Directional Variational Transformers for continuous molecular embedding

Tushar Gadhiya
Infocusp Innovations Pvt Ltd
tushar@infocusp.com

Falak Shah
Infocusp Innovations Pvt Ltd
falak@infocusp.com

Nisarg Vyas
Infocusp Innovations Pvt Ltd
nisarg@infocusp.com

Vahe Gharakhanyan
Google LLC
vaheg@google.com

Julia Yang
Google LLC

Alexander Holiday
Google LLC
aholiday@google.com

Abstract

We propose a Transformer based variational autoencoder model for generating a continuous latent space of molecules. The encoder is an attention-based model that directly incorporates the molecule's three-dimensional structure while remaining invariant to translations and rotation. The decoder is a conditional transformer model that generates SMILES representation of the molecule solely from its latent vector. We also propose a new molecular representation of a molecule called Bond String and Position (BSP), which represents a molecule as a sequence of bonds and their three-dimensional positions. We investigate the validity of the generated latent space and, by combining the learned chemical space with a Bayesian optimization system, show that we are able to efficiently locate molecules with specific target properties.

1 Introduction

Designing novel molecules with specific target properties is a central problem in materials science. Traditionally, this process would be guided by expert intuition, but the field is increasingly relying on machine learning to accelerate the search through chemical space. This has spurred the development of a wide array of models trained to predict various molecular properties, the accuracy of which is largely governed by the choice of molecular representation and the model architecture. Molecular representations largely fall into two categories: property based and model based.

Property based representations capture basic features about a molecule by either concatenating values from a lookup table of molecular properties (e.g. the viscosity of a collection of liquids), or computing basic values about the structure (e.g. tabulating the number of methyl groups in each molecule) [1–4]. While these featurizations capture enough detail to be useful benchmarks for new approaches [5], they are fundamentally limited by the availability of chemical data since each property included in the feature must be present for every molecule in the dataset.

Model based fingerprints generally use Graph Neural Networks (GNNs), Transformers or Deep Neural Networks (DNN) to create vector representations. GNNs capture molecular structure by operating directly on the molecular graph [6], while Transformers treat molecules as text by using string representations as input. The most popular of these string representations is SMILES [7], but SELFIES representation [8] has recently gained some interest. Transformers have been successfully used to predict reaction products given reactants [9], and have also been pre-trained for property prediction tasks using techniques from natural language processing [10]. While string representations may be flexible enough to describe an arbitrary molecule, they will fail to capture its rich three-

ELLIS Workshop on Machine Learning for Molecules (ML4Molecules 2022).

dimensional (3D) structure. Incorporating this structural information has been shown to improve model performance on property prediction tasks [11–13].

Here we propose a Transformer-based model that is trained with a Variational Autoencoder (VAE) objective. VAEs learn smooth, low-dimesional embeddings of their inputs, and have recently been applied within the field of chemistry to generate continuous latent spaces of molecules [14–16]. We experimented with model variants that operated on either graphs or sequences, and present results for both. We also propose a novel molecular representation that captures 3D structure, termed Bond String and Position (BSP), which incorporates both the chemical makeup (bond string) and the 3D conformation (bond position) of an arbitrary molecule. We show how the proposed representation can be used with both graph and sequence models.

