ChemoVerse: Manifold traversal of latent spaces for novel molecule discovery

Harshdeep Singh[1]*, **Nicholas McCarthy**[2], **Qurrat Ul Ain**[3], **Jeremiah Hayes**[2]
[1]Swiss Federal Institute of Technology, Lausanne, [2]Accenture Labs, Dublin [3]AI Innovation Lab, DSAI, Novartis
harshdeep.harshdeep@epfl.ch, {nicholas.mccarthy,jer.hayes}@accenture.com, qurrat.ul_ain@novartis.com

Abstract

In order to design a more potent and effective chemical entity, it is essential to identify molecular structures with the desired chemical properties. Recent advances in generative models using neural networks and machine learning are being widely used by many emerging startups and researchers in this domain to design virtual libraries of drug-like compounds. Although these models can help a scientist to produce novel molecular structures rapidly, the challenge still exists in the intelligent exploration of the latent spaces of generative models, thereby reducing the randomness in the generative procedure. In this work we present a manifold traversal with heuristic search to explore the latent chemical space. Different heuristics and scores such as the Tanimoto coefficient, synthetic accessibility, binding activity, and QED drug-likeness can be incorporated to increase the validity and proximity for desired molecular properties of the generated molecules. For evaluating the manifold traversal exploration, we produce the latent chemical space using various generative models such as grammar variational autoencoders (with and without attention) as they deal with the randomized generation and validity of compounds. With this novel traversal method, we are able to find more unseen compounds and more specific regions to mine in the latent space. Finally, these components are brought together in a simple platform allowing users to perform search, visualization and selection of novel generated compounds.

1 Introduction

Designing a new chemical entity is a time consuming, expensive, and error-prone task. Pharmaceutical companies invest billions of dollars into screening vast libraries of chemical compounds for hit and lead identification [Fleming, 2018]. The past few years has seen the rise of deep generative models that can operate over large spaces of molecular structures and embed the chemical properties of such into a vector space. By decoding from this 'latent' space of chemical structure we can generate new, previously unidentified chemical compounds.

In the domain of computational chemistry, Simplified molecular-input line-entry system or SMILES are a common string textual method for encoding and representation of molecular structures [Anderson *et al.*, 1987]. This facilitates the use of models more commonly used in natural language processing. Therefore, SMILES strings have been used as raw input strings to generative models, which are given the task of encoding and decoding the SMILES string directly [Anderson *et al.*, 1987]. Advances were made by employing variational autoencoders (VAE), a neural network comprised of an encoder that transforms a compound's representation into a compressed latent space, and a decoder that generates compounds from the latent space [Kingma and Welling, 2013]. Although, directed search of the resulting latent space is difficult. To counter this, conditional variational autoencoders (CVAE)[Kang and Cho, 2018] were used in order to facilitate the generation of new molecules with specified molecular properties. This is achieved by incorporating the molecular properties of a compound into the encoder layer and helping in the generation of more drug-like molecules [Lim *et al.*, 2018]. Generative adversarial networks (GANs) have also been applied in the same manner [Maziarka *et al.*, 2020], and have been recently combined with reinforcement learning and graph representation of molecules to optimize the generation of molecules with specified molecular properties [De Cao and Kipf, 2018].

Discovery in the latent space generated by these models is often performed using random sampling and linear interpolation, primarily due to the ease of the implementation of these methods. However, this is not suitable for most generative models as their latent spaces are generally high dimensional and sparse. While doing traversal, we will traverse regions where the data is not very well represented. In other words, this could lead to a 'dead zone' as the space of molecular samples in the training dataset are present only on a subset of the latent space [White, 2016]. Hence, decoding a point from the latent space will end up returning noisy or invalid results. It can also be challenging to incorporate contextual domain information during search, and as a result discovery of compounds with specific properties is often very inconsistent.

In this work we implemented various flavours of auto-

[*] This work was performed during an internship programme at Accenture Labs, Dublin, Ireland.

