ChemGAN challenge for drug discovery: can AI reproduce natural chemical diversity?

Mostapha Benhenda*

Abstract

Generating molecules with desired chemical properties is important for drug discovery. The use of generative neural networks is promising for this task. However, from visual inspection, it often appears that generated samples lack diversity. In this paper, we quantify this internal chemical diversity, and we raise the following challenge: can a nontrivial AI model reproduce natural chemical diversity for desired molecules? To illustrate this question, we consider two generative models: a Reinforcement Learning model and the recently introduced ORGAN. Both fail at this challenge. We hope this challenge will stimulate research in this direction.

Introduction

Drug discovery is like finding a needle in a haysack. The chemical space of potential drugs contains more than 10^{60} molecules. Moreover, testing a drug in a medical setting is time-consuming and expensive. Getting a drug to market can take up to 10 years and cost $2.6 billion (DiMasi, Grabowski, and Hansen 2016). In this context, computer-based methods are increasingly employed to accelerate drug discovery and reduce development costs.

In particular, there is a growing interest in AI-based generative models. Their goal is to generate new lead compounds *in silico*, such that their medical and chemical properties are predicted in advance. Examples of this approach include Variational Auto-Encoders (Gómez-Bombarelli et al. 2016), Adversarial Auto-Encoders (Kadurin et al. 2017a; Kadurin et al. 2017b), Recurrent Neural Networks and Reinforcement Learning (Jaques et al. 2017; Segler et al. 2017; Olivecrona et al. 2017), eventually in combination with Sequential Generative Adversarial Networks (Guimaraes et al. 2017; Benjamin et al. 2017).

However, research in this field often remains at the exploratory stage: generated samples are sometimes evaluated only visually, or with respect to metrics that are not the most relevant for the actual drug discovery process.

Rigorous evaluation would be particularly welcome regarding the **internal chemical diversity** of the generated samples. Generating a chemically diverse stream of molecules is important, because drug candidates can fail in many unexpected ways, later in the drug discovery pipeline.

Based on visual inspection, (Jaques et al. 2017, p. 8) reports that their Reinforcement Learning (RL) generative model tends to produce simplistic molecules. On the other hand, (Guimaraes et al. 2017, p.6, p.8) argues that their Objective-Reinforced Generative Adversarial Network (ORGAN) generates less repetitive and less simplistic samples than RL. However, their argument is also based on visual inspection and therefore, it remains subjective: our own visual inspection of the ORGAN-generated samples (available on the ORGAN Github:

`https://github.com/gablg1/ORGAN/tree/master/results/mol_results`) rather suggests that ORGAN produces molecules as repetitive and as simplistic as RL.

In this paper, we introduce a metric that quantifies the *internal chemical diversity* of the model output. We also submit a challenge:

Challenge: Is it possible to build a non-trivial generative model, with (part of) its output satisfying a non-trivial chemical property, such that the internal chemical diversity of this output is at least equal to the diversity found in nature for the same kind of molecules?

To illustrate this challenge, we compare RL and ORGAN generative models, with respect to the following chemical properties:

1. Being **active against the dopamine receptor D2**. The dopamine D2 receptor is the main receptor for all antipsychotic drugs (schizophrenia, bipolar disorder...).
2. **Druglikeness** as defined in (Guimaraes et al. 2017). We are interested in this property because we can use experimental results in (Guimaraes et al. 2017) to facilitate discussion. However, the notion of druglikeness in (Guimaraes et al. 2017) is different from the notion of Quantitative Estimation of Druglikeness (QED) (Bickerton et al. 2012), which is an index measuring different physico-chemical properties facilitating oral drug action.

*Startcrowd, Online Artificial Intelligence Lab. Email: mostaphabenhenda@gmail.com Discuss also the paper on the chat:
`https://gitter.im/Startcrowd/drugdiscovery`

Here, druglikeness is the arithmetic mean of the solubility (normalized logP), novelty (which equals 1 if the output is outside of the training set, 0.3 if the output is a valid SMILES in the training set, and 0 if the output is not a valid SMILES), synthesizability (normalized synthetic accessibility score (Ertl and Schuffenhauer 2009)) and conciseness (a measure of the difference of the length between the generated SMILES and its canonical representation).