2 Bond String and Position (BSP) Representation

To construct the BSP representation we generate 3D conformations for each molecule using RDKit's structure optimization methods. This process estimates the 3D coordinates of each atom in a molecule. Next, we represent the molecule as a sequence of bond tokens ($\boldsymbol{b^t}$) and bond positions ($\boldsymbol{b^p}$). The bond token and bond position for bond $\boldsymbol{b_{ij}}$ connecting i^{th} and j^{th} atom is constructed as: $\boldsymbol{b^t_{ij}} = [\boldsymbol{a_i}||\boldsymbol{e_{ij}}||\boldsymbol{a_j}]$ and $\boldsymbol{b^p_{ij}} = [\boldsymbol{a^p_i}||\boldsymbol{a^p_j}]$ respectively. Here $\boldsymbol{a_i}$ and $\boldsymbol{a_j}$ are the tokens of the two atoms in the bond, $\boldsymbol{e_{ij}}$ is the edge token representing bond type and orientation, $\boldsymbol{a^p_i}$ and $\boldsymbol{a^p_j}$ are the 3D coordinates of the atoms, and $||$ is the concatenation operation. The atom token for the i^{th} atom is constructed as $\boldsymbol{a_i} = [a^i_{an}||a^i_{ct}||a^i_{eh}||a^i_{fc}]$, where a_{an} is the atom symbol, a_{ct} is its chiral tag, a_{eh} is the number of hydrogen atoms bonded to the atom (hydrogen atoms are otherwise excluded from the representation), and a_{fc} is the atom's formal charge. The edge token is constructed as $\boldsymbol{e_{ij}} = [e^{ij}_{type}||e^{ij}_{or}]$, where e_{type} is the bond type (e.g. single, double) and e_{or} indicates bond orientation. A detailed example of the BSP representation is provided in Appendix A.1. We can also transform the bond position into a translation and rotation invariant representation by converting the coordinates into both bond lengths and angles between bonds (see [11] for a detailed explanation of how this representation can be constructed).

This representation does not require any special tokens to specify branches and rings because such information is inherently present in the coordinates of each bond. That is, we are directly using the molecule's 3D structure as our model input, instead of requiring the model to learn this structure from a SMILES string. In addition, capturing a molecule's 3D geometry provides a richer representation than that provided by a molecular graph alone.

3 Model Architecture

The model's overall objective is to represent a molecule as a fixed size latent vector that can be used to reconstruct the molecule as a SMILES string. Molecular properties should also vary continuously along the lower dimensional space the latent vectors presumably lie on (i.e. the latent vectors should be useful for property prediction tasks).

The model is an Encoder-Decoder architecture where the encoder is a Transformer-based module inspired by DimeNet and the decoder is a conditional Transformer decoder module. The encoder output is embedded using a VAE, and this embedding is then passed to the decoder to reconstruct the SMILES strings. We describe these below, while the detailed architecture can be found in Appendix A.2.

3.1 Encoder

Our BSP representation is flexible enough to be used with different encoder architectures. Since the BSP representation can be interpreted as a sequence of bonds, we can use a standard Transformer encoder that operates over sequence data. Note that in this case we use the bond position as a positional embedding, and do not include, for instance, an additional sinusoidal embedding. We call this model the Variational Transformer (VT). Alternatively, we can interpret the BSP representation as a graph in which nodes are bond tokens and edges are determined either directly from bond connectivity or based on a distance threshold. This allows us to use GNNs like DimeNet [11] that

have established strong performance on property prediction tasks. DimeNet is a pure regression model, and is therefore not directly comparable to the generative models proposed here. We make a minimal number of modifications to its architecture to incorporate it into a VAE, and call this variant DimeNet-VAE. This serves as our performance baseline.

Training the DimeNet encoder as-is with an additional VAE objective was unstable, so we modified the model architecture by replacing the sum and mean based message aggregation with attention-based aggregation over the graph. We refer to this model as the Directional Variational Transformer (DVT). The DVT's encoder contains two main components: a graph attention head and a readout head. The graph attention head performs attention-based message passing as described in equation 1. Here, b_{ji}^{l} is the bond embedding of the l^{th} layer, f_{update} and f_{int} are multilayer perceptrons (MLPs), $\mathbf{W_q^g}$, $\mathbf{W_k^g}$ and $\mathbf{W_v^g}$ are weight matrices, N_j is the neighbourhood of atom j, and $e_{RBF}^{(ji)}$ and $a_{SBF}^{(kj,ji)}$ are the distances and angles transformed using radial and spherical basis functions, respectively, as described in [11].

