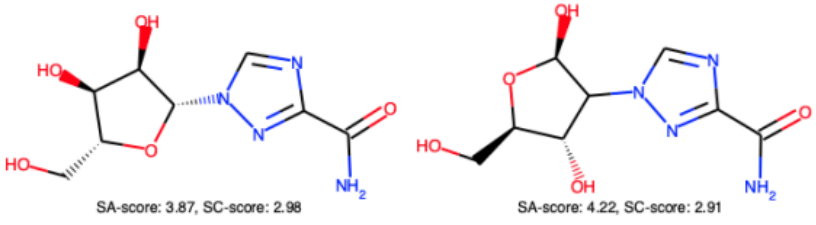
Article #15 Notes: Amortized tree generation

Source Title	Amortized Tree Generation for Bottom-Up Synthesis Planning and Synthesizable Molecular Design
Source citation (APA Format)	Gao, W., Mercado, R., & Coley, C. W. (2021). Amortized Tree Generation for Bottom-up Synthesis Planning and Synthesizable Molecular Design. *ArXiv:2110.06389 [Cs, q-Bio]*. http://arxiv.org/abs/2110.06389
Original URL	http://arxiv.org/abs/2110.06389
Source type	Journal Article (preprint)
Keywords	Deep Learning, Chemical Synthesis (found via. NYAGIM talk)
Summary of key points (include methodology)	In response to the need for generative models with improved rates of synthesizability, Gao et al. propose a deep learning model which
Research Question/Problem/ Need	Deep learning models for molecule generation are incapable of ensuring synthesizability.
Important Figures	 Framework for synthetic route generation proposed by Gao et al.  Generation model proposed by Gao et al.

Notes	Methods
	- Problem Definition
	  - Model synthetic pathways as tree structures “synthetic trees”
	  - Valid synthetic tree has 1 root node (final product)
	  - Leaf nodes must be purchasable from a catalog.
	  - Links represent reaction templates chosen from a discrete set.
	  - Reaction templates pattern defining a structural transformation between molecules to represent a chemical reaction, typically encoded as a SMARTS string.
	- Synthesis Planning
	  - Goal: infer the synthesis route of a given molecular structure.
	  - Generate a synth tree T in which the root matches the target.
	- Synthesizable molecular design
	  - Optimize a molecular structure with respect to an oracle function while ensuring synthesizability.
	- Markov property: $p(S^{(t+1)}|S^{(t)},\ldots,S^{(0)}) = p(S^{(t+1)}|S^{(t)})$
	
	Synthetic Tree Generation as a Markov Decision Process
	- Framework only allows uni- and bi-molecular reactions.
	- State space = root of synthetic tree at t.
	- Action space = {action type, reactant 1, template, reactant 2}
	
	Conditional Generation for Synthesis Planning
	- 4 models used to for molecule generation
	  - Action Type: select type of action to perform (add, expand, merge, end)
	  - Predict first reactant
	  - Predict reaction template
	  - Predict second reactant
	$a_{\text{act}}^{(t)} \sim f_{\text{act}}(S^{(t)}, M_{\text{target}}) = \sigma(\text{MLP}_{\text{act}}(z_{\text{state}}^{(t)} \oplus z_{\text{target}}))$
	$a_{\text{rt1}}^{(t)} \sim f_{\text{rt1}}(S^{(t)}, M_{\text{target}}) = \text{k-NN}_{\mathcal{C}}(\text{MLP}_{\text{rt1}}(z_{\text{state}}^{(t)} \oplus z_{\text{target}}))$
	$a_{\text{rxn}}^{(t)} \sim f_{\text{rxn}}(S^{(t)}, a_{\text{rt1}}^{(t)}, M_{\text{target}}) = \sigma(\text{MLP}_{\text{rxn}}(z_{\text{state}}^{(t)} \oplus z_{\text{target}} \oplus z_{\text{rt1}}^{(t)}))$
	$a_{\text{rt2}}^{(t)} \sim f_{\text{rt2}}(S^{(t)}, a_{\text{rt1}}^{(t)}, a_{\text{rxn}}^{(t)}, M_{\text{target}}) = \text{k-NN}_{\mathcal{C}'}(\text{MLP}_{\text{rt2}}(z_{\text{state}}^{(t)} \oplus z_{\text{target}} \oplus z_{\text{rt1}}^{(t)} \oplus z_{\text{rxn}}^{(t)}))$

Cited references to follow up on	REACTOR: Reinforcement Learning for Synthetic Pathway Generation Horwood, J., & Noutahi, E. (2020). Molecular Design in Synthetically Accessible Chemical Space via Deep Reinforcement Learning. *ACS omega, 5(51)*, 32984–32994. https://doi.org/10.1021/acsomega.0c04153
Follow up Questions	- How well does the model perform on optimizing multiple targets at the same time? - Why did the neural graph representation perform poorly compared to morgan fingerprints? - Why is it necessary to select the two reactants independently? - How can this model be scaled to interact with other deep learning predictors?

