SALSA: Semantically-Aware Latent Space Autoencoder

Kathryn E. Kirchoff[1], **Travis Maxfield**[2], **Alexander Tropsha**[2*], **Shawn M. Gomez**[3, 4†]

[1]Department of Computer Science, UNC Chapel Hill
[2]Eshelman School of Pharmacy, UNC Chapel Hill
[3]Department of Pharmacology, UNC Chapel Hill
[4]Joint Department of Biomedical Engineering at UNC Chapel Hill and NC State University
kat@cs.unc.edu, tmaxfield@unc.edu, alex_tropsha@unc.edu, smgomez@unc.edu

Abstract

In deep learning for drug discovery, molecular representations are often based on sequences, known as SMILES, which allow for straightforward implementation of natural language processing methodologies, one being the sequence-to-sequence autoencoder. However, we observe that training an autoencoder solely on SMILES is insufficient to learn molecular representations that are semantically meaningful, where semantics are specified by the structural (graph-to-graph) similarities between molecules. We demonstrate by example that SMILES-based autoencoders may map structurally similar molecules to distant codes, resulting in an incoherent latent space that does not necessarily respect the semantic similarities between molecules. To address this shortcoming we propose Semantically-Aware Latent Space Autoencoder (SALSA) for molecular representations: a SMILES-based transformer autoencoder modified with a contrastive task aimed at learning graph-to-graph similarities between molecules. To accomplish this, we develop a novel dataset comprised of sets of structurally similar molecules and opt for a supervised contrastive loss that is able to incorporate full sets of positive samples. We evaluate semantic awareness of SALSA representations by comparing to its ablated counterparts, and show empirically that SALSA learns representations that maintain 1) structural awareness, 2) physicochemical awareness, 3) biological awareness, and 4) semantic continuity.

Introduction

In drug discovery, learning the underlying semantics that govern molecular data presents an interesting challenge for deep learning. Effective learning of semantics is necessary to be successful in key tasks such as property prediction and *de novo* generation, and progress has been made in attempting to solve these tasks (Bilodeau et al. 2022). However, due to the ambiguous nature of molecular representations, models often fail to adequately capture the underlying semantics resulting in a disorganized latent space.

In the case of molecular data, *semantics* is often task-dependent but may amount to various emergent properties (e.g. structural, physicochemical, and biological properties) that are intrinsically linked to molecular structure, that is, the arrangement of constituent atoms and bonds (Honda et al. 2016). Molecular structure can be captured in the form of a graph, and thus the semantics that govern chemical manifolds may therefore be specified by the graph-to-graph similarities (i.e. structural similarities) between molecules. In this way, graph edit distance (GED) defines a semantically meaningful unit of change between molecular entities.

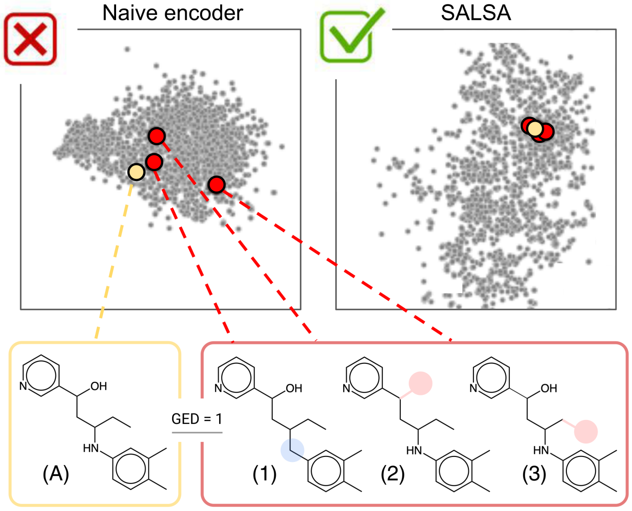

Figure 1: Given molecule (A) we consider three molecules whose graphs are structurally similar, being a single graph edit from (A). The naive autoencoder maps these similar molecules to latent codes of various proximity: (1) is mapped close to (A), while (3) is mapped far from (A). In contrast, our proposed autoencoder, SALSA, learns a *semantically-aware* space such that structurally similar molecules are collectively mapped to nearby codes.

In molecular representation learning, we can conveniently express molecular structures as linear sequences, known as simplified molecular-input line-entry system (SMILES) strings (Weininger 1988) in order to take advantage of recent progress in sequence-to-sequence modeling. Borrowing from advancements made in natural language processing (NLP), many autoencoder-based methods operating on SMILES sequences have been proposed as they provide

[*]Corresponding author
[†]Corresponding author

a promising framework to solve problems in drug discovery (Alperstein, Cherkasov, and Rolfe 2019; Gómez-Bombarelli et al. 2018; Bilodeau et al. 2022).

However, these SMILES-based autoencoders are plagued by some of the same challenges met in the field of NLP, namely, difficulties in learning latent spaces that capture underlying sentence semantics (Xu et al. 2021; Shen et al. 2020). This arises from the fact that for discrete objects such as sentences, autoencoders have the capacity to map similar data to distance latent representations. We observe an analogous problem in that SMILES-based autoencoders are not able to adequately learn the structure-based semantics that underlie chemical datasets, and as a result, these models may map semantically similar molecules to distant codes in the latent space. This phenomenon is more precisely defined as an instance in which structurally similar molecules (low GED) are mapped to distant latent representations (high Euclidean distance). We show an example of this in Figure 1. Collectively, many of these *semantically naive* events induce a disorganized latent space which limits success of these models in downstream tasks.

To remedy this shortcoming of SMILES-based autoencoders, we propose enforcing a sense of semantic awareness on to an autoencoder such that structurally similar molecules are mapped near one another in the latent space. Our proposed model, Semantically-Aware Latent Space Autoencoder (SALSA), is a modified SMILES-based transformer autoencoder that, in addition to a canonical reconstruction loss, learns a contrastive task having the objective of mapping structurally similar molecules, whose graphs are separated by a single edit distance, to similar codes in the effected latent space. In this way, we are able to learn a semantically meaningful latent space. We compare SALSA to its two ablations (a naive SMILES autoencoder and a contrastive encoder) and evaluate their latent spaces in terms of not only structural awareness, but also physicochemical and biological awareness as well as semantic continuity. We are the first, to our knowledge, to enforce structural awareness onto a SMILES-based model.