$$
\begin{aligned}
b_{ji}^{(l+1)} &= f_{update}\left(b_{ji}^{(l)}, \sum_{k\in N_j, k\neq i} \alpha_{kji} \bullet f_{int}(v_{kj}^{(l)}, e_{RBF}^{(ji)}, a_{SBF}^{(kj,ji)})\right) \\
&\text{where,} \\
\alpha_{kji} &= \frac{\exp\left(q_{ji}^{(l)} \odot k_{kj}^{(l)}\right)}{\sum_{l\in N_j} \exp\left(q_{jl}^{(l)} \odot k_{kj}^{(l)}\right)} \\
q_{ji}^{(l)} &= \mathbf{W_q^g} b_{ji}^{(l)},\ k_{kj}^{(l)} = \mathbf{W_k^g} b_{kj}^{(l)},\ v_{kj}^{(l)} = \mathbf{W_v^g} b_{kj}^{(l)}
\end{aligned}
\tag{1}
$$

The readout head aggregates all the node embeddings to generate a fixed size latent vector using an attention mechanism as described in equation 2. Here, z^0 is a learnable vector.

$$
\begin{aligned}
z^{(l+1)} &= f_{update}\left(z^{(l)}, \sum \alpha_{kji} \bullet v_{ji}^{(l)}\right) \\
q_{ji}^{(l)} &= \mathbf{W_q^r} z^{(l)},\ k_{ji}^{(l)} = \mathbf{W_k^r} b_{ji}^{(l)},\ v_{ji}^{(l)} = \mathbf{W_v^r} b_{ji}^{(l)}
\end{aligned}
\tag{2}
$$

3.2 Decoder

The decoder is a Transformer style decoder. This would typically consist of two multi-head attention layers, one for self attention and another for cross attention with the encoder output sequence; however, as the output of our encoder is a single latent vector, we can merge the two attention layers as described in [17].

3.3 Loss Function

The loss function of the proposed model has three parts, as shown in equation 3. $\mathcal{L}_r$ is the cross-entropy reconstruction loss between the ground truth and predicted SMILES tokens, $\mathcal{L}_{kl}$ is the KL-divergence loss between a standard normal and the latent distribution, and finally $\mathcal{L}_{mae}$ is the mean absolute loss between the true and predicted properties. β is a hyperparameter that weights the KL-divergence loss. To avoid vanishing KL-divergence loss we used cyclic annealing schedule for β [18].

$$
\mathcal{L} = \mathcal{L}_r + \mathcal{L}_{mae} + \beta\mathcal{L}_{kl} \tag{3}
$$

4 Experiments

We evaluate the three model variants described above (VT, DVT, Dimenet-VAE) on two tasks: molecular generation and optimization.

Data: Most existing methods use datasets of up to 250k molecules to train VAE models [16, 15, 14]. We created a dataset of 6 million molecules gathered from 3 databases: Chembl [19], CCCBDB [20] and USPTO [21]. We also collect 13 different molecular properties, as detailed in Appendix A.3. We split the dataset 9:1 into training and testing splits.

Model Size: Our expanded molecular dataset warrants larger model sizes, so all three models have six-layer encoders and decoders and we use a latent space of 512 dimensions.

4.1 Molecular Reconstruction and Validity

To test our model's ability to reconstruct molecules, we compute the number of examples in the test set that were exactly reconstructed. To evaluate the learned latent space we create novel molecules by sampling 10,000 points of 512 dimension from the normal distribution and using the trained decoder of all three models to map them into SMILES space. In summary, we test the model on the 5 following metrics:
Reconstruction: Percentage of test molecules reconstructed accurately.
Validity: Percentage of randomly sampled latent points mapped to valid molecules.
Novelty: Percentage of randomly sampled molecules not present in training data.
Uniqueness: Percentage of randomly sampled molecules that are unique.
AMAE: Average Mean Absolute Error (AMAE) of property predictions over test set.

From Table 1 we can see that our proposed models perform comparatively better than the baseline DimeNet-VAE model on all five metrics. Comparing the reconstruction accuracy of DimeNet-VAE with VT and DVT we can say that only using atom embeddings [11] to create bond embedding is not enough to accurately reconstruct the molecule. A detailed comparison on individual property prediction can be found in Appendix A.4.

Table 1: Molecule Reconstruction and Validity Test

Model	Reconstruction	Validity	Novelty	Uniqueness	AMAE
DimeNet-VAE	63.22%	94.34%	92.85%	99.97%	0.8476
VT	86.39%	95.08%	94.16%	**100%**	**0.6940**
DVT	**87.29%**	**95.57**%	**94.43%**	**100%**	0.7071

Qualitatively, by interpolating between two known molecules in the latent space we observe that the predicted properties of the intermediate points vary smoothly. Further details can be found in Appendix A.5.