Article #16 Notes: Synthesizability of molecular generative models

Source Title	The Synthesizability of Molecules Proposed by Generative Models
Source citation (APA Format)	Gao, W., & Coley, C. (2020). The Synthesizability of Molecules Proposed by Generative Models. *Journal Of Chemical Information And Modeling, 60*(12), 5714-5723. doi: 10.1021/acs.jcim.0c00174
Original URL	https://arxiv.org/pdf/2002.07007.pdf
Source type	Journal Article
Keywords	Deep Learning, Chemical Synthesis
Summary of key points (include methodology)	This paper was written to present the works of a study which analyzed various methods of increasing synthesizability rates of molecular generative models. The paper presents two types of generative frameworks, distribution learning and goal-directed benchmarks. Distribution learning takes a starting database and generates large quantities of molecules with similar properties, while goal-directed learning attempts to create molecules from scratch with defined properties. To improve synthesizability of these frameworks, the researchers tested three methods: a priori biasing, heuristic biasing, and post hoc filtering. A priori biasing aims to limit the training database to synthesizable molecules only in the hopes that this will bias the model to create synthesizable compounds. Heuristic biasing uses a heuristic scoring method to estimate the synthetic accessibility of the given compound, and post hoc filtering simply checks the set of output molecules and removes ones that aren't synthesizable. The study found that, although all approaches improved synthesizability to some extent, each one
Research Question/Problem/ Need	How well do various proposed techniques perform at ensuring synthesizability of molecular generative models?
Important Figures	 Figure S1: Illustration of the difficulty of applying heuristics to estimate synthesizability. Ribavirin (left) and its analogue (right) are structurally very similar, but their syntheses would be substantially different due to the inherent reactivity of ribose to favor substitution at the position leading to ribavirin. The SA Score and SCScore do not reflect that the right hand compound is much harder to access.
Notes	- MOSES, ChEMBL, ZINC, GDB17 - A priori: synthesizability of generative models trained on each dataset is very

	similar (but never better) than native synthesizability. - No relationship explicitly tested. Goal Directed Benchmarks - Heuristic biasing: SAScore > SMILES > SC_Score - A priori biasing not effective for goal-directed; 30.2% synth for ChEMBL and 32.7% for MOSES. - Heuristic biasing decreases the value of objective function significantly. - Proposed workflow: - Optimize and do a post-hoc filter IF only a few synthesizable candidates are desired. - If top synthesizable candidates are worse than top unsynthesizable candidates, repeat optimization with SA_Score bias. - Forward-direction molecule generation from a bag of starting reactants is a promising new approach and should match the performance of virtual screening libraries.
Cited references to follow up on	N/A
Follow up Questions	- How was the SCScore heuristic approach designed, and why did it underperform SMILES string length for heuristic biasing? - Why is optimization of properties sacrificed when synthesizability is ensured? - Is it possible to do CASP oracle biasing using only a portion of the full analysis? - Why is a priori biasing ineffective in goal-directed benchmarks?

Article #17 Notes: MPNN

Source Title	Neural Message Passing for Quantum Chemistry
Source citation (APA Format)	Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., & Dahl, G. E. (2017). Neural Message Passing for Quantum Chemistry. *ArXiv:1704.01212 [Cs]*. http://arxiv.org/abs/1704.01212
Original URL	https://arxiv.org/pdf/1704.01212.pdf
Source type	Journal Article
Keywords	Message Passing Neural Network
Summary of key points (include methodology)	The authors of this paper create a framework, called MPNN, to generalize a class of models published in literature. These models operate on discrete graph representations of molecules, which bypasses the need for computing expensive features in machine-learning based chemistry tasks. In essence, MPNNs comprise of 2 phases: message passing and readout. In the message passing phase, the nodes of the graph pass "messages" to their neighbors, and the graph is updated with the summation of these messages. In the readout phase, the graph representation is converted by a neural network model into a latent representation. To apply this concept to chemistry, the authors design a MPNN to predict the quantum molecular properties of molecules in the QM9 dataset, and they found that the model outperforms previous state of the art approaches. As future steps, the model should generalize to larger graph structures; currently, the performance of the model decreases as the input graphs get larger in size.
Research Question/Problem/ Need	How can graph neural networks be applied to chemistry-oriented problems?
Important Figures	*Table 2.* Comparison of Previous Approaches (left) with MPNN baselines (middle) and our methods (right)