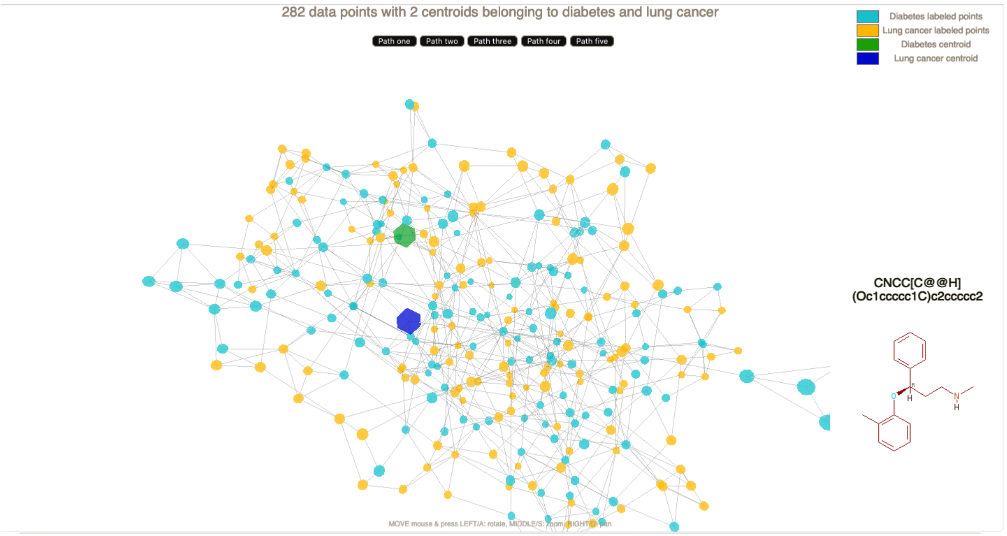

282 data points with 2 centroids belonging to diabetes and lung cancer
Path one
Path two
Path three
Path four
Path five
Diabetes labeled points
Lung cancer labeled points
Diabetes centroid
Lung cancer centroid
CNCC[C@@H]
(Oc1ccccc1C)c2ccccc2
H
N
H
MOVE mouse & press LEFT/A: rotate, MIDDLE/S: zoom, RIGHT/D: pan

Figure 1: Visualization of the latent space as a K-dimensional tree, while traversing the latent manifold of compounds used in the treatment of diabetes and lung cancer.

encoders as the generative model for producing sets of latent spaces, and in particular show that our novel implementation of Grammar VAE [Kusner *et al.*, 2017] with an additional attention mechanism [Vaswani *et al.*, 2017] is highly performative, with a low rate of invalid molecules generated. We also introduce a novel manifold interpolation method employing the Riemannian metric [Arvanitidis *et al.*, 2017] in conjunction with a set of molecular property heuristics to perform directed search and interpolation of these latent spaces in order to design novel molecules with desired properties. This combination of generation and exploration of latent space has enabled us to not only design molecules which have not been seen before but also to explore new regions of latent chemical space where more potent chemical compounds may exist.

2 System Architecture

In this section we describe the various components of our system architecture: generation of latent spaces and our algorithm for manifold traversal.

2.1 Data

We used a dataset of 250,000 molecules drawn from the ZINC dataset [Irwin *et al.*, 2012], and an additional 100,000 drawn from the ChEMBL dataset [Gaulton *et al.*, 2017]. These two datasets are comprised of commercially available drug molecules and have been used in related work using models like variational autoencoders (VAEs) [Gómez-Bombarelli *et al.*, 2018]. Molecules are represented in canonical SMILES string format, and are further processed into 1) a one-hot character encoding and 2) a set of context-free grammar (CFG) rules. Grammar rules are obtained from the OpenSMILES specification [James and Dalke, 2016], which denotes how the SMILES representation was formed based on the rules. This context free grammar (CFG) consists of 76 production rules, to which an additional seven are added, and a further nine modified in order to represent the more complex ChEMBL dataset.