We mention that recently, (Benjamin et al. 2017) considers an ORGAN with the QED definition of druglikeness. However, we also performed our own experiments with the QED property, and they did not affect our conclusions.

The metric of internal chemical diversity

Let a and b be two molecules, and m_a and m_b be their Morgan fingerprints (Rogers and Hahn 2010). Their number of common fingerprints is $|m_a \cap m_b|$ and their total number of fingerprints is $|m_a \cup m_b|$.

The Tanimoto-similarity T_s between a and b is defined by:

$$T_s(a, b) = \frac{|m_a \cap m_b|}{|m_a \cup m_b|}$$

Their Tanimoto-distance is:

$$T_d(a, b) = 1 - T_s(a, b)$$

We use rdkit implementation (Landrum 2017) of this distance.

Internal diversity

We define the *internal diversity* I of a set of molecules A of size $|A|$ to be the average of the Tanimoto-distance T_d of molecules of A with respect to each other. Formally, we have:

$$I(A) = \frac{1}{|A|^2} \sum_{(x,y) \in A \times A} T_d(x, y) \qquad (1)$$

For a sufficiently large set A, any sufficiently large subset $A' \subset A$, sampled with uniform probability, has the same internal diversity as A. This property follows from the law of large numbers. We can thus define the internal diversity of a generative model, by computing the internal diversity of a sufficiently large generated sample. This allows to formalize our challenge:

Challenge (restatement): Let N be the molecules observed in nature. Is there a non-trivial generative model G and a non-trivial chemical property P such that:

$$I(G \cap P) \geq I(N \cap P) \quad ? \qquad (2)$$

Internal chemical diversity is always smaller than 1 (because the Tanimoto-distance is smaller than 1), and it is usually much smaller. That's why we prefer this definition to the Tanimoto-variance of a set of molecules A, which is:

$$V(A) = \frac{1}{|A|^2} \sum_{(x,y) \in A \times A} T_d(x, y)^2 \qquad (3)$$

External diversity

A related notion is **external diversity**. Let A_1 and A_2 two sets of molecules. The *relative diversity* E of A_1, A_2 is defined by:

$$E(A_1, A_2) = \frac{1}{|A_1| \times |A_2|} \sum_{(x,y) \in A_1 \times A_2} T_d(x, y) \qquad (4)$$

The *external diversity* of a generative model is defined as the relative diversity between the training set and a sufficiently large generated sample.

External diversity essentially corresponds to the notion of diversity defined in (Guimaraes et al. 2017, p.5).[1]

A measure of the Tanimoto similarity between generated and natural molecules is also considered in (Segler et al. 2017, figures 7 and 12) (and their figure 11 considers the Levenshtein distance between them).

The main insight of our paper is to compare internal diversities of generated and natural molecules respectively, instead of considering the relative diversity between generated and natural molecules (and also, we measure this internal diversity with respect to the subset of molecules satisfying the property of interest).

We think measuring internal diversity is a good way to quantitatively capture the visually observed fact that generated molecules can be repetitive and simplistic (Guimaraes et al. 2017; Jaques et al. 2017).

Generative Models

Reinforcement Learning

As in the case of RL considered in (Guimaraes et al. 2017), the generator G_θ is a LSTM Recurrent Neural Network (Hochreiter and Schmidhuber 1997) parameterized by θ. G_θ generates SMILES (Simplified Molecular-Input Line-Entry System) sequences of length T (eventually padded with "_" characters), denoted by:

$$Y_{1:T} = (y_1, ..., y_T)$$

Let $R(Y_{1:T})$ be the reward function.

