Generation of novel and diverse molecules using Self-attention Generative Adversarial Networks

Abstract—**In discrete sequence based Generative Adversarial Networks (GANs), it is important to both land the samples in the initial distribution and drive the generation towards desirable properties. However, in the case of longer molecules, the existing models seem to under-perform in producing new molecules. In this work, we propose the use of Self Attention mechanism for Generative Adversarial Networks to allow long range dependencies. Self-Attention mechanism has produced improved rewards in novelty and promising results in generating molecules.**

1. Introduction

De novo molecule generation for drug synthesis is an important problem in the field of cheminformatics. Recent advances in computer hardware and software have enabled application of unsupervised generation of data to the field of Drug discovery.

In the field of unsupervised molecule generation, it is often preferred to drive the generation process towards some desirable properties, while ensuring that the produced output falls in the boundary of initial distribution. In the field of natural language generation for example, a selected sentiment might be enhanced, maybe for producing movie reviews [20]. Similarly, in materials and drug discovery, the aim is often to optimize some heuristics pertaining to the properties of the materials, for example in organic solar cells [12], OLEDs [7] or new drugs. The generation of discrete data using Recurrent Neural Networks (RNNs), in particular, Long Short-Term Memory cells (LSTMs) [14] and maximum likelihood estimation has been shown to work well in practice. However, this often suffers from the so-called exposure bias, and might lack some of the multi-scale structures or salient features of the data.

Meanwhile Generative Adversarial Networks (GANs) [9], an approach where a generator competes with a discriminative model, one trying to generate likely data while the other trying to differentiate false from the real data. GANs have shown promising results at generation of data that imitates a data distribution. Although GANs were not initially applicable to discrete data due to non-differentiability, approaches such as SeqGAN [27], MaliGAN [3] and BGAN [13] have arisen to deal with this issue. Moreover, Reinforcement Learning (RL) have shown huge success at solving issues where continuous feedback from an environment is necessary [13].

In this paper, we introduce a de novo approach to optimize the properties of a distribution of sequences, increase the diversity of the samples while maintaining the likeliness of the data distribution. In our approach, the generator is trained to maximize a weighted average of two types of rewards: novelty, validity, which are the domain-specific metrics, and the discriminator on the other hand, which is trained along with the generator in an adversarial fashion.

While the objective component of the reward function ensures that the model selects for traits that maximize the specified heuristic, the job of the discriminator is to incentivise the samples to stay within distribution limitations of the initial data.

Generative adversarial networks (GANs) can be problematic to train, often suffering from mode-collapse, when the generator ends up exploiting repetitious patterns in samples. Or from the perfect discriminator problem, when a discriminator overwhelms the generator, preventing the generator from learning or improving its capacity. Mode-collapse [9] is a setback occurring in GANs where the generator learns to produce samples with decreased variety. When generating molecules in the form of SMILES, the perfect discriminator problem might translate to producing structures with much shorter string lengths when compared to the initial training set. This can lead to chemical characteristics such as molecular weight, the number of atoms, logP, and the topological polar surface area (TPSA) that differ substantially from the initial distribution of the training set. Such a degenerative process in the generated molecules is undesirable and can prevent discovery of novel, effective, and diverse compounds.

There are many ongoing efforts to study and improve the convergence properties in GANs. Some rely on modifying the loss functions, other paths of improvement rely on larger training sets or altering the discriminator or generator network. Our approach lies in utilizing a self-attention mechanism to improve the performance of the GAN in rewards such as novelty and validity.

Self-attention [23] on the other hand, proposed in the context of Image generation, exhibits a balance between the ability to model long-range dependencies and be computationally and statistically efficient. The self-attention module essentially calculates response at a position as a weighted sum of the features at all positions, where the weights – or attention vectors – are calculated with only a small computational cost.

In this paper, we introduce the work of self-attention mechanism to the field of Generative Adversarial Networks that operate on discrete sequential data to increase the performance of existing models. Self-Attention mechanism has been proposed in the field of Transformers [23] and GANs [8] on continuous data. In order to implement the above idea, we build on ORGAN [11], a recent work that successfully combines GANs and RL to apply the GAN framework to discrete sequential data and extend it towards domain-specific rewards. We tested the performance of the attention model on the same data as the original paper and produced improved results in novelty.

