Regularizing Variational Autoencoders for Molecular Graph Generation

Xin Li[1], Xiaoqing Lyu[1], Hao Zhang[2], Keqi Hu[3], and Zhi Tang[1]

[1] Peking University, Beijing, China
{l_x, lvxiaoqing, tangzhi}@pku.edu.cn
[2] Beijing Institute of Technology, Beijing, China
gcrth@outlook.com
[3] China University of Mining & Technology, Beijing, China
keqihu@student.cumtb.edu.cn

Abstract. Deep generative models for graphs are promising for being able to sidestep expensive search procedures in the huge space of chemical compounds. However, incorporating complex and non-differentiable property metrics into a generative model remains a challenge. In this work, we formulate a differentiable objective to regularize a variational autoencoder model that we design for graphs. Experiments demonstrate that the regularization performs excellently when used for generating molecules since it can not only improve the performance of objectives optimization task but also generate molecules with high quality in terms of validity and novelty.

Keywords: Variational autoencoder · Regularization · Molecule generation · Graph generation.

1 Introduction

Generating molecules with desired properties is a challenging task with important applications such as drug design. In the last few years, considerable works [8] [13] [9] [2] using deep generative models including variational autoencoders (VAE) and generative adversarial networks (GAN) for molecule generation make use of the domain specific SMILES representation of molecules [23], a linear string notation to describe molecular structures. However, the main drawback of SMILES is that there is difficulty in capturing molecular similarity since small changes can result in drastically different structures. This shortcoming prevents some generative models from learning smooth latent variables. With recent progress in the area of deep learning on graphs [4] [7] [22] [14] [21], deep generative models for molecular graphs are attracting surging interests since the graph representation can overcome limitations of the SMILES [11] [3] [15] [22] [16].

In the task of molecular graph generation, one of the key challenges lies in the difficulty of incorporating highly complex and non-differentiable property metrics into a generative model. The two main strategies to achieve this
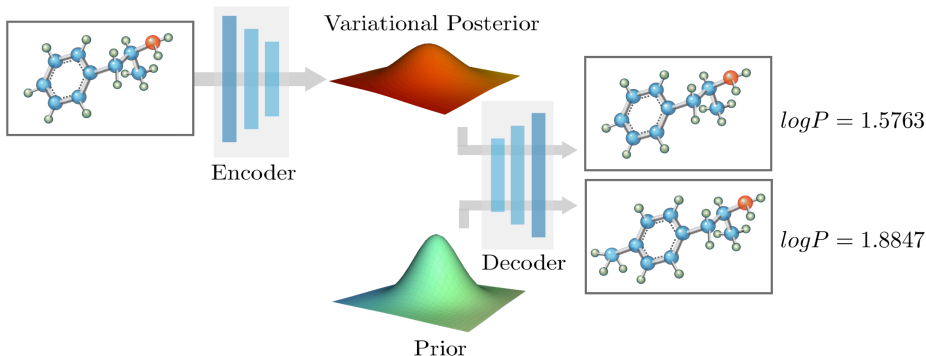
Fig. 1. Overview of the proposed OpVAE. The bottom flow depicts the regularization that encourages decoder to generate molecules with desired properties, which is detailed in Section 4.

end remain reinforcement learning-based and Bayesian optimization-based approaches. The reinforcement learning-based methods [9] [3] use reinforcement learning-based objective to provide a gradient to the policy towards the desired properties, which requires an extra reward network to predict the immediate reward. However, the extra network may arise the convergence difficulties. Besides, the reward network may not be able to get the correct prediction. The Bayesian optimization-based approaches[11] [13] perform Bayesian optimization to navigate into regions of latent space that decode into molecules with particular properties. Specifically, such methods can be divided into two phases. The first phase will focus on training a generative model. During the second phase, the Bayesian optimization is performed in the latent space. So the performance of Bayesian optimization during the second phase depends largely on the smoothness of the latent space learned during the previous phase.