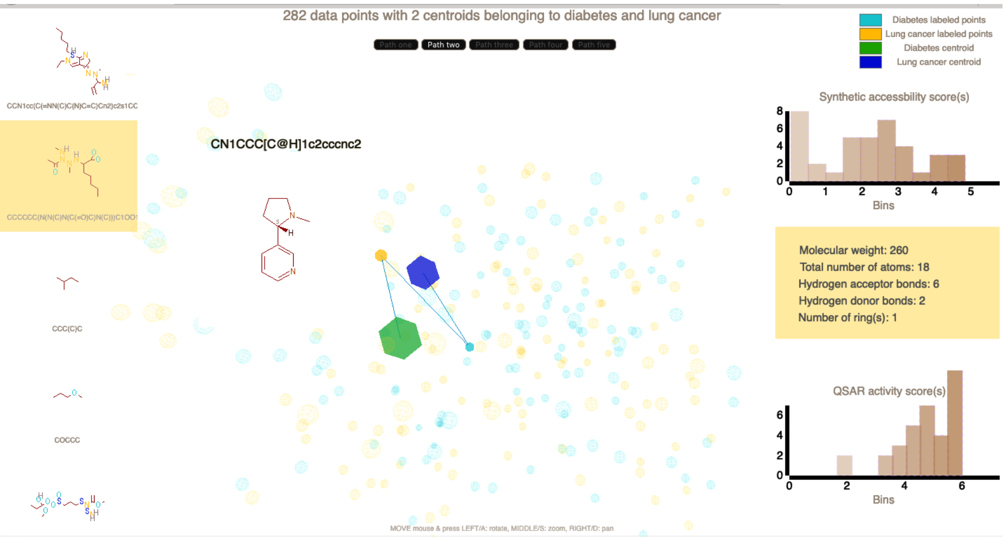
2.2 Latent Space Generation

Three models are implemented in our system: a VAE [Kingma and Welling, 2013], a Grammar VAE [Kusner *et al.*, 2017], and a Grammar VAE with self-attention [Vaswani *et al.*, 2017]. As our search algorithm is model agnostic, latent spaces can be substituted with minimal effort. However, results will vary depending on the underlying data, model architecture, and training parameters used to generate each latent space.

The ChEMBL dataset is less standardized and contains more complex molecules, therefore we perform transfer learning by initially training each model on the ZINC dataset for 50 epochs, then switching to the ChEMBL dataset for 50 epochs. Training, validation and test sets of 85%, 10%, and 5% respectively was used, with the test set comprised entirely of ChEMBL molecules. We use the Adam optimizer with a learning rate scheduler that is instantiated after 15 epochs with a factor of 0.1 (initialized at 0.001).

The encoder is comprised of three 1D convolutional layers with filters of size 9, 10 and 11 respectively, while the decoder is comprised of 3 gated recurrent units (GRU) of 501 units [Kusner *et al.*, 2017]. The structural validity of the generated compounds are checked using the open-source RDKit library [Landrum,]. Examples of test set compounds that have been encoded and decoded are shown in table 1.

As noted in the original Grammar VAE work [Kusner *et al.*, 2017] the vanilla VAE architectures encoding SMILES strings directly generally produce a very low valid decode rate on larger molecular datasets - just 17% using a conditional VAE under their bayesian optimization search methodology. By instead using a Grammar VAE to generate production rules of a grammar instead of SMILES strings directly a much higher rate of valid compounds was attained. However,


282 data points with 2 centroids belonging to diabetes and lung cancer
Path one
Path two
Path three
Path four
Path five
🟦 Diabetes labeled points
🟧 Lung cancer labeled points
🟩 Diabetes centroid
🟦 Lung cancer centroid
CCN1cc(C(=NN(C)C(N)C=C)Cn2)c2s1CC
CN1CCC[C@H]1c2cccnc2
CCCCCC(N(N(C)N(C(=O)C)N(C)))C1OO1
CCC(C)C
COCCC
Synthetic accessbility score(s)
8
6
4
2
0
0
1
2
3
4
5
Bins
QSAR activity score(s)
6
4
2
0
0
2
4
6
Bins
MOVE mouse & press LEFT/A: rotate, MIDDLE/S: zoom, RIGHT/D: pan

Molecular weight: 260
Total number of atoms: 18
Hydrogen acceptor bonds: 6
Hydrogen donor bonds: 2
Number of ring(s): 1

Figure 2: Visualization of the path in the latent space along which our algorithm is interpolating. On the left are generated compounds not present in the training data. On the right are histograms of different molecular properties, including synthetic accessibility, activity scores.