2. Related Work

Previous work has relied on specific modifications of the objective function to reach the desired properties. For example, Jaques et al., 2016[14] introduce penalties to unrealistic sequences, in absence of which RL can easily get stuck around local maxima which can be very far from the global maximum reward. Related applications by Ranzato et al., 2015 [21] and Li et al., 2016 [17] apply reinforcement learning to sequence generation in a NLP setting.

In the recent years, many methodologies have been proposed for de novo molecular generation. Ertl et al., 2017 [6] and Segler et al., 2017 [22] trained recurrent neural networks to generate drug-like molecules. [7] employed a variational autoencoder to build a latent, continuous space where property optimization can be made through surrogate optimization. Finally, Kadurin et al., 2017 [15] presented a GAN model for drug generation.

In 2018, Objective Reinforced Generative Adversarial Network for Inverse-design Chemistry, ORGAN [11] model has been proposed for optimizing various objectives to solve the inverse-design problem in Chemistry. The architecture of ORGANIC combines adversarial training and deep reinforcement learning (RL). Based on the SMILES string representation of the molecules, ORGANIC trains a generator model to create molecular structures that are penalized or rewarded by a discriminator model and a reward function which quantifies desired properties such as drug-likeliness. The discriminator attempts to classify proposed molecules as fake or real, based on a data distribution, essentially incentivizing the generator to create realistic samples

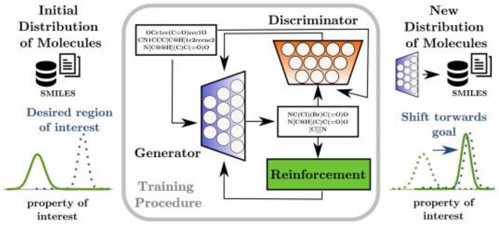

Figure 1. ORGANIC [11] illustration with a generator, a discriminator and a reinforcement metric. Arrows indicate the flow of inputs and outputs between networks.

Recently, attention mechanisms have become an integral part of models that must capture global dependencies Bahdanau et al., 2014 [1]; Xu et al. [25], 2015; Yang et al., 2016; Gregor et al., 2015 [10]; Chen et al., 2018 [4]). In particular, self-attention (Cheng et al. [4], 2016; Parikh et al., 2016 [18]), also called intra-attention, calculates the response at a position in a sequence by attending to all positions within the same sequence. Vaswani et al. [23] demonstrated that machine translation models could achieve state-of-the-art results by solely using a self-attention model. Parmar et al. [19] proposed an Image Transformer model to add self-attention into an autoregressive model for image generation. Wang et al. (Wang et al., 2018) formalized self-attention as a non-local operation to model the spatial-temporal dependencies in video sequences. In spite of this progress, self-attention has not yet been explored in the context of GANs. AttnGAN Xu et al., 2018 [26] uses attention over word embeddings within an input sequence, but not self-attention over internal model states). SAGAN learns to efficiently find global, long-range dependencies within internal representations of images.

SAGAN for Image generation proposed Self-Attention Generative Adversarial Networks (SAGANs) [8], which incorporate a self-attention mechanism into the GAN framework. The self-attention module is proven to be effective in modeling long-range dependencies in the context of image generation. In addition, the paper show that spectral normalization applied to the generator stabilizes GAN training and that two timescale update rule (TTUR) speeds up training of regularized discriminators. SAGAN achieves the state-of-the-art performance on class-conditional image generation on ImageNet.

3. Model

In this section, we elaborate on the GAN and RL setting based on ORGANIC and the self attention mechanism.

In Figure 1, the general scheme of ORGANIC is illustrated. It is a chemistry-orientated implementation of the Objective-Reinforced Generative Adversarial Networks paradigm which combines a Generative Adversarial Network with a Reinforcement Learning component.

The model is composed of three key elements: a Generator G, a Discriminator D and a Reinforcement component R. On one side, the discriminative network determines whether a molecule is likely to come from the initial distribution (positive) or not (negative). The reinforcement provides a quality metric R(x) [0, 1] that will quantify the desirability of a given molecule x, where 1 is meant to represent the desired shift in properties and 0 an undesired change. Finally, the generator has the task of generating molecules that maximizes the objective function which is a linear combination of the discriminator component and the reinforcement component, parametrized by a tunable parameter . In this way, the adversarial approach is meant to keep the generation of molecules similar to the initial distribution of data, while the reinforcement learning biases this generation towards the maximization of the reward (and, if the metric is well-defined, some desirable properties).