Our contributions are as follows:

- We propose a novel modeling framework, SALSA, that composes a transformer autoencoder with a contrastive task to achieve semantically-aware molecular representations.
- We develop a scheme for constructing a chemical dataset suited to contrastive learning of molecular entities, specifically aimed at learning structural similarities between molecules.
- We evaluate the quality of SALSA's latent space based on: 1) structural awareness, 2) physicochemical awareness, 3) biological awareness, and 4) semantic continuity.

Related Works

Sequence-Based Models. For our sequence-based (i.e. SMILES-based) representation, we are specifically interested in methods that allow for global representation of sequence inputs. Earlier methods aimed at embedding whole sequences utilized recurrent neural networks (RNNs), including long short-term memory networks (LSTMs), naturally aligned to this objective (Bowman et al. 2016; Shen et al. 2020). However, most state-of-the-art methods are based on the original transformer architecture (Vaswani et al. 2017) and do not provide a global representation of the input. Recently, authors have modified the transformer architecture to include a bottleneck (or pooling) layer allowing for a single, fixed-size global embedding of the input (Montero, Pappas, and Smith 2021; Jiang et al. 2020; Li et al. 2020). Examples of autoencoder-based methods include ChemVAE (Gómez-Bombarelli et al. 2018) and AllSMILES VAE (Alperstein, Cherkasov, and Rolfe 2019). Transformer-based models include ChemBERTa (Chithrananda, Grand, and Ramsundar 2020), SMILESTransformer (Honda, Shi, and Ueda 2019), and FragNet (Shrivastava and Kell 2021). Less common, however, is the composed architecture of a transformer autoencoder.

Contrastive Learning. For molecular data, both SMILES and graph representations have been explored in the context of contrastive learning. The FragNet model proposed by Shrivastava and Kell (2021) utilized the normalized temperature-scaled cross entropy (NT-Xent) (Sohn 2016) loss to map enumerated SMILES of identical molecules nearby in the latent space. Insofar as graphs, Wang et al. (2021) similarly used the NT-Xent loss to maximize the agreement between pairs of augmented graphs ("views") describing the same molecule; here, each view (i.e. positive sample) is obtained by masking out nodes or edges. The NT-Xent loss, although widely successful, operates solely on positive *pairs*, an issue addressed by Khosla et al. (2020) in their formulation of the Supervised Contrastive (SupCon) loss which allows for comparison among an arbitrarily sized *set* (rather than a pair) of positive instances.

Methodology

Overview of Approach

Broadly, our goal is to impart semantic awareness onto a SMILES-based transformer autoencoder such that the effected latent representations better respect the structural similarities, particularly the graph edit distance (GED), between molecular pairs. We do this by incorporating a contrastive component into the architecture that differentiates similar and dissimilar molecular graphs.

Contrastive Objective. Our contrastive task necessitates known pairs of "similar" and "dissimilar" molecules. We opt to consider as "similar" any two molecules separated by a single graph edit. Recall that the graph edit distance (GED) between two molecules, viewed as labeled graphs, is the minimum number of single edits required to make one graph isomorphic to the other. It is computationally infeasible to obtain all pairs of single GED molecules systematically from an existing dataset. To sidestep this issue, we generate a bespoke dataset of 1-GED molecular pairs. We accomplish this by defining a set of node-level transformations, or mutations, which are applied to "anchor" molecules to obtain similar (1-GED) molecules which we will refer to as “mutants”.

Autoencoder Component. We specify our SMILES-based autoencoder with a transformer encoder and decoder, and introduce an intermediate bottleneck in order to obtain fixed-length vector representations. Combined with the contrastive component, the general framework is encapsulated in Figure 2. We note that an encoder trained solely on the contrastive objective, that is, without the reconstruction loss central to an autoencoder, may learn a degenerative mapping such that our designated “similar” molecules are mapped to representations that are in fact *too similar*, being almost stacked on top of one another. In this way, the reconstruction loss provided by the autoencoder component acts as a *regularizer* (on the contrastive loss) that encourages similar molecules to be mapped to *distinct* codes.

Training Dataset

Anchor Compounds. We utilize the dataset developed by Popova, Isayev, and Tropsha (2018), which contains approximately 1.5 million SMILES sequences sourced from the ChEMBL database (version ChEMBL21), a chemical database comprised of drug-like or otherwise biologically-relevant molecular compounds (Bento et al. 2014). After procuring the full dataset, the set of compounds was run through a standard curation pipeline; for an in-depth description of the curation process, please refer to Popova, Isayev, and Tropsha (2018). We further filter the dataset by SMILES length, allowing only molecules with SMILES length less than or equal to 110 characters, leaving 1,256,277 compounds. These compounds constitute the full set of “anchors” from which we will generate 1-GED “mutant” compounds, further explained in the following section.

Generating Mutant Compounds. We define a molecular graph generally as $g = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V} = \{v_0, ..., v_A\}$ is the set of nodes, where each $v_a \in \{\texttt{C}, \texttt{O}, \texttt{N}, \texttt{S}, \texttt{Br}, \texttt{Cl}, \texttt{I}, \texttt{F}, \texttt{P}, \texttt{B}, \$\}$ (atom types), and $\mathcal{E} = \{(v_a, v_b) | v_a, v_b \in \mathcal{V}\}$ is the set of edges (bonds). Note that atom type \$ is a stand-in for any atom type not in the remaining list, analogous to an unk character in natural language models.

Here, we will differentiate anchors from mutants with a tilde, i.e. anchor graphs as g and mutant graphs as $\tilde{g}$. Given an anchor, we consider its graph, $g_i \in G$ where G is the anchor set (sourced from ChEMBL) and i is the index identifying the anchor in G. We obtain a mutated graph, or mutant, by randomly sampling a mutation operator $t(\cdot) \sim \mathcal{T}$ and applying that mutation to the anchor, $t(g_i) = \tilde{g}_{i(j)}$ where i again identifies the original anchor, and j is the index of the mutant graph within the anchors’ positive sample set.