- For the case of dopamine D2 activity, we take:

$$R(Y_{1:T}) = P_{\text{active}}(Y_{1:T})$$

where $P_{\text{active}}(Y_{1:T})$ is the probability for $Y_{1:T}$ to be D2-active. This probability is given by the predictive model made in (Olivecrona et al. 2017)[2], and available online at

[1] The only difference is that in the definition of (Guimaraes et al. 2017, p.5), only a random subset of molecules of the training set is considered. For faster computations, we also consider a random subset of the training set (of 3000 samples).

[2] This reward function is slightly different than the function in (Olivecrona et al. 2017), which is: $-1 + 2 \times P_{\text{active}}$.

`https://github.com/MarcusOlivecrona/REINVENT/releases`

- For the case of druglikeness, we take:

$$R(Y_{1:T}) = L(Y_{1:T})$$

where $L(Y_{1:T})$ is the druglikeness of $Y_{1:T}$.

The generator G_θ is viewed as a Reinforcement Learning agent: its state s_t is the currently produced sequence of characters $Y_{1:t}$, and its action a is the next character y_{t+1}, which is selected in the alphabet $\mathcal{Y}$. The agent policy is: $G_\theta(y_{t+1}|Y_{1:t})$. It corresponds to the probability to choose y_{t+1} given previous characters $Y_{1:t}$.

Let $Q(s,a)$ be the action-value function. It is the expected reward at state s for taking action a and for following the policy G_θ, in order to complete the rest of the sequence. We maximize its expected long-term reward:

$$J(\theta) = \mathbb{E}[R(Y_{1:T})|s_0, \theta] = \sum_{y_1 \in \mathcal{Y}} G_\theta(y_1|s_0) Q(s_0, y_1)$$

For any full sequence $Y_{1:T}$, we have:

$$Q(s = Y_{1:T-1}, a = y_T) = R(Y_{1:T})$$

For $t < T$, in order to calculate the expected reward Q for $Y_{1:t}$, we perform a N-time Monte Carlo search with the rollout policy G_θ, represented as:

$$\text{MC}(G_\theta(Y_{1:t}, N)) = \{Y_{1:T}^1, ..., Y_{1:T}^N\}$$

where $Y_{1:t}^n = Y_{1:t}$ and $Y_{t+1:T}^n$ is randomly sampled via the policy G_θ.

For $t < T$, Q is given by:

$$Q(s = Y_{1:t-1}, a = y_t) = \frac{1}{N} \sum_{n=1}^{N} R(Y_{1:T}^n)$$

Objective-Reinforced Generative Adversarial Network (ORGAN)

To obtain an ORGAN, (Guimaraes et al. 2017) brings a Character-Aware Neural Language Model (Kim et al. 2016) D_ϕ parameterized by ϕ. Basically, D_ϕ is a Convolutional Neural Network (CNN) whose output is given to a LSTM. D_ϕ is fed with both training data and data generated by G_θ. It plays the role of a *discriminator*, to distinguish between the two: for a SMILES $Y_{1:T}$, the output $D_\phi(Y_{1:T})$ is the probability that $Y_{1:T}$ belongs to the training data.

For the case of dopamine D2-activity, the reward function becomes:

$$R(Y_{1:T}) = \lambda D_\phi(Y_{1:T}) + (1-\lambda) P_{\text{active}}(Y_{1:T})$$

and for the case of druglikeness:

$$R(Y_{1:T}) = \lambda D_\phi(Y_{1:T}) + (1-\lambda) L(Y_{1:T})$$

where $\lambda \in [0,1]$ is a hyper-parameter. For $\lambda = 0$, we get back the RL case, and for $\lambda = 1$, we obtain a Sequential Generative Adversarial Network (SeqGAN) (Yu et al. 2017).

The networks G_θ and D_ϕ are trained adversarially (Schmidhuber 1992; Goodfellow et al. 2014), such that the loss function for D_ϕ to minimize is given by:

$$\min_\phi \mathbb{E}_{Y\sim\text{train}}[\log D_\phi(Y)] + \mathbb{E}_{Y\sim\text{gen}}[\log(1 - D_\phi(Y))] \tag{5}$$

Experiments

As in (Guimaraes et al. 2017), we pre-train the models 240 epochs with Maximum Likelihood Estimation (MLE), on a random subset of **15k molecules from the ZINC database** of 35 million commercially-available compounds for virtual screening, used in drug discovery (Sterling and Irwin 2015). Then we further train the models with RL and ORGAN respectively, for 30 and 60 epochs more.