In this work, we design a variational autoencoder for matrix representations of graphs. We then formulize a regularizer to encourage the generation of molecules with desired properties, which avoids extra networks and the requirement of smoothness of latent space. Monte Carlo approximation of the regularizer is used in the training procedure. Since the approximation is differentiable, we can train the model by stochastic gradient optimization methods. We demonstrate the effectiveness of our framework with two benchmark molecule datasets to generate molecules with desired properties.

2   Related Work

Recently, there has been significant advances in molecule generation. Previous works [8] [13] [9] [2] have explored the generative models on SMILES. Gómez-Bombarelli et al. [8] built generative models with recurrent neural networks. Kusner et al. [13] utilized a parse tree from a context-free grammar to improve

the validity of generated molecules. Dai et al. [2] took a step further towards the validity by enforcing constraints on the generative model. In addition to the above VAE-based works, Guimaraes et al. [9] used GAN to address the generation. For graph representations, Simonovsky et al. [22] have explored generating molecular graphs by extending VAE. Jin et al. [11] proposed a generative model by combining a tree-structured scaffold with original graphs. Liu et al. [15] incorporated chemical constraints into generative models by specifying a generative procedure and employing masking. Ma et al. [16] reformulated the constrained objective of VAE by constructing Lagrangian function.

3 Model

3.1 Variational Autoencoder

We adopt AEVB algorithm [12] to learn a generative model $p_\theta(G|z)$ using a particular encoder $q_\phi(z|G)$, which is an approximation of actual posterior $p_\theta(z|G)$ that is intractable. The objective is to maximize the evidence lower bound (ELBO) with respect to θ and ϕ:

$$\mathcal{L}_{ELBO} = -D_{KL}(q_\phi(z|G)||p_\theta(z)) + \mathbb{E}_{q_\phi(z|G)}[\log p_\theta(G|z)]. \tag{1}$$

The first term as a regularization is the divergence of the variational posterior $q_\phi(z|G)$ from the prior $p_\theta(z)$[4], which allows for learning more general latent representations instead of simply encoding an identity mapping. The expectation term, interpreted as negative reconstruction loss, is maximized when $p(G|z)$ assigns a high probability to the observed G.

3.2 Molecules as Graphs

Each molecule can be represented by an undirected graph with a set of labeled nodes associated with the atoms and a set of labeled edges associated with bonds between atoms. We restrict the domain to a collection of molecular graphs which have at most N nodes, $T-1$ node types and $R-1$ edge types.

The decoder outputs a matrix $\widetilde{X} \in R^{N \times T}$ and a tensor $\widetilde{A} \in R^{N \times N \times R}$, which are denoted by $\widetilde{G} = (\widetilde{A}, \widetilde{X})$. The row $\widetilde{X}(i,:)$ is a categorical distribution over the type of node i, which satisfy $\sum_{t=0}^{T} \widetilde{X}(i,t) = 1$. $\widetilde{X}(i,0)$ is the probability that node i is nonexistent. Similarly, the fiber $\widetilde{A}(i,j,:)$ is a categorical distribution over the edge type between nodes i and j, which satisfy $\sum_{r=0}^{R} \widetilde{A}(i,j,r) = 1$. $\widetilde{A}(i,j,0)$ is the probability that the edge between nodes i and j is nonexistent.

We assume that the node type and the edge type are independent. The tuple $\widetilde{G}$ hence becomes a random graph model. A one-hot $G = (A, X)$ can be sampled via categorical sampling from $\widetilde{G} = (\widetilde{A}, \widetilde{X})$.

---

[4] The prior is a standard normal in this paper.

3.3   Encoder and Decoder

In this paper, the encoder is parameterized as a diagonal normal distribution $q(z|G) = \mathcal{N}(z; \mu, \psi)$ with covariance matrix $\psi = \text{diag}(\sigma^2)$, where μ and σ are outputs of the encoding graph neural networks.