Our set of mutation (graph transformation) operators, $\mathcal{T} = \{Add, Replace, Remove\}$, is rationally defined to avoid transformations that would drastically alter graph topology, i.e. separating a molecule into disconnected graphs or breaking and forming rings. Furthermore, we require mutants to be chemically valid molecular graphs, and we normalize all SMILES using the RDKit canonicalization algorithm (RDKit 2023). Given these specifications, our mutation operators are defined as follows.

- Node addition (*Add*): Append a new node, and a corresponding edge, to an existing node in the graph.
- Node substitution (*Replace*): Change the atom type of an existing node in the graph.
- Node deletion (*Remove*): Remove a singly-attached node and its corresponding edge from the graph.

For both *Add* and *Replace*, incoming atom types are drawn from the observed atom type distribution in the original ChEMBL dataset. For each anchor, g_i, we generate 10 distinct mutants that constitute the “positive” sample set, $P(i)$, for that anchor:

$$P(i) = \{\tilde{g}_{i(1)}, \tilde{g}_{i(2)}, \ldots, \tilde{g}_{i(10)}\} \in \tilde{G} \quad (1)$$

Our final training set is made up of the entire set of anchors and their respective mutants, amounting to 13,819,047 total training compounds. We show an example of a batch

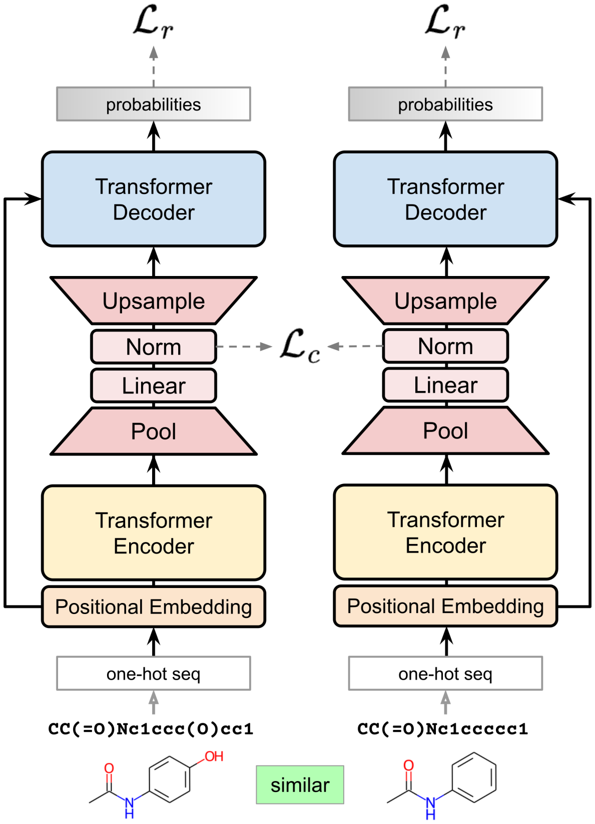

Figure 2: Overview of SALSA architecture. SALSA operates on multi-mutant batches, but here, we show a single (positive) anchor–mutant pair for simplicity. The reconstruction loss ($\mathcal{L}_r$) is computed on the output sequence probabilities. In the case of a positive pair (similar molecules) as shown, the contrastive loss ($\mathcal{L}_c$) aims to push their normalized representations close together in the latent space. Note that weights between the two networks are shared, thus, only a single model is trained and used for inference.

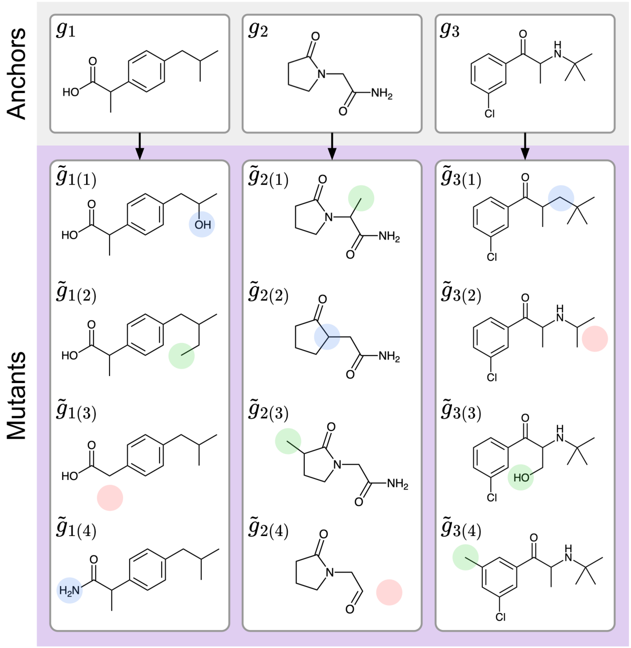

Figure 3: An example batch from the mutated dataset composed of three anchors, g_1, g_2, g_3, and their respective sets of mutants (positive samples), $P(i) = \{\tilde{g}_{i(1)}, \tilde{g}_{i(2)}, \tilde{g}_{i(3)}, \tilde{g}_{i(4)}\}$. Negative samples are defined between anchors and all other molecules in the batch not in that anchor's set $P(i)$. Colored atoms of mutant compounds correspond to single graph edits from anchor to mutant: Add (green), $Replace$ (blue), and $Remove$ (red).

composed of three anchors, with five mutants per anchor in Figure 3.