3.1. Data Representation

The molecules were represented as characters strings using the SMILES encoding and then hot-encoded as binary matrices. SMILES are able to capture the topology of a molecular species via defined grammar rules. In a string, the characters represent atoms, bond types, cycles and branches within a molecular graph. In this manner, the set of characters and rules already contain some heuristics about how molecules are built. However, this encoding also implies several inconveniences, such as the existence of invalid representations. SMILES strings can be valid 'NCc1cc[nH]n1" or invalid like "[C[[[N", meaning they can represent a molecule under the grammar defined by SMILES. Furthermore, the representation is non-unique, several strings can map to the same molecule (although there exists canonization algorithms).

The terms that arrive from the different attributes of SMILES are not mutually exclusive. This is an advantage in some cases, while a dis-advantage in others. The same molecule can have different SMILE notations. One such case of this is Ethanol. Ethanol can be written in 3 different formats (OCC, CCO, COC). However, the algorithm is written in such a way so that only one unique SMILE is formed from a given molecule. This is a beautiful way of converting all molecules to their corresponding string formats without losing a lot of information. Also, isomeric SMILES can be represented to represent various isomers of the given molecule. Configuration at the tetrahedral centres and double bond geometry can also be accurately represented by SMILES without the loss of information.

For the hot-encoding, we build a dictionary of m possible characters, each is assigned an index for the hot-encoding. For example, the ith character in a string x_i, will get encoded to a binary vector of length m, with 0's except in the index that maps to the character represented. A maximum length T is decided based on the training set and on the expected size of the string. In this way, each molecule is encoded to a n × T binary matrix, which can be converted back to a SMILES and then to a molecule.

In our work we employ the RDKit [28] package for manipulation and verification of SMILES.

3.2. Adversarial approach

Although the generation of valid SMILES can be trivial if one relies on simple permutations of carbons and oxygen atoms, such a strategy can lead to millions of different possible molecules which might not be of any interest for a given problem domain. In order to generate molecular species that resemble a given initial distribution, a key strategy is to make use of adversarial training. Generative Adversarial Networks (GANs) are a generative model that aims to minimize the divergence between a real data distribution p_{data} and the distribution p_{synth} generated by an implicit generative model G. The main idea is that two different neural networks play a game against each other: given an initial training distribution p_{data}, the generator G samples x from a distribution p_{synth} generated with random noise z, while a discriminator D looks at samples, either from p_{synth} or from p_{data}, and tries to classify their identity (y) as either real (x $\in p_{\text{data}})orfake(x \notin p_{\text{data}})$.

The GAN setting is then formulated as a zero sum game where the value is the cross-entropy loss between the discriminator's prediction and the true identity of the samples. This is implemented in practice as a min-max optimization problem, one model is optimized with respect to the performance of another model alternatively:

$$\min_{G} \max_{D} [\mathbf{E}_{x\sim p_{data}(x)}[\log D(x)] +$$
$$\mathbf{E}_{z\sim p_{synthetic}(z)}[\log(1 - D(G(z)))]]$$

In our setting the discriminator D is a Convolutional Neural Network (CNN) parameterized= by , while the generator G is an RNN parameterized by using LSTM (Long Short Term Memory) cells [30] that generates sequences $X_{1:T}$ = (x_1, ..., x_T) of length T, which in our case might represent valid SMILES strings. GANs are able to work remarkably well for continuous variable outputs since the gradient is propagated between networks during optimization. For discrete variable outputs, this proves difficult since these objects are sampled from a multinomial distribution, with probabilities given by the output of a softmax function, which is not differentiable and hence cannot be optimized with respect to each other easily.

The generator (which has previously been trained on the training set using MLE) generates batches of molecules, which are analyzed by the discriminator and the metric; then the former is trained to simultaneously fool the discriminator and maximize the reward.

3.3. Reinforcement Learning

It is only with recent developments that GANs have been adapted for discrete objects. One of the strategies is to bypass the generator differentiation problem by treating the generation of discrete sequences as a stochastic policy in an RL setting. The gradient is in this case approximated as a gradient policy update .