Suppose that $h_i^{(l)}$ is the hidden state of the node i at layer l. We define the following layer-wise propagation rule similar to [21] for the signal $h_i^{(l+1)}$ of the node i :

$$h_i^{(l+1)} = \sigma \Big(\sum_{r=1}^{R-1} \sum_{j=1}^{N} \frac{\widehat{A}(i,j,r)}{c_{i,r}} f_r^{(l)}(h_j^{(l)}, x_i) \Big), \tag{2}$$

where $\widehat{A}(:,:,r) = A(:,:,r) + I$ for each edge label r, with I being the identity matrix, which means a self-connection between layers is added for each edge type. The messages from neighbors (including the node i itself) that are obtained by an edge type-specific affine function $f_r^{(l)}$ are then accumulated. Besides linear transformation, we utilize normalization constant $c_{i,r}$ to ensure that the accumulation of messages will not completely change the scale of the feature representations. In this paper, $c_{i,r} = |\mathcal{N}_{i,r}|$, where $\mathcal{N}_{i,r}$ is the set of neighbors of the node i (including i itself) and the edge label between the node i and node $j \in \mathcal{N}_{i,r} \setminus \{i\}$ is r. Finally, the normalized messages are passed through an element-wise activation function $\sigma(\cdot)$ such as ReLU [17]. Note that R and N have the same meaning as in Section 3.2. It remains to define the initial hidden state at the first layer:

$$h_i^{(1)} = \sigma \Big(\sum_{r=1}^{R-1} \sum_{j=1}^{N} \frac{\widehat{A}(i,j,r)}{c_{i,r}} f_r^{(0)}(x_i) \Big), \tag{3}$$

where $x_i = X(i,:)$.

For graph-level outputs, we consider the following aggregation method proposed by [14] after $L - 1$ layers of propagation:

$$h_G = \tanh \Big(\sum_{i=1}^{N} \sigma(v(h_i^{(L)}, x_i)) \odot \tanh(u(h_i^L, x_i)) \Big), \tag{4}$$

where v and u are neural networks that take the concatenation of h_i^L and x_i as input and their output layers are both linear. $\sigma(v(h_i^{(L)}, x_i))$ is explained as a soft attention mechanism to decide how relevant node i can be to the current molecule generation task. We then perform an element-wise multiplication $\odot$ and take a sum over all weighted output vectors of the nodes to obtain the graph-level representation h_G. Finally, the μ and σ are generated from two multi-layer perceptrons (MLPs).

As mentioned in Section 3.1, the decoder draws latent variables z from the variational posterior $q_\sigma(z|G)$ or the prior $p_\theta(z)$ and outputs a random graph model $\widetilde{G}$ where a graph G is sampled. This is simply done by MLPs in this paper.

4   Regularization

4.1   Formularization

In this section, we design an interpretable regularizer to encourage the generation of molecule with desired properties in this paper. The differentiable regularization term can provide a gradient to the decoder towards the desired metrics. More details will be discussed after the review of several traditional approaches to calculating the chemical properties.

- **LogP:** The octanol-water partition coefficient (logP) serves as a measure of lipophilicity. Developed by [6], the atom-based method is an effective approach to calculating logP, which assigns to the individual atoms in the molecule additive contributions to molecular logP. The logP of small molecules can be calculated as the sum of the contributions of each of the atoms in the molecules. Since the high accuracies have been achieved in [24], where the 68 atomic contributions to logP have been determined, the approach is a common and standard approach to calculating logP.

- **SAS:** The synthetic accessibility score (SAS) [5] has been used as a measure to estimate ease of synthesis of molecules. It is calculated based on a combination of fragment contributions and a complexity penalty [5]. Between the two constituent parts of SAS, the fragmentScore is calculated as a sum of contributions of all fragments in the molecule divided by the number of fragments in this molecule, whereas the complexityPenalty is calculated as a combination of structural features such as rings, stereo centers, etc.

The central idea of this paper is to use regularization that encourages the decoder to generate molecules with desired properties to formulate a differentiable objective. Let S be the property to be calculated. It is justifiable that the expectation of S with respect to the distribution $p_\theta(G|z)$ is utilized as a regularizer. The expectation may then be formally written as

$$\mathbb{E}p_\theta(G|z)[S] = \sum_G p_\theta(G|z)S_G, \tag{5}$$

By analyzing the calculation schemes of logP and SAS, we can unify these methods into a single common framework. A specific substructure in a molecular graph is called a pattern. Each pattern is associated with one additive contribution. The property of a molecule G to be calculated is the summation of all the contributions for the patterns that occur in this molecule. For the calculation of logP in [6], each one of 68 atomic types can be regarded as a specific pattern with one central atom, its neighboring atoms and the bonds between the central atom and its neighbors. For the calculation of SAS in [5], patterns include the fragments, rings, etc. For any properties to be calculated, one specific pattern is associated with one contribution. Let the set of possible patterns be denoted as

