Molecular Property Prediction and Molecular Design Using a Supervised Grammar Variational Autoencoder

Article *in* Journal of Chemical Information and Modeling · February 2022

DOI: 10.1021/acs.jcim.1c01573

3 authors:

André Oliveira
Universidade Federal do ABC (UFABC)
2 PUBLICATIONS **1** CITATION
SEE PROFILE

Juarez L. F. Da Silva
University of São Paulo
344 PUBLICATIONS **7,085** CITATIONS
SEE PROFILE

Marcos G. Quiles
Universidade Federal de São Paulo
102 PUBLICATIONS **605** CITATIONS
SEE PROFILE

Molecular Property Prediction and Molecular Design Using a Supervised Grammar Variational Autoencoder

André F. Oliveira,*,† Juarez L. F. Da Silva,‡ and Marcos G. Quiles*,¶

†*Associate Laboratory for Computing and Applied Mathematics, National Institute for Space Research, P.O. Box* 515, 12227-010, *São José dos Campos, SP, Brazil*
‡*São Carlos Institute of Chemistry, University of São Paulo, P.O. Box* 780, 13560 – 970, *São Carlos, SP, Brazil*
¶*Institute of Science and Technology, Federal University of São Paulo,* 12247-014, *São José dos Campos, SP, Brazil*

E-mail: andre.oliveira@inpe.br; quiles@unifesp.br

Abstract

Some of the most common applications of machine learning (ML) algorithms dealing with small molecules usually fall within two distinct domains, namely, the prediction of molecular properties and the design of novel molecules with some desirable property. Here we unite these applications under a single molecular representation and ML algorithm by modifying the grammar variational autoencoder (GVAE) model with the incorporation of property information into its training procedure, thus creating a supervised GVAE (SGVAE). Results indicate that the biased latent space generated by this approach can successfully be used to predict the molecular properties of the input molecules, produce novel and unique molecules with some desired property and also estimate the properties of random sampled molecules. We illustrate these possibilities by sampling novel molecules from the latent space with specific values of the lowest unoccupied molecular orbital (LUMO) energy after training the model using the QM9 dataset. Furthermore, the trained model is also used to predict the properties of a hold-out set and the resulting mean absolute error (MAE) shows values close to chemical accuracy for the dipole moment and atomization energies, even outperforming ML models designed to exclusive predict molecular properties using the SMILES as molecular representation. Therefore, these results show that the proposed approach is a viable way to provide generative ML models with molecular property information in a way that the generation of novel molecules is likely to achieve better results, with the benefit that these new molecules can also have their molecular properties accurately predicted.

Abbreviations

ML, MS, QC, DNN, DFT, QM9, QM7-X, VAE, GVAE, SGVAE, SMILES

1 Introduction

Data-driven machine learning (ML) techniques have been increasingly used in the computational materials science (MS) domain due to its ability to speed up the process of calculating the physical-chemical properties of materials or even predicting new candidates for further investigation by experimental techniques or conventional quantum-chemistry (QC) calculations.[1,2] Instead of a costly and time-consuming process involving human expertise, computer simulation and subsequent experimental synthesis,[3] ML models can successfully learn the relationship between structure and properties of a molecule in order to perform low-cost predictions. The increasing

availability of open QC databases[4] combined with the development of powerful deep neural network (DNN) architectures[5] has allowed the production of a multitude of such models with a wide range of applications in materials science.[6,7]

Among these applications of ML in MS, two have received a lot of attention: (i) prediction of molecular properties with QC accuracy and lower computational costs compared to standard frameworks such as density functional theory[8] (DFT) and wave function-based methods,[9] or (ii) the design of novel materials candidates from the starting point of a particular property value or desired functionality.[6] As for the former, conventional approaches rely on the training of a DNN to find a relationship between the molecular representation (structure) and the properties of molecular systems as a way to make predictions on unseen data (molecules).[10–16] As for the latter, the common procedure usually involves the use of a deep variational autoencoder[17] (VAE) model to map the molecules as a simple and continuous distribution over a latent space, from which one can sample for new data points that can be translated into novel molecules with analogous structures as those in the original dataset.[18–21] Recently, the use of recurrent neural networks (RNN) have also gained attraction for molecular design.[22–25]

Molecular graph representations are arguably more suitable for the property prediction task, as they can incorporate more information about the 3D configuration of a molecule and thus encode additional chemical information in comparison to linear notations.[6,26,27] On the other hand, and largely due to the fact that there exists powerful models for text sequence modeling,[28] the molecular representation known as simplified molecular-input line-entry system (SMILES)[29] is the most commonly employed representation for molecular design using the VAE framework.[6] SMILES represents the molecular structure using a sequence of characters that denote the atoms by their atomic symbols. Furthermore, it can also represent topological features such as bonds and branches, with the use of special characters and parentheses, respectively.[29]

Here, we built on previous studies and unite these approaches under an unique molecular representation and DNN architecture. To do so, we follow the study reported by Bombarelli et al.,[18] but instead of modifying the standard VAE,[17] we adapt the grammar variational autoencoder (GVAE)[28] model by incorporating a predictor network in its architecture in order to predict the properties of the input molecules based on their latent representation.