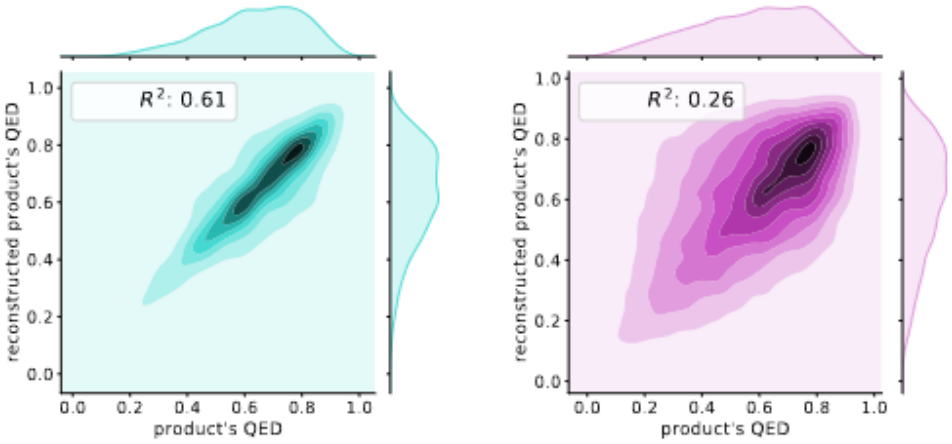
Target	BAML	BOB	CM	ECFP4	HDAD	GC	GG-NN	DTNN	enn-s2s	enn-s2s-ens5
mu	4.34	4.23	4.49	4.82	3.34	0.70	1.22	-	**0.30**	0.20
alpha	3.01	2.98	4.33	34.54	1.75	2.27	1.55	-	**0.92**	0.68
HOMO	2.20	2.20	3.09	2.89	1.54	1.18	1.17	-	**0.99**	0.74
LUMO	2.76	2.74	4.26	3.10	1.96	1.10	1.08	-	**0.87**	0.65
gap	3.28	3.41	5.32	3.86	2.49	1.78	1.70	-	**1.60**	1.23
R2	3.25	0.80	2.83	90.68	1.35	4.73	3.99	-	**0.15**	0.14
ZPVE	3.31	3.40	4.80	241.58	1.91	9.75	2.52	-	**1.27**	1.10
U0	1.21	1.43	2.98	85.01	0.58	3.02	0.83	-	**0.45**	0.33
U	1.22	1.44	2.99	85.59	0.59	3.16	0.86	-	**0.45**	0.34
H	1.22	1.44	2.99	86.21	0.59	3.19	0.81	-	**0.39**	0.30
G	1.20	1.42	2.97	78.36	0.59	2.95	0.78	.84[2]	**0.44**	0.34
Cv	1.64	1.83	2.36	30.29	0.88	1.45	1.19	-	**0.80**	0.62
Omega	0.27	0.35	1.32	1.47	0.34	0.32	0.53	-	**0.19**	0.15
Average	2.17	2.08	3.37	53.97	1.35	2.59	1.36	-	**0.68**	0.52

Notes	$$m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw}) \quad (1)$$ $$h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) \quad (2)$$ where in the sum, $N(v)$ denotes the neighbors of v in graph G. The readout phase computes a feature vector for the whole graph using some readout function R according to $$\hat{y} = R(\{h_v^T \mid v \in G\}). \quad (3)$$ Equations for message passing and readout.
Cited references to follow up on	N/A
Follow up Questions	- Can the MPNN framework be further generalized to molecule generation problems? - How significant is the error between DFT calculations and empirical values (nature)? -