---
[5] **SAS = fragmentScore − complexityPenalty**

Q. Given a molecular graph G, the properties mentioned above can be calculated according to

$$S_G = \sum_{q \in Q} n_q c_q, \tag{6}$$

where S_G is the property of molecule G, n_q is the number of occurrences of the pattern q, and c_q is the contribution for pattern q.

By combining Equation 5 with Equation 6, we have

$$\mathbb{E}_{p_\theta(G|z)}[S] = \sum_G \sum_{q \in Q} p_\theta(G|z) n_q c_q, \tag{7}$$

We let $\mathbb{E}_{p_\theta(G|z)}[n_q] = \sum_G p_\theta(G|z) n_q$, which is the expectation of number n_q with respect to the distribution $p_\theta(G|z)$. $E_{p_\theta(G|z)}[n_q]$ can also be interpreted as the probability of pattern q of a graph sampled from $p_\theta(G|z)$. Let $E_{p_\theta(G|z)}[n_q] = p_q$ for simplicity. We obtain

$$\mathbb{E}_{p_\theta(G|z)}[S] = \sum_{q \in Q} p_q c_q, \tag{8}$$

Equation 8 indicates that the expectaion can be evaluated as long as the problem of computing p_q is solved. However, evaluating p_q is computationally expensive. In practice, we can appeal to Monte Carol approximation for evaluating the expectation of S, which is differentiable with respect to θ. More details will be discussed in Section 4.2.

4.2   Training

We train our model to maximize the following objective function:

$$\mathcal{L} = \mathcal{L}_{ELBO} + \mathbb{E}_{p_\theta(G|z)}[S]. \tag{9}$$

As done in [12], reparameterized trick and Monte Carlo gradient estimator are employed for evaluating $\mathcal{L}_{ELBO}$. In order to evaluate $\mathbb{E}_{p_\theta(G|z)}[S]$, we also consider Monte Carol approximation as mentioned above. Latent variables are first sampled from the prior $p_\theta(z)$. The decoder takes z as input and outputs $\widetilde{G}$. Then we sample patterns from $\widetilde{G}$. For each $q^{(m)} \sim \widetilde{G}$, we further assume that an occurrence of a pattern can be represented by a 2-tuple $q^{(m)} = (V^{(m)}, E^{(m)})$, where $V^{(m)}$ is the set of atoms in this occurrence, $E^{(m)}$ is the set of bonds in this occurrence. Under the indenpendence assumption in Section 3.2, the probability $p_{q^{(m)}}$ of a pattern for one specific occurrence is given by

$$p_{q^{(m)}} = \prod_{it \in V^{(m)}} \prod_{ijr \in E^{(m)}} p_{it} p_{ijr}, \tag{10}$$

where p_{it} is an element of $\widetilde{X}$, i.e., $\widetilde{X}(i,t)$ and p_{ijr} is an element of $\widetilde{A}$, i.e., $\widetilde{A}(i,j,r)$. The spirit is that the atom it is represented by node i in $\widetilde{X}$ and its

Table 1. Comparison with baselines in terms of validity and novelty on QM9 and ZINC. The rows "% Valid" and "% Novel" are the validity and novelty in percentages, respectively. The results of baselines are copied from [22] and [16].

Method	QM9		ZINC	
	% Valid	% Novel	% Valid	% Novel
OpVAE	**100.00**	**99.97**	**100.00**	**100.00**
CVAE	10.30	90.00	0.70	**100.00**
GVAE	60.20	80.90	7.20	**100.00**
GraphVAE	55.70	76.00	57.10	71.90
SeVAE	96.60	97.50	34.90	**100.00**

label is t, which is a similar explanation to the bond ijr. Because p_{it} and p_{ijr} are differentiable with respect to θ, parameters can be updated by using stochastic gradient ascent. Note that the latent variable z for evaluating $\mathbb{E}_{p_\theta(G|z)}[S]$ is sampled from the prior $p_\theta(z)$ as shown in the bottom flow of Figure 1.