Therefore, as a result of the present ideas, we proposed a supervised GVAE (SGVAE) algorithm. The loss function of both the GVAE and predictor network are concatenated and set to be minimized altogether as part of the training procedure. In this sense, the SGVAE can be seen as a special case of semi-supervised deep generative models[30,31] where all labels are available, and thus not having the need to use a predictor network to estimate the missing values. Instead, we use the predictor network to bias the training process in order to generate a latent space arranged by property value.

The resulting biased latent space should not only allow the sampling of novel molecules, but also facilitates the identification of relationships between the molecular structure and molecular properties in a way that the latter could be more accurately inferred by a ML model. A feed forward neural network is then trained to predict the properties of unseen molecules using their latent representation generated by the SGVAE.

We evaluated the SGVAE performance using three datasets, QM9,[32] PubChemQC,[33] and QM7-X,[34] however, our discussion will be based primarily on the results obtained for the QM9[32] and PubChemQC[33] databases, while the results for the QM7-X[34] are reported within the electronic supporting information (ESI). For the QM9, we tested the model's predictive capacity in a total of 7 physical-chemical properties, while only the energy of the lowest unoccupied molecular orbital (LUMO) was employed for the PubChemQC compounds. We also report the results of a number of metrics evaluating how the incorporation of property information during the GVAE training improved the generation of novel and unique molecules, and also allowed the accurately prediction of properties values from unseen data. Comparison with values of such metrics reported in the literature are provided as well, showing that the presented approach can be a fast alternative to unit these two important applications of ML models into the materials science field.

2 Methods

2.1 Molecular Databases

Here, we summarized the most important features of the selected molecular databases to test the SGVAE performance, while additional computational details can be found elsewhere, e.g., QM9,[14,32,35] PubChemQC,[33,36] and QM7-X.[34,37]

2.1.1 Quantum-chemistry QM9 Dataset

The QC QM9 dataset is a public dataset that holds information on the energetic, electronic, and structural properties of ∼134 kilo molecules composed of hydrogen, carbon, nitrogen, oxygen, and fluorine.[32] It has been often used for the training and testing of machine learning models, specially those which seek to identify relationships between the structure and the properties of molecules.[11,14,38]

Instead of using the original QM9 dataset, we chose to perform our study using the modified QM9 dataset reported by Pinheiro et al.,[14] where the authors performed several analyses to rule out molecules that presented inconsistencies within the dataset, which could somehow affect the predictive capacity of the trained ML models.[14] The revised QM9 dataset reports the SMILES string representation of 130 127 molecules, both from GDB-17[39] and from the DFT-B3LYP/6-31G-(2df,p) relaxed geometries, as well as the same 15 physical-chemical properties reported in the original dataset, however, using the atomization energies calculated by Faber et al.[9]

For computational facility, we used the RDKit python package[40] to read and convert to canonical format the SMILES representation from the GDB-17, extracting the production rules of strings with up to 34 characters. Out of 15 physical-chemical properties presented in the dataset, we tested our model for a total of 7 properties, which are essential to determine a wide range of physical-chemical behaviors of a molecule. The chosen properties are:

1. Three thermodynamic properties: internal energy of atomization at 0 K (U_0), internal energy of atomization at 298.15 K (U), and enthalpy of atomization at 298.15 K (H).
2. Highest occupied molecular orbital (HOMO) energy ($\varepsilon_{\text{HOMO}}$), LUMO energies ($\varepsilon_{\text{LUMO}}$), and the energy separation between the HOMO-LUMO states (Δ_ε).
3. Dipole moment (μ).

All quantities are given in eV, while μ is in Debye.

2.1.2 PubChemQC PM6 Dataset

The PubChemQC PM6,[33] herein called just PubChemQC, is a public available dataset which reports the optimized molecular geometries and electronic properties calculated by the semi-empirical PM6 method for 94.0 % of the 91.6 million molecules cataloged in PubChem Compounds.[41] In addition to neutral states, it also reports the calculations for the cationic, anionic, and spin flipped electronic states of 56.2 %, 49.7 %, and 41.3 % of the molecules, respectively, resulting in a total of 221 million PM6 calculations.

In this study we randomly select a small subset of 100 000 molecules with up to 50 SMILES characters, 16 more in comparison to the QM9 dataset, and composed by the hydrogen, carbon, nitrogen, oxygen, fluorine, phosphorus, sulfur, and chlorine atoms, i.e., it includes additional species beyond the QM9 dataset. Finally, to train the model using property information, we chose to use the calculated values for the $\varepsilon_{\text{LUMO}}$ energy reported in the database.