Article #18 Notes: MoleculeChef

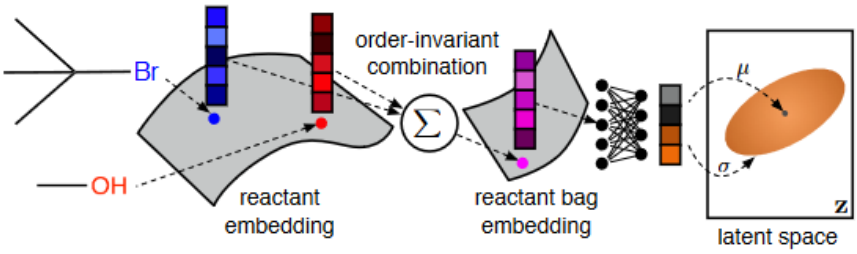
Source Title	A Model to Search for Synthesizable Molecules
Source citation (APA Format)	Bradshaw, J., Paige, B., Kusner, M. J., Segler, M. H. S., & Hernández-Lobato, J. M. (2019). A Model to Search for Synthesizable Molecules. *ArXiv:1906.05221 [Physics, Stat]*. http://arxiv.org/abs/1906.05221
Original URL	http://arxiv.org/abs/1906.05221
Source type	Journal Article
Keywords	Synthesizable Molecular Design
Summary of key points (include methodology)	Traditional models for molecule generation fail to account for the synthesizability of the generated molecules. The authors of this paper introduce the framework for a new approach, MoleculeChef, which incorporates synthesizability as an implicit bias to the model. This is done by using two neural networks: a model to predict an initial bag of starting reactants, and a model to predict the chemical reactions that occur between them that lead to an output compound. If the bag of reactants contain molecules that can all be purchased from a vendor, then the output molecule is synthesizable.
Research Question/Problem/ Need	Current models for molecule generation do not consistently generate synthesizable molecules.

Important Figures

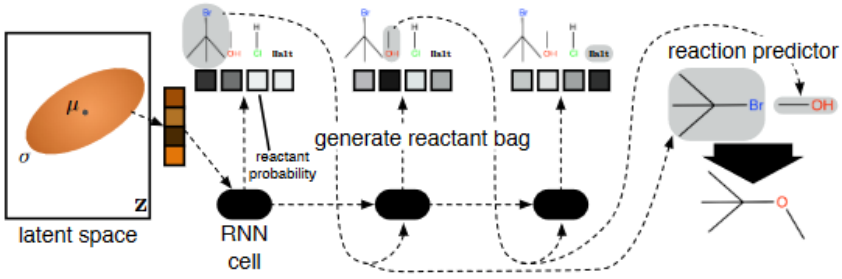


(a) Reachable Products (b) Unreachable Products

Figure 8: Assessing the correlation between the QED scores for the original product and its reconstruction (see text for details). We assess on two portions of the test set, products that are made up of only reactants in MOLECULE CHEF's vocabulary are called 'Reachable Products', those that have at least one reactant that is absent are called 'Unreachable Products'.



Embedding of reactants



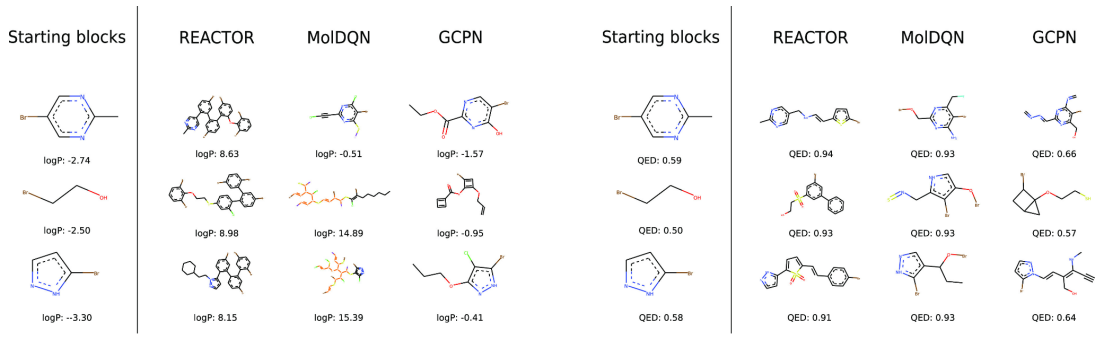
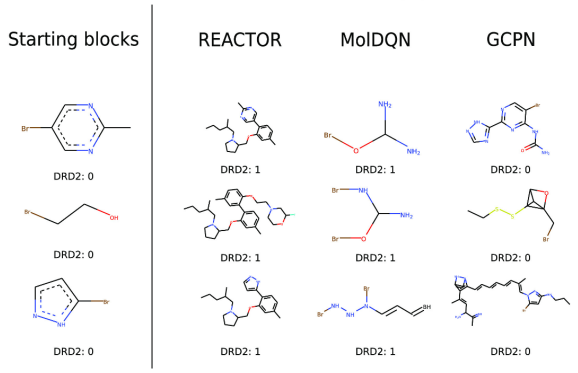
	Decoding mechanism