Algorithm 1 Interpolation with manifold traversal

```
**Input**: latent space $L$, source and destination points $s$ and $d$, points of interest $R$, path length $m$
**Parameter**: $k$ heuristics and associated edge weights $W^k$
**Output**: a path of $m$ equidistant points from $s$ to $d$
**for** each reference point $i$ in $R$ **do**
    Find $n$ nearest neighbors of $i \rightarrow N_i$
    **for** each $(i, j)$, where j $\in N_i$ **do**
        1. Calculate Jacobian matrix $J_{ij}$ of decoder output $(dec)$ w.r.t to latent space $(L)$ $\frac{\partial dec(L_i j)}{\partial L_{ij}}$
        2. Obtain $k$ heuristic distances $H_{ij}^k$ between the 2 points
        3. Determine path edge weight based on Jacobian and heuristic: $P_{ij} \leftarrow \sum_{ij}^{N}$ J$_{ij}$ + $\sum_k$H$_{ij}^k \cdot W^k$ and store in a k-d tree.
**end for**
Apply a path finding algorithm such as Yen's algorithm (with A*) to determine a path from $s$ to $d$ on the k-d tree.
Segment the path using $m$ equidistant points.
**return** $P \leftarrow$ (s, p$^1$, .., $p^{m-2}$, $d$)
```

as OpenSMILES is a context-free instead of regular grammar it is still unable to model certain subtle characteristics of the SMILES grammar such as paired ring bonds; for example the SMILES string 'c1ccccc1C2CCCC2' would be decoded as 'c1ccccc1C2CCCC', incorrectly dropping the final paired digit. By incorporating a self-attention layer in the Grammar VAE architecture this effect was mitigated, and increased the validity of decoded test set molecules from 61% to 70%.

2.3 Manifold Traversal of Latent Space

Once a latent space has been generated we can apply our manifold traversal algorithm to generate interpolative paths in the latent space in order to decode molecules with desired properties. A common approach here is to use linear or spherical interpolation [White, 2016], however both approaches assume that the latent space is Euclidean and flattened out, and generally produce noisier results [Arvanitidis *et al.*, 2017]. In our algorithm, source and destination points are selected in the latent space; these can either be latent space encoding of single molecules, or cluster centroids of molecules labeled with a desired property. Points of interest are the set of known molecules with desired properties, for example all molecules used in treatment of a specific condition. The goal is to define a path from source to destination points in the latent space through regions of interest, factoring in any additional user-specified heuristics such as synthetic accessibility, binding activity, or drug-likeness that will augment generated molecules.

Interpolation is performed by first calculating the Jacobian distances for all points of interest. This helps us to understand how much each latent space point differs from another based on the representation learnt by the model, and understand the stretching and rotational transformations of the local neighborhood of each point with respect to other points.

A k-dimensional tree is built using the resulting Jacobian distances as edge weights between compounds in the points of interest. Therefore, placing compounds with greater structural similarity in closer proximity on the tree. A k-d tree is chosen since it divides the domain of search into half at each level. Hence search for a node in the tree can be done in logarithmic time and makes the data structure run time efficient. Edges can also be weighted by user-specified domain heuristics, adding a weighted cost to augment the paths produced to generate molecules more relevant for a specific target. These heuristics are listed below:

- Fingerprint Similarity: A fingerprint is a series of binary digits (bits) that represent the presence or absence of particular substructures in the molecule. The similarity can be tested using cosine or Tanimoto distance metrics. The Tanimoto similarity takes into account the

Table 1: Input and corresponding decoded SMILES strings of test set compounds generated by the model. Tanimoto similarity is used to measure the the structural similarity. An observation that can be made here is that small changes in the generated structure w.r.t to actual structure might lead to disproportionate changes in the similarity.