Faulty-Positive Filtering. Although our mutation operators ensure *chemical validity*, they do not ensure *physicochemical proximity* of mutants to anchors. Due to the complex nature of quantum mechanics underlying molecular interactions, a single graph edit mutation may effect great differences in the physicochemical properties between anchor and mutant. We circumvent such phenomena by filtering out mutants that are *too dissimilar* from their respective anchor based on the Mahalanobis distance between the physicochemical properties of an anchor and those of its mutants. Mahalanobis distance between an anchor g_i and mutant $\tilde{g}_{i(j)}$ is defined as:

$$d_M\left(g_i, \tilde{g}_{i(j)}\right) = \sqrt{\left(x_i - \tilde{x}_{i(j)}\right)^{\mathsf{T}} \Sigma^{-1} \left(x_i - \tilde{x}_{i(j)}\right)} \quad (2)$$

where x_i and $\tilde{x}_{i(j)}$ are the physicochemical property vectors for g_i and $\tilde{g}_{i(j)}$, respectively. The covariance matrix, Σ, corresponds to the distribution of physicochemical properties computed over initial anchor set G. We computed physicochemical properties corresponding to the standard collection of RDKit descriptors, and then filtered out descriptors having any invalid property values in order to obtain a real-valued property vector for each molecule.

Modeling Framework

The core architecture of SALSA is based on the encoder-decoder transformer paradigm proposed by Vaswani et al. (2017), with an additional autoencoder specification. The SALSA transformer takes SMILES sequences as input and additionally considers the similarity relationships between those SMILES inputs (denoted either "similar" or "dissimilar"), as determined by the structural similarity of their corresponding molecular graphs.

SMILES Input. While the original transformer operated on natural language sequences, SALSA operates on SMILES sequences corresponding to molecular graphs. A SMILES (simplified molecular-input line-entry system) sequence is an ordered list of the atom and bond types encountered during a depth-first traversal of a spanning tree of the associated molecular graph (e.g. the SMILES sequence of ibuprofen is "CC(Cc1ccc(cc1)C(C(=O)O)C)C"). We adopt a simple tokenization strategy yielding a vocabulary of 39 tokens, including the most common atom and bond types present in drug-like organic molecules in addition to a start token ,"<", end token, ">", pad token, "X", and an unknown token, "$", used in cases where SALSA encounters an atom type not present in the provided vocabulary.

SALSA Architecture. We modify the original transformer architecture into an autoencoder aiming to reproduce the original input. This is accomplished by introducing a pooling layer and a subsequent upsampling layer between the encoder and decoder, and in this way imposing an autoencoder "bottleneck" that produces fixed-size latent representations. Specifically, whereas the intermediate output of the original transformer encoder is a vector of size $\mathbb{R}^{L\times H}$ for a sequence of length L and hidden dimension size H, SALSA's encoder is designed to output a latent vector of fixed size $\mathbb{R}^S$. This is accomplished by first applying a component-wise mean pooling from $\mathbb{R}^{L\times H} \rightarrow \mathbb{R}^H$ before projecting $\mathbb{R}^H \rightarrow \mathbb{R}^S$.

The SALSA latent vector is constrained to live on the unit hypersphere embedded in $\mathbb{R}^S$, and so we therefore normalize the output of the "Pooling" layer. It is the output of the Pooling layer, $z \in \mathbb{R}^S$, that is input into the contrastive loss, explained in the next section. Then, as the transformer decoder is designed to accept an input of size equal to the output of the encoder, i.e. $\mathbb{R}^{L\times H}$, we first pass the latent vector through a linear layer with the appropriate output dimension and reshape as needed before passing to the decoder. This is referred to as "Upsample" in Figure 2. Note that this method of injecting the latent vector into the transformer decoder resembles the method called "memory" in Li et al. (2020), where it was demonstrated to yield superior results over an alternative strategy.

Loss Function. We define a compound loss function, composed of: (1) a contrastive objective defined over a batch of inputs, and (2) an reconstruction task, characteristic of sequence autoencoders. For our contrastive objective, we adapt the supervised contrastive (SupCon) loss (Khosla et al. 2020). The SupCon loss allows for multiples positive comparisons per anchor, resulting in improved performance relative to naive contrastive losses, which operate on the assumption of only a single positive sample per anchor. The SupCon loss is defined as

$$\mathcal{L}_c = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i \cdot z_p/\tau)}{\sum_{a \in A(i)} \exp(z_i \cdot z_a/\tau)}, \quad (3)$$

where I is the set of anchors in a batch, $A(i)$ is the set of all samples sharing a batch with anchor i, having latent code z_i, and $P(i)$ are those elements of $A(i)$ that are similar to i, and I is the set of anchors in the batch, using the terminology of Section .

The autoencoder, operating on SMILES, is trained with a reconstruction loss with causal masking. For a single SMILES sequence s_i and its associated latent vector z_i, the loss is defined as:

$$\mathcal{L}_r = -\frac{1}{T} \sum_{t=1}^{T} \log p_\theta(s_i^{(t)} | z_i, s_i^{(<t)}), \quad (4)$$

where T is the length of the sequence s_i and $p_\theta(s_i^{(t)} | z_i, s_i^{(<t)})$ is the output of the decoder at position t along the sequence. The full reconstruction loss $\mathcal{L}_r$ is the average of all per-sequence losses. The final loss computation is a weighted combination of the two terms,

$$\mathcal{L} = \lambda \mathcal{L}_c + (1 - \lambda) \mathcal{L}_r \quad (5)$$

where $0 \leq \lambda \leq 1$ is a hyperparameter that weights the contributions of the contrastive loss and the reconstruction loss, respectively. We train SALSA with $\lambda = 0.5$, and make comparisons to either ablation, $\lambda = 1$ and $\lambda = 0$ described later in Experiments and Analysis.

Implementation Details. We use $l = 8$ layers for both the encoder and the decoder with a hidden dimension of size $h = 512$, and $m = 8$ heads in the multi-head attention blocks. Our main results are of models trained with $S = 32$ latent dimensions, although we also investigated reduced latent dimensions, $S \in \{16, 8, 4, 2\}$. For the contrastive loss, we set temperature $\tau = 0.7$, following Khosla et al. (2020).

Experiments and Analysis

We are interested in gathering a comprehensive understanding of SALSA's latent space relative to both a naive autoencoder and a contrastive encoder. To this end, we ask four questions of our SALSA representations:

(1) **Structural Awareness:** Does SALSA encode information about structural (graph-to-graph) relationships?