Dopamine D2 activity

In table 1, we show the proportion of valid SMILES output (Prop. Valid SMILES), the average probability of activity on dopamine D2 (Avg. P_a), the average internal diversity (Avg. int. div.), the proportion of molecules with probability of activity greater than 0.8 (Prop. $P_a > 0.8$), and most importantly, the average internal diversity among samples with probability of activity greater than 0.8. That's the most important column, because it is related with our open problem.

The averages are computed over the set of valid SMILES, whereas the proportions are computed over all the generated SMILES (both valid and invalid).

We compute those quantities for a D2-active set of 8324 molecules from ExCAPE-DB (Sun et al. 2017) (which is essentially the training set of the SVM classifier in (Olivecrona et al. 2017)) (DRD2), for the output of the Reinforcement Learning model after 30 epochs (RL 30) and 60 epochs (RL 60), and for the output of ORGAN with $\lambda = 0.04$ after 30 epochs and 60 epochs (ORGAN-0.04 30, ORGAN-0.04 60) and for $\lambda = 0.5$ after 60 epochs (ORGAN-0.5 60). All those outputs have 32k samples.

[TABLE 1 HERE]

The most interesting case is RL after 30 epochs. In this case, we can see that increasing the probability of D2 activity is contradictory with keeping diversity. After 30 epochs, internal diversity is still pretty good overall, even higher than the DRD2 diversity baseline.

However, when we only keep the molecules of interest, with $P_a > 0.8$, internal diversity dramatically drops to vanishingly small levels.

For ORGAN-0.04, results are mostly analogous to RL. We note that at 30 epochs, diversity for $P_a > 0.8$ is 2 orders of magnitude better than RL 30. However, it still remains one order of magnitude lower than the DRD2 baseline, and at 60 epochs, diversity has dropped to levels similar with RL.

For ORGAN-0.5, learning the D2 property still did not start after 60 epochs. The situation is analogous to the SeqGAN case ($\lambda = 1$) described in (Guimaraes et al. 2017):

	Prop. Valid SMILES	Avg. P_a	Avg. int. div.	Prop. $P_a > 0.8$	Avg. int. div. for $P_a > 0.8$
DRD2	0.996636	0.911519	0.089478	0.876367	0.081972
RL 30	0.379844	0.160777	0.112864	0.018906	$8.65864e-05$
RL 60	0.536	0.389979	0.014994	0.078438	0.000775
ORGAN-0.04 30	0.425375	0.097810	0.242544	0.013531	0.005826
ORGAN-0.04 60	0.604406	0.342687	0.028563	0.100969	0.000170
ORGAN-0.5 60	0.264687	0.006502	0.324884	0.000187	0.0

Table 1: Experimental results for probability of D2 activity P_a

high diversity, but no learning of the objective. In particular, that's why the internal diversity for $P_a > 0.8$ is indetectable: there are only 6 samples satisfying the desired property, among 32k.

The intermediate cases between $\lambda = 0.04$ and $\lambda = 0.5$ are analogous to either of them. It is hard to situate the tipping point, between the cases where training is just slow, and where training will never take off.