Notes	Background
	- Virtual screening
	  - Enumerate all possible combinations of precursors
	  - Calculate properties and filter interesting ones
	- The molecular search problem
	  - Replace enumeration with search
	  - Early methods = molecular space to latent space and back
	  - Segler et al. and Gomez-Bombarelli et al. use NLP methods
	
	Model
	- https://github.com/john-bradshaw/molecule-chef
	- 2 Model components:
	  - Encoder/Decoder (latent space bag of reactants)
	  - Reaction predictor to generate a multiset of molecules
	- Encoder and Decoder
	  - Options for representation: ignore molecular structure and learn a distinct representation for each molecule, or compute fingerprints.
	  - $q(z \mid x)$ maps from multiset of reactants to a latent embedding
	  - Summation over individual reactant embeddings (GGNN)?
	  - Predictive Penalty Loss
	  - Optimization of QED

Cited references to follow up on	N/A
Follow up Questions	- How is MoleculeChef's latent space z enumerated? - Can MoleculeChef be expanded to do searches for structurally similar molecules? - How frequent is the rate of false positives generated by the reaction predictor? - Was there a specific reason for choosing GGNN as the embedding generator?

Article #19 Notes: REACTOR - Reinforcement Learning of Synthetic Pathways

Source Title	Molecular Design in Synthetically Accessible Chemical Space via Deep Reinforcement Learning
Source citation (APA Format)	Horwood, J., & Noutahi, E. (2020). Molecular Design in Synthetically Accessible Chemical Space via Deep Reinforcement Learning. *ACS omega, 5(51)*, 32984–32994. https://doi.org/10.1021/acsomega.0c04153
Original URL	https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7774092/
Source type	Journal Article
Keywords	Reinforcement Learning, Synthetic Pathway Generation
Summary of key points (include methodology)	In order to address synthesizability in generative models, the authors of this paper propose a framework for molecule generation involving the generation of synthetic pathways with reinforcement learning. In order to do so, they formulate the problem as a markov decision process. In contrast to previous approaches, their framework, called REACTOR, generates molecules by iteratively applying reaction templates to a set of initial starting reactants. Such an approach ensures synthesizability implicitly, while being able to optimize for molecular properties via the reward function. However, it seems that the generated results are inherently un-druglike.
Research Question/Problem/ Need	Deep learning models for molecule generation ignore synthesizability of results.

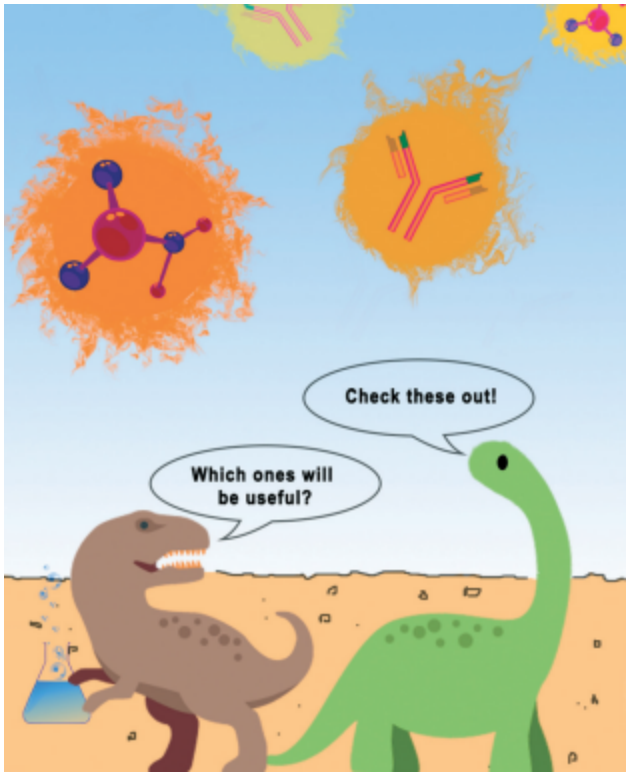
Important Figures	 (a) Samples for cLogP objective (b) Samples for QED objective  (c) Samples for DRD2 objective Generated samples by goal-driven generation of synthesis routes for property optimization.