4.2 Discrete Bayesian Optimization

The continuous latent spaces of VAE models have been used to generate novel molecules with specific properties [15, 16]. However, the property values for these new molecules have not been verified experimentally. To address this, we designed a test that performs Bayesian optimization over a set of molecules with known ground truth properties. We assess the utility of the learned latent space by examining how quickly it can locate the molecules with the desired (lowest or highest) property values from the set.

We have a set of molecule with known property value. We start by randomly sampling one molecule to use as initial training set for Gaussian Process (GP) regression. For each iteration, we evaluate the GP on the molecules that are not part of the train set. Next, we select the molecule with the lowest predicted score, and add it to the training set of GP.

First, sort the set of molecules in the increasing order of their ground truth property value. We define rank of the molecule as its index in the sorted list. To measure the performance of the model we compute the absolute difference between rank of the molecule and the iteration number at which the molecule was selected. For example if the molecule with rank k is selected at iteration n then it will yield an error of $|\mathrm{k} - \mathrm{n}|$. Let us call this metric M1. We also measure iteration number at which the molecule with the lowest property was selected, let us call this metric M2. We repeat the experiment 100 times with different initial samples and report average metrics. Table A.6 shows the performance of all three models on ESOL and FreeSolv datasets provided by DeepChem [22]. As we can see for both datasets, the VT and DVT models performed better than the DimeNet-VAE model in most cases. Additional experimental results can be found in Appendix 5.

Table 2: Discrete Bayesian Optimization

Dataset	Samples	Metric	Dimenet-VAE	VT	DVT
ESOL	1121	M1	104.54 ± 0.34	$\mathbf{100.0 \pm 0.51}$	101.23 ± 0.60
		M2	$\mathbf{10.95 \pm 9.07}$	18.39 ± 7.89	23.01 ± 11.32
FreeSolv	628	M1	61.70 ± 0.72	57.16 ± 0.87	$\mathbf{52.47 \pm 0.69}$
		M2	24.27 ± 7.97	11.04 ± 3.39	$\mathbf{10.66 \pm 4.99}$

5 Conclusion

We proposed a Transformer-based variational autoencoder model for generating a continuous latent space of molecules, in conjunction with a novel molecular representation, Bond String and Position, that captures 3D structure and can be made invariant to rotations and translations. We showcased the effectiveness of the learned latent space by applying it in a Bayesian optimization setting, where we efficiently located molecules with optimal properties.

References

[1] I. Muegge and P. Mukherjee, "An overview of molecular fingerprint similarity search in virtual screening," *Expert opinion on drug discovery*, vol. 11, no. 2, pp. 137–148, 2016.

[2] A. Cereto-Massagué, M. J. Ojeda, C. Valls, M. Mulero, S. Garcia-Vallvé, and G. Pujadas, "Molecular fingerprint similarity search in virtual screening," *Methods*, vol. 71, pp. 58–63, Jan. 2015.

[3] G. Graziano, "Fingerprints of molecular reactivity," *Nature Reviews Chemistry*, vol. 4, no. 5, pp. 227–227, 2020.

[4] H. L. Morgan, "The generation of a unique machine description for chemical structures-a technique developed at chemical abstracts service.," *Journal of Chemical Documentation*, vol. 5, pp. 107–113, May 1965.

[5] Z. Wu, B. Ramsundar, E. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing, and V. Pande, "Moleculenet: a benchmark for molecular machine learning," *Chem. Sci.*, vol. 9, pp. 513–530, 2018.

[6] C. W. Coley, R. Barzilay, W. H. Green, T. S. Jaakkola, and K. F. Jensen, "Convolutional embedding of attributed molecular graphs for physical property prediction," *Journal of Chemical Information and Modeling*, vol. 57, pp. 1757–1772, July 2017.