Here are 10 samples for ORGAN with $\lambda = 0.04$ after 30 epochs, selected such that $P_a > 0.8$ (most diverse case):

CCOCCNC[C@H]1CCCN1CCc1ccsc1
CCCOC[C@H]1Cc2ccccc21
CC[C@H]1CCNCOc2ccccc21
CC[C@H]1CCN(CCc2ccccc2)c1
CCCO[C@@H]1CCN(C)Cc2ccccc21
CCC[C@@H]1CCC[NH+]1CC[C@H]1CCCn1
CC[C@@H]1CCN(CCc2ccccc2)c1
CC[C@H]1CCN(Cc2ccccc2)c1
CCOC1CCN(CCCNCCCc2ccccc2)c1
CCCN1CCO[C@H]1C[C@@H]1CCOc2ccccc21

Druglikeness

In table 2, we show the proportion of valid SMILES output (prop. Valid SMILES), average druglikeness (Avg. L), the average internal diversity (Avg. int. div.), the proportion of molecules with druglikeness greater than 0.8 (Prop. $L > 0.8$), and most importantly, the average internal diversity among samples with druglikeness greater than 0.8. Again, that's the most important column, because it is related with our challenge.

Again, the total averages are computed over the set of valid SMILES, whereas the proportions are computed over all the generated SMILES (both valid and invalid).

We compute those quantities for the training set ZINC of 15k molecules (ZINC), which serves as a baseline, for the output of the Reinforcement Learning model after 200 epochs (RL 200) and for the output of ORGAN with $\lambda = 0.8$ after 200 epochs (ORGAN 200). Those outputs have 6400 samples.

[TABLE 2 HERE]

Results show that ORGAN indeed improves over RL, since it is able to raise internal diversity to detectable levels. However, ORGAN diversity still remains 2 orders of magnitudes lower than ZINC diversity when $L > 0.8$. ORGAN diversity also remains 3 orders of magnitude lower than the total diversity of ZINC, which corresponds to the level of internal diversity to which most eyes are used to. We conclude that both RL and ORGAN for $\lambda = 0.8$ fail to generate diverse molecules for this property.

Here are 10 SMILES samples from ORGAN for $\lambda = 0.8$ and 200 epochs:

Cc1ccccc1CCSc1ccccc1C
COCCc1ccccc1CCCCCCSC
CCCCCCn1cccc1CCCSC
COCc1ccccc1CCCc1ccccc1CC
CCCC(=O)CCCc1ccccc1CCCc1ccccc1C
COCCNC(=O)CCc1ccccc1
CCOC(=O)CCCCc1ccccc1CCN(C)CCCCc1ccccc1C
COCCCC(=O)CC(C)CCCCc1ccccc1C
CCCC(=O)CSCCC(=O)CCCCC(=O)OC
CCC(=O)COCCCCCCC(=O)CCCSC

Conclusion and future work

We conclude that both RL and ORGAN fail to match natural chemical diversity for desired molecules, although ORGAN is slightly better than RL. For future work, ORGAN training can be improved by considering 2 distinct problems:

- The perfect discriminator problem in adversarial training
- The imbalance between different objectives in Reinforcement Learning

The perfect discriminator problem

In ORGAN training, the discriminator D_ϕ quickly becomes perfect: it perfectly distinguishes between training data and generated data. In general, this situation is not very good for adversarial learning (Arjovsky and Bottou 2017). Here, the discriminator still teaches something to the generator. On average, according to the discriminator, the probability for a generated sample to belong to the training set still remains far from 0, although always smaller than 0.5. This probability is transmitted to the generator through the reward function.

However, not being able to 'fool' the discriminator, even in the SeqGAN case of $\lambda = 1$ (without any other objective), shows generator weakness: it shows inability to reproduce a plain druglike dataset like ZINC. Training a SeqGAN properly should be a first step towards improving ORGAN.

To achieve this, it might be possible to take a larger generator, to replace the discriminator loss in equation (5) with another function (like CramerGAN (Bellemare et al. 2017)),

	Prop. Valid SMILES	Avg. L	Avg. int. div.	Prop. $L > 0.8$	Avg. int. div. for $L > 0.8$
ZINC	1	0.661094	0.331222	0.020133	0.025986
RL 200	0.975625	0.917358	0	0.974844	0
ORGAN-0.8 200	0.943906	0.906885	0.000151	0.940625	0.000150

Table 2: Experimental results for Druglikeness L

and to use one-sided label smoothing (Salimans et al. 2016, p.4).