5   Experiments

5.1   Datasets

Two benchmark datasets are used for the experiments. **QM9**[19] is a subset of the massive 166 billion organic molecules GDB-17 chemical database [20]. The dataset contains about 134K molecular graph of up to 9 heavy atoms with 4 distinct atomic numbers (carbon, oxygen, nitrogen and fluorine) and 4 bond types. **ZINC** [10] is a curated set that contains about 250K drug molecules of up to 38 heavy atoms with 9 distinct atomic numbers and 4 bond types.

5.2   Quality of the Generated Molecules

Baselines.   We compared our method OpVAE against CVAE [8], GVAE [13], GraphVAE [22] and SeVAE [16]. For evaluation, we sampled 1000 latent vectors from the prior and performed maximum-likelihood decoding for each one.

Evaluation Measures.   We use the following 2 statistics to evaluate the quality of the generated molecules. **Validity** is defined as the ratio between the number of valid and all generated molecules sampled from the prior. **Novelty** is defined as the ratio between the number of valid samples that don't occur in the training set and the number of all valid samples.

Comparison with Baselines.   Table 1 reports the results obtained by the proposed method and baselines. our approach OpVAE shows a significant improvement over its competitors in terms of validity and novelty since OpVAE is not designed to boost the validity or novelty percentage. For the validity,

Table 2. Optimization for different objectives. The results of baselines are copied from [22] and [16]. The Naive VAE is trained to maximize $\mathcal{L}_{ELBO}$.

Objective	OpVAE	VAE (Naive)	ORGAN	MolGAN
LogP	**0.96**	0.51	0.55	0.89
SAS	**1.00**	0.67	0.83	0.95

an intuitive explanation might be that the regularization invests much effort in forcing decoder to generate patterns with higher properties and these patterns are chemically valid, which encourages sampled molecular graphs that implicitly contain the patterns to be valid. In the same way, novel molecules are obtained when the encoder explores the possibilities for generating patterns with higher properties.

5.3   Objectives Optimization

In order to demonstrate the effectiveness of the proposed regularization, following [9] and [3], we chose to optimize the objectives LogP and SAS and compare against their works. We trained our model over full QM9 dataset for 30 epochs similarly to the experiments performed by [3] but differently from those of [9], where the model is trained on 5k QM9 subset. We normalized all scores within $[0, 1]$ by using the codes [6] of [3]. As shown in table 2, SeVAE beats both Mol-GAN and ORGAN in terms of objective scores, which proves that the proposed regularization is very effective.

6   Conclusion

Incorporating complex and non-differentiable property metrics into deep generative models is a challenging subject. We built graph neural networks into VAE for generating molecular graph. By resorting to a differentiable regularizer, we addressed the property optimization. We introduced several metrics to validate the quality of our proposed method. The advantages of the proposed method are reflected in experimental results.