Notes	Computational drug design methods have tradeoffs
	- Molecular generation
	  - Novelty and validity
	- Molecular optimization
	  - Properties
	- Atom-based RL = combinatorial enumeration of molecular states
	  - Slow convergence
	
	Methodology
	- Use of markov decision process
	- Molecular Design via Synthesis Trajectories
	  - Embed synthesizability directly into framework
	- Formulation of MDP
	  - State space = any valid molecule
	  - Action space is defined hierarchically
	$$R = r_1 + r_2 + \ldots + r_k \rightarrow (p_1, \ldots, p_m)$$
	  -
	  - Where r represents a reactant and p_{i-m} are the products of the reaction.
	  - Starting space initialized with a randomly sampled building block that matches at least one reaction template.

	- Experiments
	  - Used 2 reactant templates
	  - Select missing reactants based on what will most improve next state's reward
	- Current reinforcement learning techniques are poorly suited for large discrete action spaces.
	- BRICS retrosynthesis rules are used to limit original reactant set
	  - Set further reduced as a hyperparameter to the model
	Results
	- Evaluated on QED and clogP
	- Reaction templates and dataset seem to be rather small for a reinforcement learning framework
	- "We prioritize the evaluation of each method based on the total number of active molecules identified, as determined by the environment reward model, given that this corresponds most to the underlying objective of *de novo* design. Indeed, in a hit discovery scenario, a user may be most interested in identifying the maximal number of unique potential hits, leaving potency optimization to later stages in the lead optimization process."

Cited references to follow up on	N/A
Follow up Questions	- Why did the authors choose to pursue synthesizability of lead molecules if further optimization of these molecules may break synthesizability? - Are the generated molecules good enough to pass initial clinical trials? - What caused the indirect optimization of DRD2 as detailed in the results section? - How can this model be improved to optimize already good leads?

Article #20 Notes: What are the drugs of the future?

Source Title	What are the drugs of the future?
Source citation (APA Format)	Ngo, H. X., & Garneau-Tsodikova, S. (2018). What are the drugs of the future? *MedChemComm*, *9*(5), 757–758. https://doi.org/10.1039/C8MD90019A
Original URL	https://pubs.rsc.org/en/content/articlelanding/2018/md/c8md90019a
Source type	News/Review Article
Keywords	Small Molecule vs Biologics
Summary of key points (include methodology)	This article attempts to highlight the pros and cons of small molecule therapeutics and biologics. Recent advancements in biologics, alongside a lag in the production of small molecule drugs, led many to believe that the small molecule drug industry will eventually be replaced by biologic treatment entirely.
Research Question/Problem/ Need	Will small molecule drugs or biologics be more prominent in the future?
Important Figures	 Supposed to represent the continuous advancement of new technologies to replace existing ones.
Notes	Small-molecule drugs

	- Blockbuster model generates a lot of money
	- Pros:
	  - More predictable pharmacokinetic properties due to low molecular weight
	  - Simpler manufacturing
	  - Simpler dosing protocols
	  - Stable and oral bioavailability
	  - Affordable
	- Cons:
	  - Simple design = high competition
	  - Some targets may not be druggable; small molecules work by mimicking biological substrates or allosterically targeting hydrophobic protein pockets.
	Biologic Drugs
	- Pros
	  - Rising usage
	- Cons
	  - High cost of manufacturing
	  - Fragile and sensitive to degradation
	  - Only available via injection.
	  - Patients can develop immune responses to biologics over time = loss of effectiveness.
	Future
	- Small molecule treatments still necessary for affordability and treatment of chronic diseases

Cited references to follow up on	N/A
Follow up Questions	- How will the article's future outlook change if biologics become cheaper to produce in the future? - How important is oral bioavailability to a consumer? - Do small molecule drugs or biologic drugs cover a wider range of targets? - Is drug immunity an issue for small molecule drugs as well?