[7] D. Weininger, A. Weininger, and J. L. Weininger, "Smiles. 2. algorithm for generation of unique smiles notation," *Journal of chemical information and computer sciences*, vol. 29, no. 2, pp. 97–101, 1989.

[8] M. Krenn, F. Häse, A. Nigam, P. Friederich, and A. Aspuru-Guzik, "Selfies: a robust representation of semantically constrained graphs with an example application in chemistry," *arXiv preprint arXiv:1905.13741*, 2019.

[9] P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. A. Hunter, C. Bekas, and A. A. Lee, "Molecular transformer: A model for uncertainty-calibrated chemical reaction prediction," *ACS Central Science*, vol. 5, pp. 1572–1583, Aug. 2019.

[10] S. Chithrananda, G. Grand, and B. Ramsundar, "Chemberta: Large-scale self-supervised pretraining for molecular property prediction," 2020.

[11] J. Gasteiger, J. Groß, and S. Günnemann, "Directional message passing for molecular graphs," in *International Conference on Learning Representations (ICLR)*, 2020.

[12] O. T. Unke and M. Meuwly, "PhysNet: A neural network for predicting energies, forces, dipole moments, and partial charges," *Journal of Chemical Theory and Computation*, vol. 15, pp. 3678–3693, May 2019.

[13] L. Maziarka, T. Danel, S. Mucha, K. Rataj, J. Tabor, and S. Jastrzebski, "Molecule attention transformer," *arXiv*, 2020.

[14] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, "Automatic chemical design using a data-driven continuous representation of moleculesmaterial design," *ACS Central Science*, vol. 4, pp. 268–276, Jan. 2018.

[15] W. Jin, R. Barzilay, and T. Jaakkola, "Junction tree variational autoencoder for molecular graph generation," in *International conference on machine learning*, pp. 2323–2332, PMLR, 2018.

[16] Q. Liu, M. Allamanis, M. Brockschmidt, and A. Gaunt, "Constrained graph variational autoencoders for molecule design," in *Advances in Neural Information Processing Systems* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), vol. 31, Curran Associates, Inc., 2018.

[17] L. Fang, T. Zeng, C. Liu, L. Bo, W. Dong, and C. Chen, "Transformer-based conditional variational autoencoder for controllable story generation," *arXiv*, 2021.

[18] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin, "Cyclical annealing schedule: A simple approach to mitigating kl vanishing," *arXiv preprint arXiv:1903.10145*, 2019.

[19] D. Mendez, A. Gaulton, A. P. Bento, J. Chambers, M. De Veij, E. Félix, M. Magariños, J. Mosquera, P. Mutowo, M. Nowotka, M. Gordillo-Marañón, F. Hunter, L. Junco, G. Mugumbate, M. Rodriguez-Lopez, F. Atkinson, N. Bosc, C. Radoux, A. Segura-Cabrera, A. Hersey, and A. Leach, "ChEMBL: towards direct deposition of bioassay data," *Nucleic Acids Research*, vol. 47, pp. D930–D940, 11 2018.

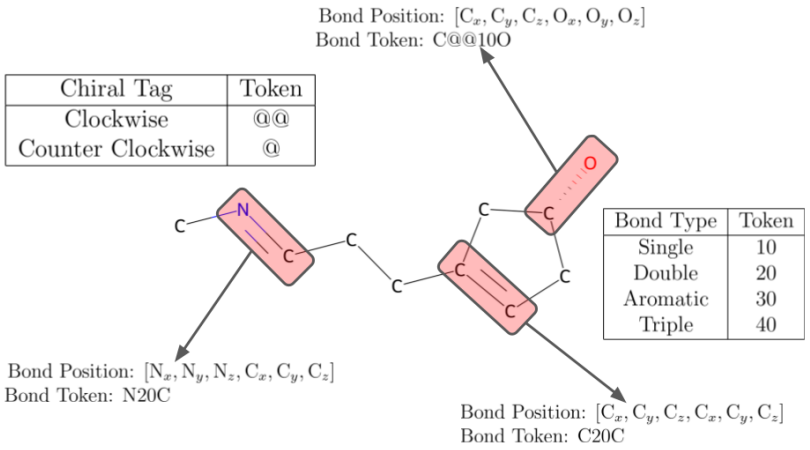
[20] R. D. Johnson *et al.*, "Nist computational chemistry comparison and benchmark database," *http://srdata. nist. gov/cccbdb*, 2006.