With a policy gradient, we treat G as an agent in an RL game where we consider states s, actions a from an action space A and a reward function Q. A state st is an already generated partial sequence of characters $X_{1:T}$. We have an action space 'A' that encompasses all possible characters to select for the next character x_{t+1}. Next a reward function Q(s, a) that represents the expected reward for taking action a in state s. Each episode is the completion of a fully generated sequence of fixed length T, which is rewarded with he function RT (X). The agent's stochastic policy is given by $G(x_{\text{t}}|X_{1:\text{t1}})$ and we wish to maximize its expected long term reward J():

$$J(\theta) = E\left[R_T \mid s_\theta, \theta\right] = \sum_{x_1 \in X} G_\theta\left(x_1 \mid s_\theta\right) \cdot Q\left(s_\theta, x_1\right)$$

For any full sequence $X_{1:T}$, we have $Q(s = X_{1:T}1, a = X_T) = RT(X)$. In order to calculate which action a is best for partial sequences at intermediate timesteps, we need to consider the expected future reward when the sequence is completed. To calculate Q in such cases, we perform N-time Monte Carlo search with the canonical rollout policy G.

That means from a partial sequence $X_{1:T}$, we sample stochastically from 1 to N, completed sequences $X_{1:T}$:Tn via the policy G. This formulation allows us to choose the source of completion reward R, either from the discriminator or from a quality metric. In ORGAN, the parameter controls the contribution of the source of each reward, where = 0 represents a complete RL approach and represents a complete GAN training. Typically one will want to use a value in-between.

The quality metric for RL can be defined from the myriad of functions designed for properties of interest in molecules. Some examples include Log P [24], Synthetic Accessibility Score [6], Natural Product-likeliness [5], Chemical Beauty (QED) [2] and presence or absence of substructures that can be calculated with chemoinformatic tools [16]. Complex properties such as HOMO-LUMO bandgap energies, Photoelectric Conversion Efficiency (PCE) and Redox potentials can be calculated with quantum chemistry methods. In practice, these can be estimated order of magnitude much faster, but less accurately, with machine learning regression tools such as Neural Networks and Gaussian Processes.

3.4. Attention Mechanism

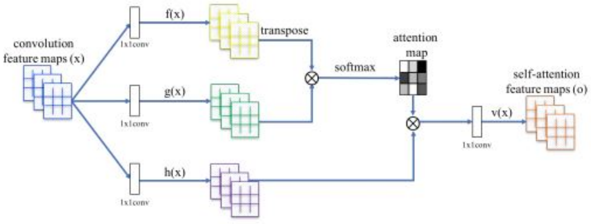

Figure 2. Attention mechanism as proposed by the SAGAN. The denotes matrix multiplication. The softmax operation is performed on each row.

Most GAN-based models (Radford et al., 2016; Salimans et al., 2016; Karras et al., 2018) for image generation are built using convolutional layers. Convolution processes the information in a local neighborhood, thus using convolutional layers alone is computationally inefficient for modeling long-range dependencies in images. In this section, we adapt the non-local model of (Zhang et al., 2018) to introduce self-attention to the GAN framework, enabling the discriminator to efficiently model relationships between widely separated spatial regions.

The image features from the previous hidden layer $x \in R^{\mathrm{C\ x\ N}}$ are first transformed into two feature spaces f, g to calculate the attention, where f(x) = W_fx, g(x) = W_gx

$$\beta_{j,i} = \frac{\exp\left(s_{ij}\right)}{\sum_{i=1}^{N}\exp\left(s_{ij}\right)}, where s_{ij} = f\left(x_i\right)^T g\left(x_j\right),$$

and j,i indicates the extent to which the model attends to the ith location when synthesizing the jth region. Here, C is the number of channels and N is the number of feature locations of features from the previous hidden layer. The output of the attention layer is o = (o_1, o_2, ..., o_j , ..., o_N) R(C×N) , where

$$o_j = v\left(\sum_{i=1}^{N}\beta_{j,i}h\left(x_i\right)\right), h\left(x_i\right) = W_h x_i, v\left(x_i\right) = W_v x_i$$

In addition, we further multiply the output of the attention layer by a scale parameter and add back the input feature map. Therefore, the final output is given by, $y_{\mathrm{i}} = \gamma o_{\mathrm{i}} + x_{\mathrm{i}}$, (3) where is a learnable scalar and it is initialized as 0. Introducing the learnable γ allows the network to first rely on the cues in the local neighborhood – since this is easier – and then gradually learn to assign more weight to the non-local evidence.

4. Experiments

In this section, the computational experiments and the data sets used are discussed. For the dataset, we have used 'small molecules' which constitute of a 5000 molecules subset of the dataset of roughly 134 thousand stable small molecules, which is itself a subset of all molecules with up to nine heavy atoms.