Actual Compound	Generated compound	Similarity
OC(=O)CSc1oc2ccccc2n1	OC(=O)CCS1cc2ccccs2n1	0.141
CC(=O)C(=Cc1ccccc1)c2ccccc2	O.CCCN(Cc1ccccc1c2)cccnn2	0.171
Cc1occc1C(=O)NCc2onc(n2)c3ccccc3	Nc1nnc1NC(=O)NCSc2nn(s2)c3ccccc3	0.221
Cc1ccc(SCC(=O)NNC(=O)c2cncc(Br)c2)cc1	Clc1ccc(SCC(=O)NC(=O)c2cncc(Br)c2cc1)CF	0.373
CCc1ccc(cc1)N(CC(=O)N2CCOCC2)S(=O)(=O)C	CSc1ccc(cc1N(C(=O)Oc2CCCCC2S(=O)(=O)))ON	0.235

Table 2: Samples of the compounds generated while interpolating on the latent space path as shown in Figure 2. These compounds are not present in the train/test datasets, and we assign a potential target label based on labels of their nearest neighbours in the test set. SAS score indicates if a molecule is difficult to synthesize. According to the Lipinski's rule of five [Benet *et al.*, 2016], the molecular mass for an active drug should be less than 500 daltons. Activity range demonstrates the binding activity of a compound: the greater the range more active it is.

Generated compound	Activity range	SAS Score	Molecular weight	Potential Label
CS1(C2)CC(C(C3C(C))C)[C@]1n1C2n1C3C4CC4C	Less than 5	6.186	299.89	DIABETES
CCC1(C)C(C)CCCCS1C(C1(O([C@+2](C1))))S	Between 5 and 7	6.251	274.49	DIABETES
CSCCCC2(C=C(O))S(C=O)C2C1[C@@](C1(C))CC	Between 5 and 7	5.990	301.49	DIABETES
CC1(N(S))NN1CCCNc1ncnc2CCn2c1CC	Greater than 7	5.233	299.89	LUNG CANCER
CS1(C2)CC(C(C3C(C))C)[C@]1n1C2n1C3C4CC4Cl	Less than 5	6.186	309.44	LUNG CANCER
CCCC1CCC(S)N(C1)C(O)NCC	Less than 5	4.355	232.93	DIABETES

structural properties whereas cosine does not.

- Synthetic Accessibility (SA Score): A molecule synthetic accessibility is a score which is between 1 (easy to produce) and 10 (very difficult to produce). This is calculated based on fragment contributions and molecule complexity. The absolute difference between two molecules is taken into account.
- Drug-likeliness: This is a score which takes into account if the molecule is 'drug-like'. This is evaluated using several parameters such as molecular weight, solubility in water or lipophilic efficiency. The absolute difference between two molecules is taken into account.
- Binding activity: This demonstrates the potency to a target for a potential drug compound; less than 5 is considered inactive, 5-7 of intermediate activity, greater than 7 active.

Yen's algorithm [Yen, 1971] in combination with the A* algorithm is then applied on the k-d tree to find the shortest path from source to destination given the user constraints. Once the shortest path is found, we interpolate along this path equidistantly and decode the points on the latent space using the generator to generate compounds. Multiple paths can be found by taking into account the shortest path and either perturbing it or by changing the number of interpolation points between the source and the destination, and intuitively this increases the overall number of novel generated compounds.

Manifold traversal is inherently more useful than linear or spherical interpolation as it gives users greater flexibility in path exploration under various conditions, and empirically demonstrates a much higher rate of valid decoded molecules. For example, when considering the *diabetes* and *lung cancer* centroids; linear interpolation with 100 equidistant points decoded along the path of centroids generated just 3 compounds with valid structures. On the contrary, applying manifold traversal with fingerprint similarity and Yen's algorithm as heuristic and perturbing the source and destination points produced 4 different paths. These four paths generated a total of 156 valid, novel compounds along the interpolated manifold between the latent regions of *diabetes* and *lung cancer* labeled molecules. Samples of these generated compounds can be seen in Table 2. Specific regions to mine within the latent space can be found by plotting different paths, bound them and finding the overlap region where compounds with the right structure and specific characteristics can be found.