[21] D. Lowe, "Chemical reactions from us patents (1976–sep2016). figshare https://figshare. com/articles," *Chemical_reactions_from_US_patents_1976-Sep2016_/5104873*, 2017.

[22] B. Ramsundar, P. Eastman, P. Walters, V. Pande, K. Leswing, and Z. Wu, *Deep Learning for the Life Sciences*. O'Reilly Media, 2019. `https://www.amazon.com/Deep-Learning-Life-Sciences-Microscopy/dp/1492039837`.

A Appendix

A.1 Bond String and Position Representations examples



Chiral Tag	Token
Clockwise	@@
Counter Clockwise	@

Bond Type	Token
Single	10
Double	20
Aromatic	30
Triple	40

Figure 1: Depiction of how a bond token is generated.

A.2 Model Architecture

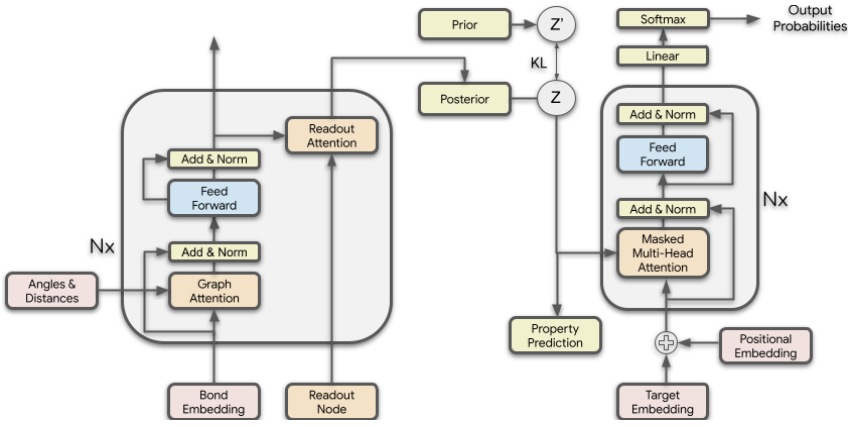


Figure 2: Illustration of the DVT model. The main component is a graph-based encoder module which operates on our BSP inputs and returns a fixed size latent vector. The decoder module is the standard decoder from the Transformer model: it takes the latent vector and generates a SMILES representation of the molecule.

A.3 Properties of molecules

Table 3 refers to list of property used for model training.

Table 3: List of properties

Property	Description
GAP	HOMO/LUMO gap
MW	Molecular weight
ALOGP	Calculated ALogP
HBA	Number of hydrogen bond acceptors
HBD	Number of hydrogen bond donors
PSA	Polar surface area
ROTB	Number rotatable bonds
RO5V	Number of violations of Lipinski's rule-of-five
CXA	The most acidic pKa calculated
CXB	The most basic pKa calculated
CX_LOGP	The calculated octanol/water partition coefficient
CX_LOGD	The calculated octanol/water distribution coefficient
QED	Weighted quantitative estimate of drug likeness