Acknowledgement

---

[6] Available at https://github.com/nicola-decao/MolGAN

References

1. Bengio, S., Wallach, H.M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.): Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada (2018), http://papers.nips.cc/book/advances-in-neural-information-processing-systems-31-2018
2. Dai, H., Tian, Y., Dai, B., Skiena, S., Song, L.: Syntax-directed variational autoencoder for structured data. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net (2018), https://openreview.net/forum?id=SyqShMZRb
3. De Cao, N., Kipf, T.: Molgan: An implicit generative model for small molecular graphs. arXiv preprint arXiv:1805.11973 (2018)
4. Duvenaud, D.K., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., Adams, R.P.: Convolutional networks on graphs for learning molecular fingerprints. In: Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R. (eds.) Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. pp. 2224–2232 (2015), http://papers.nips.cc/paper/5954-convolutional-networks-on-graphs-for-learning-molecular-fingerprints
5. Ertl, P., Schuffenhauer, A.: Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. J. Cheminformatics **1**, 8 (2009), https://doi.org/10.1186/1758-2946-1-8
6. Ghose, A.K., Crippen, G.M.: Atomic physicochemical parameters for three-dimensional-structure-directed quantitative structure-activity relationships. 2. modeling dispersive and hydrophobic interactions. Journal of Chemical Information and Computer Sciences **27**(1), 21–35 (1987)
7. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: Precup and Teh [18], pp. 1263–1272, http://proceedings.mlr.press/v70/
8. Gómez-Bombarelli, R., Duvenaud, D.K., Hernández-Lobato, J.M., Aguilera-Iparraguirre, J., Hirzel, T.D., Adams, R.P., Aspuru-Guzik, A.: Automatic chemical design using a data-driven continuous representation of molecules. CoRR **abs/1610.02415** (2016), http://arxiv.org/abs/1610.02415
9. Guimaraes, G.L., Sanchez-Lengeling, B., Farias, P.L.C., Aspuru-Guzik, A.: Objective-reinforced generative adversarial networks (ORGAN) for sequence generation models. CoRR **abs/1705.10843** (2017), http://arxiv.org/abs/1705.10843
10. Irwin, J.J., Sterling, T., Mysinger, M.M., Bolstad, E.S., Coleman, R.G.: Zinc: a free tool to discover chemistry for biology. Journal of chemical information and modeling **52**(7), 1757–1768 (2012)
11. Jin, W., Barzilay, R., Jaakkola, T.: Junction tree variational autoencoder for molecular graph generation. arXiv preprint arXiv:1802.04364 (2018)
12. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Bengio, Y., LeCun, Y. (eds.) 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings (2014), http://arxiv.org/abs/1312.6114
13. Kusner, M.J., Paige, B., Hernández-Lobato, J.M.: Grammar variational autoencoder. In: Precup and Teh [18], pp. 1945–1954, http://proceedings.mlr.press/v70/kusner17a.html

14. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.S.: Gated graph sequence neural networks. In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings (2016), https://iclr.cc/archive/www/doku.php%3Fid=iclr2016:accepted-main.html
15. Liu, Q., Allamanis, M., Brockschmidt, M., Gaunt, A.: Constrained graph variational autoencoders for molecule design. In: Advances in Neural Information Processing Systems. pp. 7795–7804 (2018)
16. Ma, T., Chen, J., Xiao, C.: Constrained generation of semantically valid graphs via regularizing variational autoencoders. In: Bengio et al. [1], pp. 7113–7124, http://papers.nips.cc/paper/7942-constrained-generation-of-semantically-valid-graphs-via-regularizing-variational-autoencoders
17. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Fürnkranz, J., Joachims, T. (eds.) Proceedings of the 27th International Conference on Machine Learning (ICML-10), June 21-24, 2010, Haifa, Israel. pp. 807–814. Omnipress (2010), https://icml.cc/Conferences/2010/papers/432.pdf
18. Precup, D., Teh, Y.W. (eds.): Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017, Proceedings of Machine Learning Research, vol. 70. PMLR (2017), http://proceedings.mlr.press/v70/
19. Ramakrishnan, R., Dral, P.O., Rupp, M., Von Lilienfeld, O.A.: Quantum chemistry structures and properties of 134 kilo molecules. Scientific data **1**, 140022 (2014)
20. Ruddigkeit, L., Van Deursen, R., Blum, L.C., Reymond, J.L.: Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. Journal of chemical information and modeling **52**(11), 2864–2875 (2012)
21. Schlichtkrull, M., Kipf, T.N., Bloem, P., Van Den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: European Semantic Web Conference. pp. 593–607. Springer (2018)
22. Simonovsky, M., Komodakis, N.: Graphvae: Towards generation of small graphs using variational autoencoders. In: International Conference on Artificial Neural Networks. pp. 412–422. Springer (2018)
23. Weininger, D.: Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. Journal of Chemical Information and Computer Sciences **28**(1), 31–36 (1988), https://doi.org/10.1021/ci00057a005
24. Wildman, S.A., Crippen, G.M.: Prediction of physicochemical parameters by atomic contributions. Journal of Chemical Information and Computer Sciences **39**(5), 868–873 (1999), https://doi.org/10.1021/ci990307l