3 Conclusions and Future Work

In this work we presented a model-agnostic platform for performing manifold traversal on latent spaces with user specified domain heuristics. This interpolation method allows us to add more context and direction to search and discovery of molecules in the latent space. Methods for exploration of latent spaces generated from datasets of millions of molecules provide an extremely valuable tool for virtual drug screening, and an ability to facilitate rapid drug discovery.

Some avenues for future work in this domain include: implementation of alternative models to produce latent-spaces of various characteristics; more sophisticated methods for curve fitting in high dimensional spaces such as Bézier curves and Gaussian regression; alternative search methods such as using evolutionary and genetic algorithms on the latent space. Future work would also focus on implementing latent space evaluation metrics using this interpolation method to understand the underlying aspects of these spaces.

References

[Anderson *et al.*, 1987] E. Anderson, G.D. Veith, D. Weininger, and Minn.) Environmental Research Laboratory (Duluth. *SMILES, a Line Notation and Computerized Interpreter for Chemical Structures*. Environmental research brief. U.S. Environmental Protection Agency, Environmental Research Laboratory, 1987.

[Arvanitidis *et al.*, 2017] Georgios Arvanitidis, Lars Kai Hansen, and Søren Hauberg. Latent space oddity: on the curvature of deep generative models. *arXiv preprint arXiv:1710.11379*, 2017.

[Benet *et al.*, 2016] Leslie Z Benet, Chelsea M Hosey, Oleg Ursu, and Tudor I Oprea. Bddcs, the rule of 5 and drugability. *Advanced drug delivery reviews*, 101:89–98, 2016.

[De Cao and Kipf, 2018] Nicola De Cao and Thomas Kipf. Molgan: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973*, 2018.

[Fleming, 2018] Nic Fleming. How artificial intelligence is changing drug discovery. *Nature*, 557(1):S55–S57, 2018.

[Gaulton *et al.*, 2017] Anna Gaulton, Anne Hersey, Michał Nowotka, A Patrícia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J Bellis, Elena Cibrián-Uhalte, et al. The chembl database in 2017. *Nucleic acids research*, 45(D1):D945–D954, 2017.

[Gómez-Bombarelli *et al.*, 2018] 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. *ACS central science*, 4(2):268–276, 2018.

[Irwin *et al.*, 2012] John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. Zinc: a free tool to discover chemistry for biology. *Journal of chemical information and modeling*, 52(7):1757–1768, 2012.

[James and Dalke, 2016] Vander-meersch T James, Craig A and A. Dalke. Opensmiles specification, 2016.

[Kang and Cho, 2018] Seokho Kang and Kyunghyun Cho. Conditional molecular design with deep generative models. *Journal of chemical information and modeling*, 59(1):43–52, 2018.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Kusner *et al.*, 2017] Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1945–1954. JMLR. org, 2017.

[Landrum,] Greg Landrum. Rdkit: Open-source cheminformatics.

[Lim *et al.*, 2018] Jaechang Lim, Seongok Ryu, Jin Woo Kim, and Woo Youn Kim. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *Journal of cheminformatics*, 10(1):1–9, 2018.

[Maziarka *et al.*, 2020] Łukasz Maziarka, Agnieszka Pocha, Jan Kaczmarczyk, Krzysztof Rataj, Tomasz Danel, and Michał Warchoł. Mol-cyclegan: a generative model for molecular optimization. *Journal of Cheminformatics*, 12(1):1–18, 2020.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[White, 2016] Tom White. Sampling generative networks. *arXiv preprint arXiv:1609.04468*, 2016.

[Yen, 1971] Jin Y Yen. Finding the k shortest loopless paths in a network. *management Science*, 17(11):712–716, 1971.