A.4 Property wise model performance

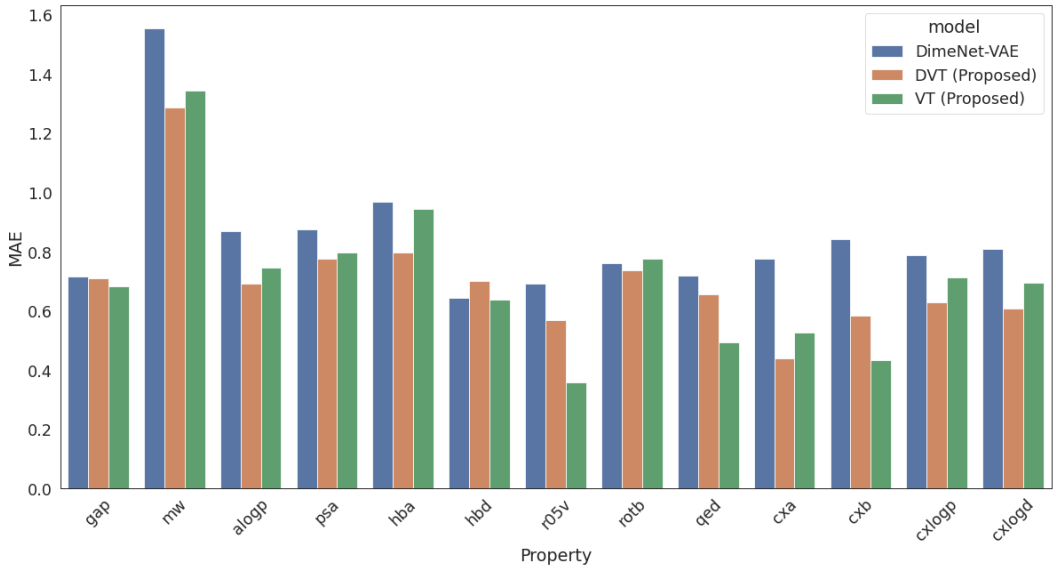


Figure 3: Property wise MAE of different models.

A.5 Latent space interpolation

We select two molecules from the training dataset with the highest and lowest drug likeness (QED) score. Next, we linearly interpolate between their embeddings to generate 10 intermediate samples. We also predict the QED score for each intermediate sample. Figure 4 shows the generated intermediate molecules using all three models along with their predicted QED score. As we move from left (low QED molecules) to right (high QED molecules), the predicted scores of the intermediate molecules increase smoothly.

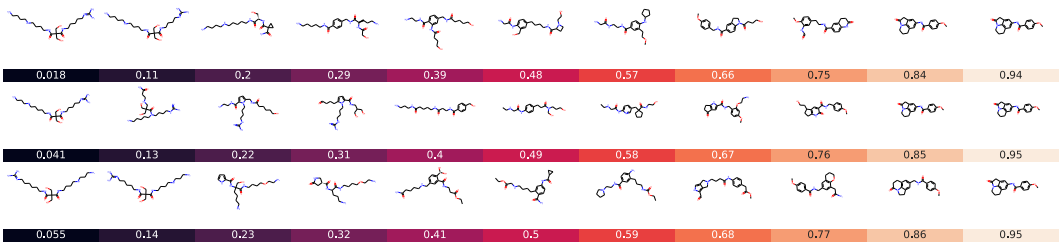
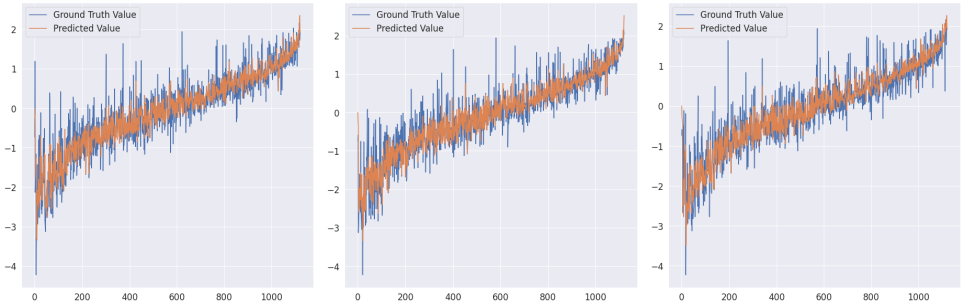


Figure 4: Interpolation between two molecules using DimeNet-VAE (first row), VT (second row) and DVT (third row). The first molecule in each row is the starting point and the last molecule is the ending point, values represent QED score.

A.6 Discrete Bayesian Optimization

Figure 5 shows the ground truth and predicted property value of the molecule selected at each iteration over ESOL dataset using DimeNet-VAE, VT and DVT. As we can see the molecules with the lower scores were selected at the beginning only.



a b c

Figure 5: Discrete Bayesian optimization results over ESOL dataset using (a) DimeNet-VAE, (b) VT and (c) DVT. The x-axis indicates the iteration number and the y-axis indicates the property value.