(2) **Physicochemical Awareness:** Does SALSA implicitly encode information about physicochemical properties?

(3) **Biological Awareness:** Does SALSA inform tasks for biological prediction?

(4) **Semantic Continuity:** Does SALSA produce interpolations that are semantically reasonable?

Baselines

- **Naive Autoencoder:** We are interested in SALSA's performance relative to a naive autoencoder trained solely on SMILES reconstruction. To obtain a naive autoencoder, we trained SALSA with weighting hyperparameter, $\lambda = 0$. We abbreviate this model as "Naive".
- **Contrastive Encoder:** We are also interested in how the reconstruction objective influences the effectiveness of the contrastive task in achieving structural awareness. To obtain a contrastive encoder, we trained SALSA with weighting hyperparameter, $\lambda = 1$. We abbreviate this model as "Contra".

Structural Awareness

In order to evaluate the degree to which representations capture structural awareness, we compute metrics of correlation between Euclidean distance (EuD) in the latent space and graph edit distance (GED) in the data space. Correlation metrics necessitate *a priori* knowledge of GEDs between molecular pairs of interest, not unlike the anchor–mutant pairs generated for our training set. Thus, we extend our mutation process to generate sets of mutants having known GEDs (one to five) from their anchors, which we will refer as "supermutants".

Supermutant Evaluation Set. We extend our mutation process, as defined earlier in the Methodology section, to iteratively generate sets of n-GED supermutants where $n \in \{1, 2, 3, 4, 5\}$. For a given anchor, g_i, we apply a random mutation, $t^{(1)}(\cdot) \sim \mathcal{T}$, to generate a 1-GED (super)mutant, $\tilde{g}_i^{(1)} = t^{(1)}(g_i)$, to which another random mutation operator is applied to generate a 2-GED supermutant, $\tilde{g}_i^{(2)} = t^{(2)}(\tilde{g}_i^{(1)})$, and so-on. One step in this iterative process may be generalized as:

$$\tilde{g}_i^{(n+1)} = t^{(n+1)}(\tilde{g}_i^{(n)}) \quad (6)$$

where $\tilde{g}_i^{(n+1)}$ is the supermutant, and n is the depth of the mutation path, a reliable proxy for the GED between the anchor and mutant. For our supermutants, we use an anchor set independent of the training set drawn from the ChEMBL23 dataset. We draw 5000 random anchors and for each generate n-GED supermutants where $n \in \{1, 2, 3, 4, 5\}$, resulting in 30,000 total compounds. Example of a supermutant set and associated anchor is shown in Figure 4.

GED-EuD Correlation. With our set of independent anchors and associated supermutants, we can evaluate the correlation between GED, $d_{\texttt{GE}}$, between molecular graphs and Euclidean distance, $d_{\texttt{Eu}}$ or EuD, in the latent space. For a given anchor, g_i, and one of its supermutants $\tilde{g}_i^{(n)}$:

$$d_{\texttt{Eu}}(g_i, \tilde{g}_i^{(n)}) = \|z_i - \tilde{z}_i^{(n)}\|_2 \quad (7)$$

$$d_{\texttt{GE}}(g_i, \tilde{g}_i^{(n)}) = n \quad (8)$$

where z_i and $\tilde{z}_i^{(n)}$ are the latent representations of the anchor and the supermutant, respectively. Eq. (7) gives us 5000 EuDs at each n-GED depth $n \in \{1, 2, 3, 4, 5\}$, and we

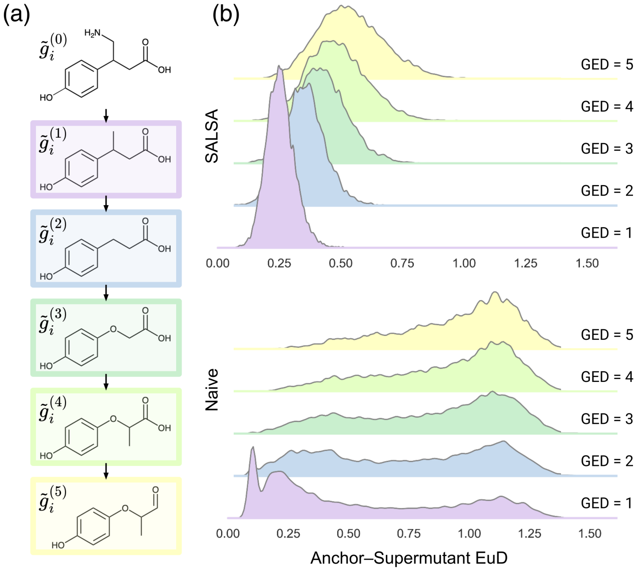

Figure 4: (a) Example of supermutants, $S(i)$, generated from an anchor, g_i. (b) Anchor–supermutant Euclidean distances, by n-GED, in Naive and SALSA latent spaces. For both subfigures, supermutants are color-coded according to n-GED (1-GED: purple, 2-GED: blue, 3-GED: green, etc).

show the resulting distributions in Figure 4(b) for Naive and SALSA. We note that the Contra distribution is practically indistinguishable from SALSA so, for brevity, it is not included in Figure 4(b).

We then calculate two measures of rank correlation, Spearman correlation coefficient (ρ) and Kendall correlation coefficient (τ), between GED and EuD for each anchor-supermutant set, $i \in [1, 5000]$. Then we compute the average Spearman ρ and Kendall τ across all 5000 correlations. We perform this analysis on SALSA, Naive, and Contra space at 32 dimensions, and we further investigated SALSA performance at lower dimensions, $d \in \{16, 8, 4, 2\}$. Results for this evaluation are shown in Table 1.