The discriminator might also overfit training data. Taking a larger training set could help, we took 15k samples here (less than 1MB), and this is small compared with training sets in Natural Language Processing. On the other hand, datasets in drug discovery rarely exceed 10k molecules, and therefore, it could also be interesting to look in the direction of low-data predictive neural networks (Altae-Tran et al. 2017).

Once adversarial training is stabilized, it might be interesting to replace all classifiers in the reward function with discriminators adversarially trained on different datasets. Various desired properties might be instilled into generated molecules with multiple discriminators. This might better transmit the chemical diversity present in the various training sets.

Imbalance in multi-objective RL

The main issue is the imbalance between the various objectives in the reward function, a problem occurring also in RL. Multi-objective reinforcement learning is a broad topic (for a survey, see (Roijers et al. 2013)).

A problem here is that with a weighted sum, the agent always focuses on the easiest objective, and ignores harder ones. Moreover, the relative difficulty between objectives evolves over time. For example, the average probability of D2 activity initially grows exponentially, and so this growth is small when this probability is near 0.

Using time-varying adaptive weights might help. Moreover, those weights might not necessarily be linear: For example, the reward function can be of the form $(x^\lambda + y^\lambda)^{1/\lambda}$, which converges towards $\min(x, y)$ as $\lambda \to -\infty$. Using an objective function of the form $\min(x, y)$ focuses the generator on the hard objective (but in our experiments, due to the perfect discriminator problem, it did not work).

Morever, in the reward function, a penalty can be introduced for newly generated molecules that are too similar with the generated molecules already having the desired properties.

In any case, the (varying) relative weights between different objectives must be determined automatically, and not through guesswork. In a drug discovery setting, a molecule must simultaneously satisfy a large number of objectives. For example, for an antipsychotic drug, it is not enough to be active against D2. The molecule must also pass toxicity and druglikeness tests. Moreover, to avoid side-effects, the molecule must not be active with D3, D4, serotonin, or histamine. That's a lot of objectives to include in the reward function.

Finally, there is also further work to improve the definition of internal diversity, in order to exclude trivial solutions (for example, a generative model reproducing the training set can also have high internal diversity). This will facilitate the attribution of financial prizes.

Acknowledgement

Computations were performed with 2 GPUs Nvidia Tesla M60, available from Microsoft Azure Free Trial.

References

[Altae-Tran et al. 2017] Altae-Tran, H.; Ramsundar, B.; Pappu, A. S.; and Pande, V. 2017. Low data drug discovery with one-shot learning. *ACS central science* 3(4):283–293.

[Arjovsky and Bottou 2017] Arjovsky, M., and Bottou, L. 2017. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862.*

[Bellemare et al. 2017] Bellemare, M. G.; Danihelka, I.; Dabney, W.; Mohamed, S.; Lakshminarayanan, B.; Hoyer, S.; and Munos, R. 2017. The cramer distance as a solution to biased wasserstein gradients. *arXiv preprint arXiv:1705.10743.*

[Benjamin et al. 2017] Benjamin, S.-L.; Carlos, O.; L., G. G.; and Alan, A.-G. 2017. Optimizing distributions over molecular space. an objective-reinforced generative adversarial network for inverse-design chemistry (organic). *ChemRxiv Preprint https://doi.org/10.26434/chemrxiv.5309668.v3*.

[Bickerton et al. 2012] Bickerton, G. R.; Paolini, G. V.; Besnard, J.; Muresan, S.; and Hopkins, A. L. 2012. Quantifying the chemical beauty of drugs. *Nature chemistry* 4(2):90–98.

[DiMasi, Grabowski, and Hansen 2016] DiMasi, J. A.; Grabowski, H. G.; and Hansen, R. W. 2016. Innovation in the pharmaceutical industry: new estimates of r&d costs. *Journal of health economics* 47:20–33.

[Ertl and Schuffenhauer 2009] Ertl, P., and Schuffenhauer, A. 2009. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *Journal of cheminformatics* 1(1):8.