We tested the performance of the self attention model using two drug-likeness indicators: novelty and validity, both of which are contained in the [0,1] interval. GCC 9.1.0 with python 3.6.8 were used to carry out the experiments.

In both cases, a 200-epoch optimization was carried out.

4.1. Novelty

Novelty of SMILES reward awards 1.0 for a SMILES that's not encountered in the training set, but also is a valid SMILES following all the grammatical rules and awards 0.0 otherwise. The motive of this reward is to see if the model helps in developing the ASCII strings to new molecules and SMILES different from those in the training set.

4.2. Validity

The validity of SMILES reward awards 1.0 for a valid SMILES, which follows all the grammatical rules of the SMILES representation and awards 0.0 otherwise. The motive of this reward is to see if the model helps in developing ASCII strings which follow the correct grammatical rules of SMILES upon training and how self-attention can increase the performance.

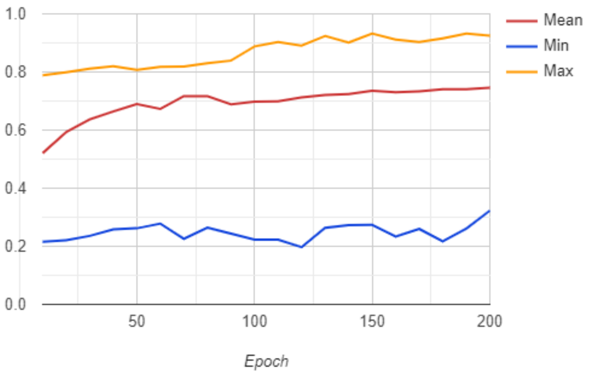

Figure 3. evolution of the mean score during the 200 epochs of training for novelty.

5. Results

The evolution of the training for 200 epochs is shown in Figure 3 for novelty. In comparison to the ORGANIC model, our model has consistently outperformed (as seen in Figure 4) during the 200 epochs and reaches a new peak at 0.75 mean reward. This supports the initial hypothesis that long range dependencies are better captured using an attention mechanism hence producing newer lead molecules.

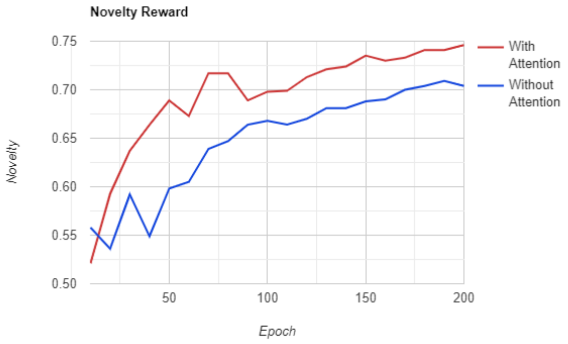

Figure 4. Comparison of Novelty reward with ORGANIC.

Similarly the evolution of the training is shown in Figure 5 for novelty. In comparison, the ORGANIC model has show a higher reward while training at almost every stage(as seen in Figure 6), however the self-attention model catches up in the end to reach the same peak mean reward at 0.775. This can be explained as a trade-off or the cost of adapting the attention mechanism to the model as it takes longer time to perfect the function of producing valid SMILES strings.

6. Conclusion

In this work, we have proposed the adaptation of self-attention mechanism to the Generative Adversarial Networks designed for molecule generation. The self-attention module is effective in modeling long-range dependencies and is shown to help the model generate molecules with higher novelty. This contributes to the initial hypothesis that the presence of elements in molecules are interdependent on each other and that relation needs to be explored.

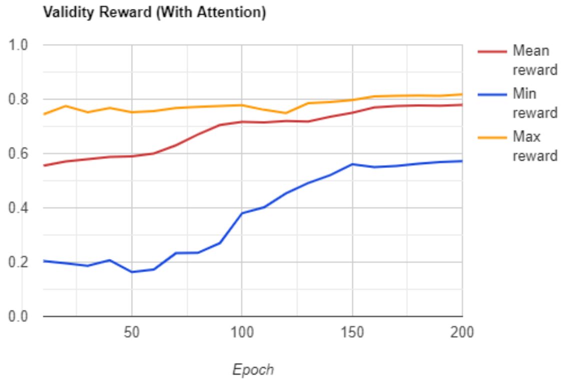

Figure 5. evolution of the mean score during the 200 epochs of training for validity.