For this analysis, SALSA and Contra are on par having the highest correlations (among the 32-d spaces), compared to Naive. Although, SALSA does perform slightly worse than Contra, presumably due to the regularizing nature of the reconstruction task. We note that Naive has a wide standard error that is revealed in further detail in Figure 4(b). The bimodal distribution of Naive at n-GED may be interpreted as single graph edits inducing changes to SMILES strings that are either mild (the left mode) or vast (the right mode). SALSA comparatively produces distributions that are consistently unimodal, although the distribution flattens with increasing n-GED indicating that the correlation may not hold between anchors and mutants that are substantially different. Lastly, we find that with decreasing dimensionality, SALSA's performance does not significantly degrade until $d = 4$, suggesting avenues for potential exploration into applications that necessitate operation in exceptionally small dimensional spaces.

Method	Spearman's ρ	Kendall's τ
Naive-32	0.514 ± 0.53	0.467 ± 0.48
Contra-32	**0.888 ± 0.20**	**0.841 ± 0.23**
SALSA-32	**0.878 ± 0.20**	**0.824 ± 0.23**
SALSA-16	0.849 ± 0.24	0.789 ± 0.26
SALSA-8	0.807 ± 0.28	0.741 ± 0.30
SALSA-4	0.587 ± 0.46	0.518 ± 0.42
SALSA-2	0.351 ± 0.57	0.300 ± 0.50

Table 1: Spearman's ρ and Kendall's τ for GED-EuD correlation. We compare SALSA, Contra, and Naive trained at $d = 32$. We further compare SALSA models trained at reduced dimensions. The highest performing methods are in boldface.

Physicochemical Awareness

Although we train SALSA to explicitly learn structural awareness, we would expect to also learn (implicitly) some degree of higher order semantic awareness, such as property or activity awareness. Therefore, we investigated the extent to which latent representations capture information about physicochemical properties. To accomplish this we compute the correlation between property difference (PropΔ) and latent space Euclidean distance (EuD). This evaluation task is inspired from analogous tasks in NLP that correlate embedding similarity and human labels (Luong, Socher, and Manning 2013).

We encode a sample of 2000 molecules into SALSA, Naive, and Contra space, and from those latent representations, compute the EuD between each pair for all three models. Following, we calculate 10 physicochemical properties (chosen for their relevance to drug discovery) for each molecule using RDKit (Fujimoto and Gotoh 2023; Wei et al. 2020). As an illustrative example, Figure 5(a) shows Uniform Manifold Approximation and Projection (UMAP) (McInnes, Healy, and Melville 2018) reductions of a large set of 10,000 compounds in Naive, Contra, and SALSA space, color-coded by "Number of Aromatic Rings".

For each property, we compute the property difference (PropΔ) between each molecular pair. Then, for each of the 10 properties across we compute the Spearman's rank correlation coefficient (ρ) between PropΔ and EuD (for all three models). We perform this analysis on 10 random draws of 2000 molecules to obtain standard error; results are shown in Figure 5(b). We find that SALSA achieves the highest correlation among models for nine out of 10 properties. This is an intriguing finding as it is not obvious as to how SALSA's framework enables better performance over either Naive or Contra. One explanation could be that the contrastive loss works to sharply bring similar molecules close together, creating pockets of *local* organization, while the reconstruction loss enforces regularization such that the clusters disperse achieving more *global* organization.

Biological Awareness

We next investigate biological awareness, evaluated through performance on a virtual screening benchmark task. Vir-

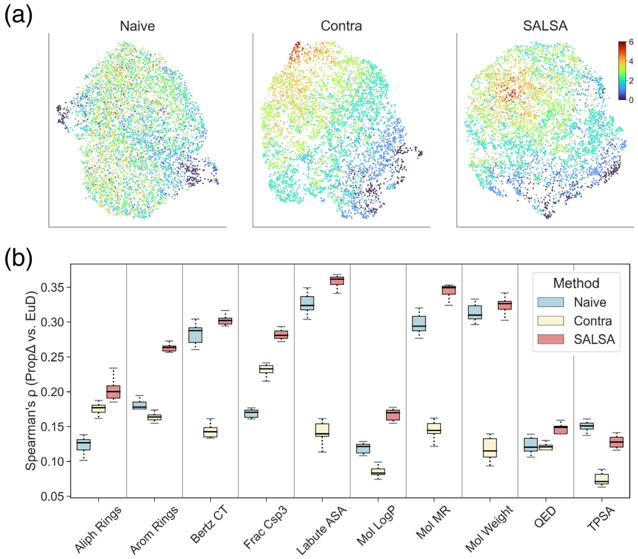

Figure 5: (a) UMAP reduction of 10,000 compounds, color-coded by Number of Aromatic Rings for Naive, Contra, and SALSA spaces. (b) Box plots of PropΔ-EuD Spearman's ρ correlations across 10 physicochemical properties for Naive, Contra, and SALSA spaces.

tual screening is a drug discovery task that involves selecting compounds from a candidate pool most likely to be active against a given biological target, given some prescribed notion of molecular similarity. This task essentially assesses the biological property awareness for a given molecular representation, as sufficiently semantically-aware representations should result more accurate retrieval of active compounds. We also used this benchmark task to compare against other state-of-the-art methods in addition to our ablation baselines.

RDKit Virtual Screening Benchmark. We utilize the RDKit benchmarking platform (Riniker and Landrum 2013), which evaluates a model's virtual screening capabilities against verified protein targets. The benchmark includes 69 protein targets and for each protein target, a dataset composed of a small number of "actives" against the protein and a large number of decoy (inactive) compounds. Given a protein target, the objective is to retrieve active compounds from the collective decoy–actives pool given a fixed number ($n = 20$) of query molecules. We compare SALSA not only to Naive and Contra, but also to a variety of other molecular representations, including handcrafted: ECFP4 (Rogers and Hahn 2010) and RDKit descriptors (RDKit 2023)), SMILES-based: ChemBERTa (Chithrananda, Grand, and Ramsundar 2020), and graph-based: Hu et al. (Hu et al. 2019), and iMolCLR (Wang et al. 2022). We show the resulting overall area under the receiver operating curve (AUROC) for each method in Table 2. SALSA demonstrates superior performance relative to ECFP4 and the Naive autoencoder, and is further competitive against the additionally included deep learning-based methods. The results on this biologically-relevant task further indicate SALSA's *comprehensive* semantic awareness relative to its ablated counterparts, Naive and Contra.