[Gómez-Bombarelli et al. 2016] Gómez-Bombarelli, R.; Duvenaud, D.; Hernández-Lobato, J. M.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; and Aspuru-Guzik, A. 2016. Automatic chemical design using a data-driven continuous representation of molecules. *arXiv preprint arXiv:1610.02415.*

[Goodfellow et al. 2014] Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, 2672–2680.

[Guimaraes et al. 2017] Guimaraes, G. L.; Sanchez-Lengeling, B.; Farias, P. L. C.; and Aspuru-Guzik, A. 2017. Objective-reinforced generative adversarial networks (organ) for sequence generation models. *arXiv preprint arXiv:1705.10843.*

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

[Jaques et al. 2017] Jaques, N.; Gu, S.; Bahdanau, D.; Hernández-Lobato, J. M.; Turner, R. E.; and Eck, D. 2017. Sequence tutor: Conservative fine-tuning of sequence generation models with kl-control. In *International Conference on Machine Learning*, 1645–1654.

[Kadurin et al. 2017a] Kadurin, A.; Aliper, A.; Kazennov, A.; Mamoshina, P.; Vanhaelen, Q.; Khrabrov, K.; and Zhavoronkov, A. 2017a. The cornucopia of meaningful leads: Applying deep adversarial autoencoders for new molecule development in oncology. *Oncotarget* 8(7):10883.

[Kadurin et al. 2017b] Kadurin, A.; Nikolenko, S.; Khrabrov, K.; Aliper, A.; and Zhavoronkov, A. 2017b. drugan: an advanced generative adversarial autoencoder model for de-novo generation of new molecules with desired molecular properties in silico. *Molecular Pharmaceutics.*

[Kim et al. 2016] Kim, Y.; Jernite, Y.; Sontag, D.; and Rush, A. M. 2016. Character-aware neural language models. In *AAAI*, 2741–2749.

[Landrum 2017] Landrum, G. 2017. Rdkit: Open-source cheminformatics. *http://www.rdkit.org.*

[Olivecrona et al. 2017] Olivecrona, M.; Blaschke, T.; Engkvist, O.; and Chen, H. 2017. Molecular de novo design through deep reinforcement learning. *arXiv preprint arXiv:1704.07555.*

[Rogers and Hahn 2010] Rogers, D., and Hahn, M. 2010. Extended-connectivity fingerprints. *Journal of chemical information and modeling* 50(5):742–754.

[Roijers et al. 2013] Roijers, D. M.; Vamplew, P.; Whiteson, S.; and Dazeley, R. 2013. A survey of multi-objective sequential decision-making. *Journal of Artificial Intelligence Research* 48:67–113.

[Salimans et al. 2016] Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; and Chen, X. 2016. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, 2234–2242.

[Schmidhuber 1992] Schmidhuber, J. 1992. Learning factorial codes by predictability minimization. *Neural computation* 4(6):863–879.

[Segler et al. 2017] Segler, M. H.; Kogej, T.; Tyrchan, C.; and Waller, M. P. 2017. Generating focussed molecule libraries for drug discovery with recurrent neural networks. *arXiv preprint arXiv:1701.01329.*

[Sterling and Irwin 2015] Sterling, T., and Irwin, J. J. 2015. Zinc 15–ligand discovery for everyone. *J. Chem. Inf. Model* 55(11):2324–2337.

[Sun et al. 2017] Sun, J.; Jeliazkova, N.; Chupakin, V.; Golib-Dzib, J.-F.; Engkvist, O.; Carlsson, L.; Wegner, J.; Ceulemans, H.; Georgiev, I.; Jeliazkov, V.; et al. 2017. Excape-db: an integrated large scale dataset facilitating big data analysis in chemogenomics. *Journal of Cheminformatics* 9(1):17.

[Yu et al. 2017] Yu, L.; Zhang, W.; Wang, J.; and Yu, Y. 2017. Seqgan: sequence generative adversarial nets with policy gradient. In *AAAI-17: Thirty-First AAAI Conference on Artificial Intelligence*, volume 31. Association for the Advancement of Artificial Intelligence.