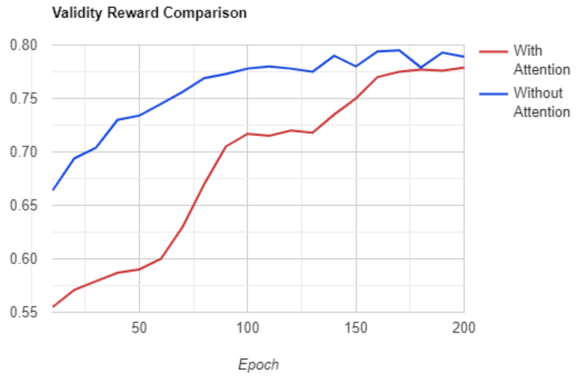

Figure 6. evolution of the mean score during the 200 epochs of training for validity.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] G Richard Bickerton, Gaia V Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L Hopkins. Quantifying the chemical beauty of drugs. *Nature chemistry*, 4(2):90–98, 2012.

[3] Tong Che, Yanran Li, Ruixiang Zhang, R Devon Hjelm, Wenjie Li, Yangqiu Song, and Yoshua Bengio. Maximum-Likelihood augmented discrete generative adversarial networks. 2017.

[4] Xi Chen, Nikhil Mishra, Mostafa Rohaninejad, and Pieter Abbeel. Pixelsnail: An improved autoregressive generative model. In *International Conference on Machine Learning*, pages 864–872. PMLR, 2018.

[5] Peter Ertl, Silvio Roggo, and Ansgar Schuffenhauer. Natural product-likeness score and its application for prioritization of compound libraries. *Journal of chemical information and modeling*, 48(1):68–74, 2008.

[6] Peter Ertl and Ansgar Schuffenhauer. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics*, 1(1):1–11, 2009.

[7] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. 2016.

[8] Han Zhang Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. http://arxiv.org/abs/1805.08318v2. Accessed: 2021-6-18.

[9] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. 2014.

[10] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. In *International Conference on Machine Learning*, pages 1462–1471. PMLR, 2015.

[11] Gabriel Lima Guimaraes, Benjamin Sanchez-Lengeling, Carlos Outeiral, Pedro Luis Cunha Farias, and Alán Aspuru-Guzik. Objective-reinforced generative adversarial networks (ORGAN) for sequence generation models. 2017.

[12] Johannes Hachmann, Roberto Olivares-Amaya, Sule Atahan-Evrenk, Carlos Amador-Bedolla, Roel S Sánchez-Carrera, Aryeh Gold-Parker, Leslie Vogt, Anna M Brockway, and Alán Aspuru-Guzik. The harvard clean energy project: Large-scale computational screening and design of organic photovoltaics on the world community grid. *J. Phys. Chem. Lett.*, 2(17):2241–2251, 2011.

[13] R Devon Hjelm, Athul Paul Jacob, Tong Che, Adam Trischler, Kyunghyun Cho, and Yoshua Bengio. Boundary-seeking generative adversarial networks. 2017.

[14] S Hochreiter and J Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997.

[15] Artur Kadurin, Sergey Nikolenko, Kuzma Khrabrov, Alex Aliper, and Alex Zhavoronkov. drugan: an advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico. *Molecular pharmaceutics*, 14(9):3098–3104, 2017.

[16] Greg Landrum. Rdkit documentation. *Release*, 1(1-79):4, 2013.

[17] Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*, 2016.

[18] Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*, 2016.

[19] N Parmar, A Vaswani, J Uszkoreit, Ł Kaiser, N Shazeer, A Ku, and D Tran. Image transformer. arxiv e-prints (feb. *arXiv preprint cs.CV/1802.05751*, 2018.

[20] Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. Learning to generate reviews and discovering sentiment. 2017.

[21] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

[22] Marwin HS Segler, Thierry Kogej, Christian Tyrchan, and Mark P Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS central science*, 4(1):120–131, 2018.

[23] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[24] Scott A Wildman and Gordon M Crippen. Prediction of physicochemical parameters by atomic contributions. *Journal of chemical information and computer sciences*, 39(5):868–873, 1999.

[25] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057. PMLR, 2015.

[26] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1316–1324, 2018.

[27] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. SeqGAN: Sequence generative adversarial nets with policy gradient. 2016.