Method	Modality	AUROC
ECFP4	Handcrafted	0.62 ± 0.10
RDKit descriptors	Handcrafted	0.63 ± 0.03
Hu et al.	Graph	0.67 ± 0.10
iMolCLR	Graph	0.57 ± 0.09
ChemBERTA	SMILES	0.68 ± 0.12
Naive	SMILES	0.57 ± 0.07
Contra	SMILES	0.70 ± 0.09
SALSA	**SMILES**	**0.73 ± 0.10**

Table 2: Performance on RDKit VS benchmark. We compare SALSA against its ablations, Naive and Contra. In addition, we compare against ECFP4, RDKit descriptors, and a variety of deep learning-based methods: Hu et al., iMolCLR, and ChemBERTa.

Semantic Continuity (Interpolations)

We investigate SALSA's ability to generate reasonable molecular interpolations between pairs of endpoint molecules, as higher quality interpolations suggest better semantic continuity in the latent space (Shen et al. 2020). To get interpolations, we choose pairs of "endpoint" molecules, calculate the spherical linear interpolation (*slerp*) midpoint (White 2016) between them, and then decode out interpolant molecules from the midpoint code. We do not perform this evaluation on Contra as it lacks a decoder for generating. Figure 6(a) shows a case study of the three most common interpolants for a pair of molecules, for both the SALSA decoder and the Naive decoder. Qualitatively, we can discern that SALSA generates interpolants that are more structurally similar to the endpoints.

We then quantify SALSA's interpolation capability more comprehensively. To this end, we consider five classes of compounds, and for each class, choose a representative set of five molecules. We take all pairwise combinations within each class and determine the most common midpoint interpolants for each pair. Then, to determine "reasonableness" of interpolants, we calculate the Tanimoto distance—a common measure of chemical similarity—between each interpolant and either of their endpoint molecules. Tanimoto distance, d_T, is defined as

$$d_T(b_m, b_e) = \frac{|b_m \cup b_e| - |b_m \cap b_e|}{|b_m \cup b_e|} \in [0, 1] \qquad (9)$$

where b_m is the ECFP4 representation for the midpoint interpolant, and b_e is the ECFP4 representation for either endpoint molecule. Resulting endpoint–midpoint Tanimoto distances are shown in Figure 6(b). SALSA generates interpolants that, on average, have a lower Tanimoto distance (therefore, are more similar) to their endpoints. These results are indicative of improved semantic continuity in the SALSA space relative to the Naive space.

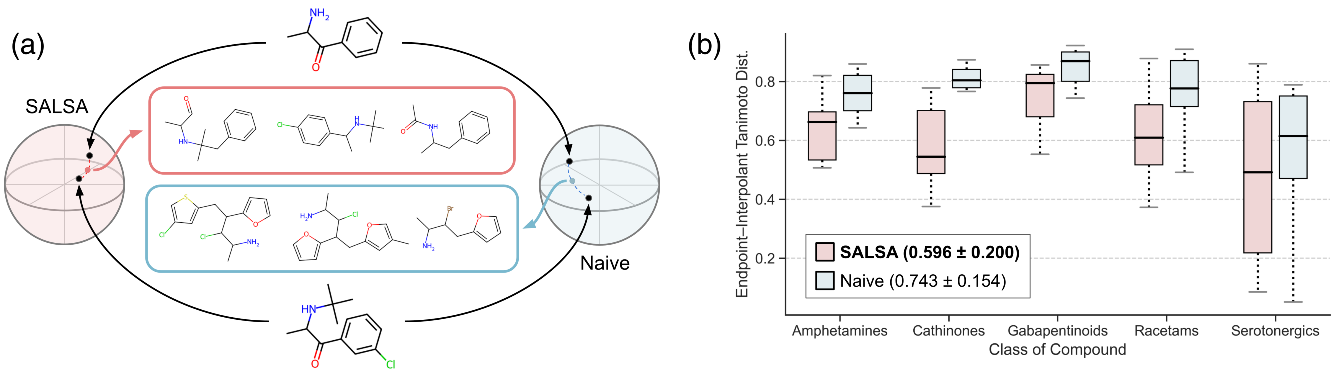

Figure 6: (a) Three most common midpoint interpolants between cathinone (top) and bupropion (bottom), generated from either SALSA space or Naive space. (b) Box plots showing the distribution of endpoint-interpolant Tanimoto distances per compound class for both Naive and SALSA. Legend shows overall mean and standard error (lower is better).

Discussion

Beyond the scope of molecular modeling, we look to provide insight as to how SALSA's methodological basis relates to a larger body of deep learning research. The SALSA paradigm may be viewed as a cousin to denoising adversarial autoencoders (DAAEs) (Makhzani et al. 2015), particularly as applied to text or sequence data (Shen et al. 2020). The goal of the latter work, much like ours, is to coerce a sequence autoencoder to embed related sequences near one another. We opt for an objective function that, although distinct from that of Shen et al. (2020), we argue conceptually accomplishes a similar goal, nonetheless, to that of the DAAE objective.

Our dual objective function for SALSA combines a *reconstruction* loss and a *contrastive* loss, which we claim acts similarly to the dual objective of the DAAE, combining a *denoising* technique and an *adversarial* loss. To support this claim, we refer to the work of (Wang and Isola 2020), wherein it was demonstrated that the contrastive loss, when restricted to latent vectors on the unit sphere and given the limit of infinite negative samples, simplifies into two components: an *alignment* loss and a *uniformity* loss. The alignment loss acts to align the latent representation of positive pairs, while the uniformity loss encourages the distribution of all latent vectors to be uniformly distributed on the unit sphere. Each of these losses has a conceptual counterpart in the DAAE, where the alignment loss acts similarly to the denoising objective and the uniformity loss acts like the adversarial component. In presenting this methodological comparison, we hope to provide a more general context for the techniques explored in SALSA, outside applications to molecular modeling.

Conclusion

In this work, we proposed SALSA, a framework for learning semantically meaningful latent representations. Specifically, we sought to learn molecular representations informed by the structural similarities between molecules. We trained a model with this intention and defined a direct evaluation metric, GED-EuD correlation, to show local structural awareness in latent space. Furthermore, we showed that SALSA produces more semantically reasonable interpolants, and that SALSA implicitly uncovers physicochemical and biological properties, revealing a wider context of latent organization. Although we defined our primary semantic objective to be the *structural* similarities between *molecules*, the SALSA paradigm could be applied to any user-defined semantics based on *x* similarity between *y* data. In this way, the SALSA paradigm could be potentially applied across a number of data types in various domains.

Acknowledgements

This work was supported by the following grants to SMG from the National Institutes of Health - U24DK116204 (NIDDK), U01CA238475 (NCI), R01CA233811 (NCI), U01CA274298 (NCI). (https://www.nih.gov/). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript. T.M. was supported by the National Institute of General Medical Sciences of the NIH under Award Number T32GM086330.

References

Alperstein, Z.; Cherkasov, A.; and Rolfe, J. T. 2019. All SMILES Variational Autoencoder. arXiv:1905.13343.

Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; and Overington, J. P. 2014. The ChEMBL bioactivity database: an update. *Nucleic Acids Res.*, 42: D1083–90.

Bilodeau, C.; Jin, W.; Jaakkola, T.; Barzilay, R.; and Jensen, K. F. 2022. Generative models for molecular discovery: Recent advances and challenges. *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, 12(5).

Bowman, S. R.; Vilnis, L.; Vinyals, O.; Dai, A. M.; Jozefowicz, R.; and Bengio, S. 2016. Generating Sentences from a Continuous Space. arXiv:1511.06349.

Chithrananda, S.; Grand, G.; and Ramsundar, B. 2020. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. arXiv:2010.09885.

Fujimoto, T.; and Gotoh, H. 2023. Feature Selection for the Interpretation of Antioxidant Mechanisms in Plant Phenolics. *Molecules*, 28(3).

Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; and Aspuru-Guzik, A. 2018. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, 4(2): 268–276. PMID: 29532027.

Honda, H.; Lo, Y.-C.; Gui, R.; Torres, J.; and Lee, S.-M. 2016. *Quantitative Methods in System-Based Drug Discovery*. IntechOpen.

Honda, S.; Shi, S.; and Ueda, H. R. 2019. SMILES Transformer: Pre-trained Molecular Fingerprint for Low Data Drug Discovery. arXiv:1911.04738.

Hu, W.; Liu, B.; Gomes, J.; Zitnik, M.; Liang, P.; Pande, V.; and Leskovec, J. 2019. Strategies for Pre-training Graph Neural Networks. arXiv:1905.12265.

Jiang, J.; Xia, G. G.; Carlton, D. B.; Anderson, C. N.; and Miyakawa, R. H. 2020. Transformer VAE: A Hierarchical Model for Structure-Aware and Interpretable Music Representation Learning. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 516–520.

Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised Contrastive Learning. arXiv:2004.11362.

Li, C.; Gao, X.; Li, Y.; Peng, B.; Li, X.; Zhang, Y.; and Gao, J. 2020. Optimus: Organizing Sentences via Pretrained Modeling of a Latent Space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 4678–4699. Online: Association for Computational Linguistics.

Luong, M.-T.; Socher, R.; and Manning, C. D. 2013. Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, 104–113. Sofia, Bulgaria.

Makhzani, A.; Shlens, J.; Jaitly, N.; and Goodfellow, I. J. 2015. Adversarial Autoencoders. *CoRR*, abs/1511.05644.

McInnes, L.; Healy, J.; and Melville, J. 2018. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction.

Montero, I.; Pappas, N.; and Smith, N. A. 2021. Sentence Bottleneck Autoencoders from Transformer Language Models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 1822–1831. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Popova, M.; Isayev, O.; and Tropsha, A. 2018. Deep reinforcement learning for de novo drug design. *Sci Adv*, 4(7): eaap7885.

RDKit. 2023. RDKit: Open-source cheminformatics. https://www.rdkit.org. Accessed: 2023-01-01.

Riniker, S.; and Landrum, G. A. 2013. Open-source platform to benchmark fingerprints for ligand-based virtual screening. *JCIM*, 5(1): 26.

Rogers, D.; and Hahn, M. 2010. Extended-connectivity fingerprints. *J. Chem. Inf. Model.*, 50(5): 742–754.

Shen, T.; Mueller, J.; Barzilay, R.; and Jaakkola, T. 2020. Educating text autoencoders: Latent representation guidance via denoising. In *International conference on machine learning*, 8719–8729. PMLR.

Shrivastava, A. D.; and Kell, D. B. 2021. FragNet, a Contrastive Learning-Based Transformer Model for Clustering, Interpreting, Visualizing, and Navigating Chemical Space. *Molecules*, 26(7).

Sohn, K. 2016. Improved deep metric learning with multi-class N-pair loss objective. In *Advances in neural information processing systems*, 1857–1865.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need.

Wang, T.; and Isola, P. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, 9929–9939. PMLR.

Wang, Y.; Magar, R.; Liang, C.; and Barati Farimani, A. 2022. Improving Molecular Contrastive Learning via Faulty Negative Mitigation and Decomposed Fragment Contrast. *J. Chem. Inf. Model.*, 62(11): 2713–2725.

Wang, Y.; Wang, J.; Cao, Z.; and Farimani, A. B. 2021. MolCLR: Molecular Contrastive Learning of Representations via Graph Neural Networks.

Wei, W.; Cherukupalli, S.; Jing, L.; Liu, X.; and Zhan, P. 2020. Fsp3: A new parameter for drug-likeness. *Drug Discov. Today*, 25(10): 1839–1845.

Weininger, D. 1988. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.*, 28(1): 31–36.

White, T. 2016. Sampling Generative Networks.

Xu, M.; Wang, H.; Ni, B.; Guo, H.; and Tang, J. 2021. Self-supervised graph-level representation learning with local and global structure.