2.2 Model's Architecture

We adapted the original GVAE[28] to incorporate information of the molecular properties from the input data into the training procedure, thus creating a supervised GVAE (SGVAE). Analysis are then carried out to investigate how the changes in the GVAE loss function affected both, the configuration of the latent space and the overall performance of the model.

The GVAE directly incorporates information about the structure of discrete data by using a grammar. In this context, a formal grammar describes the process of creating a string based on the symbols of a language, which are valid according to the language's syntax.[42] For instance, the SMILES grammar has a set of specific symbols and rules that demonstrates how

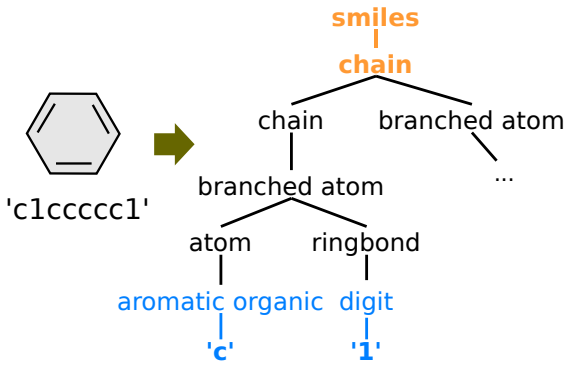

Figure 1: Subset of a parse tree exhibiting part of the production rules necessary to generate the SMILES string representing the benzene molecule

to generate a string representing any molecule. Given such grammar, every valid discrete object can be described as a parse tree from the grammar,[28] where a parse tree represents the syntactic structure of a string according to the grammar. The GVAE encodes and decodes directly to and from these parse trees, which ensures that all outputs are valid based on the grammar.

Figure 1 exemplifies a parse tree generated from the rules of production needed to create the SMILES string for the benzene molecule. The start rule is denoted in orange and the rules that decode to terminals, i.e., each character of the SMILES string for the benzene molecule, are all in blue. For each input molecule, the GVAE extracts the rules of production from its SMILES string via a parse tree. Each rule is then converted to a one-hot vector and the collection of such vectors can be written as a matrix with dimensions of T(**X**) x K, where T(**X**) is the number of productions applied in total to generate the output string **X** and K is the total number of production rules in the entire grammar. A deep convolutional neural network is used as an encoder to map such matrix in the latent space.

A decoder, in the form of a deep recurrent neural network, maps a point in the latent space back to the T(**X**) x K matrix while it keeps track of the state of the parsing to ensure that any sequence of production rules generated from the decoder is valid, thus masking out any invalid rule. We use the same network architecture for both the encoder and decoder as the original implementation.

To incorporate information of the input molecule's property into the GVAE model, a feed forward neural network (FNN) was added to the GVAE's architecture in order to "shape" the latent space by property value. The input for the FNN is the latent representation of the production rules regarding each molecule, whereas its output is the predicted property value of the input molecule. Figure 2 shows the model's overall architecture. Notice that only a small portion of the GVAE's architecture is displayed. The GVAE and the FNN are jointly trained and the loss function to be minimized is given by the following equation,

$$\mathscr{L}_{SGVAE} = \mathscr{L}(x;\theta,\phi) + \frac{1}{N}\sum_{i=0}^{n} ||y_i - \hat{y}_i||^2 \,, \quad (1)$$

where the first term in equation 1 is the evidence lower bound (ELBO), the loss function of the standard VAE, while and the second term is the mean squared error (MSE) for the predictor network on Figure 2, with y_i being the true property value and $\hat{y}_i$ the predicted property value.

Therefore, the model will be encouraged to learn a latent representation of a molecule that will minimize both the ELBO and the mean squared error, arguably shaping the latent space in such a way that the mapping between a property value and a molecular representation is prone to yield better results. To test this hypothesis, a second FNN was trained to predict molecular properties using the latent representation of the input molecules generated by both models, the SGVAE and the standard GVAE, as a way to investigate whether the changes in latent space configuration were indeed beneficial. Results are reported using the mean absolute error (MAE).

2.3 Experimental Setup and Model Evaluation

For the encoder, we used three 1D convolutional layers of filter sizes 9, 9, 10 and 9, 9, 11 convolution kernels, respectively, followed by one fully connected layer of width 435. The latent space dimension was set to be 56 for all runs. The decoder fed into 3 layers of gated recurrent unit (GRU) networks[43] with hidden dimension of 500 neurons. The last layer of the RNN decoder defines a probability distribution over all possible characters at each position in the string of a SMILES representation.[18]

The FNN which was incorporated into the

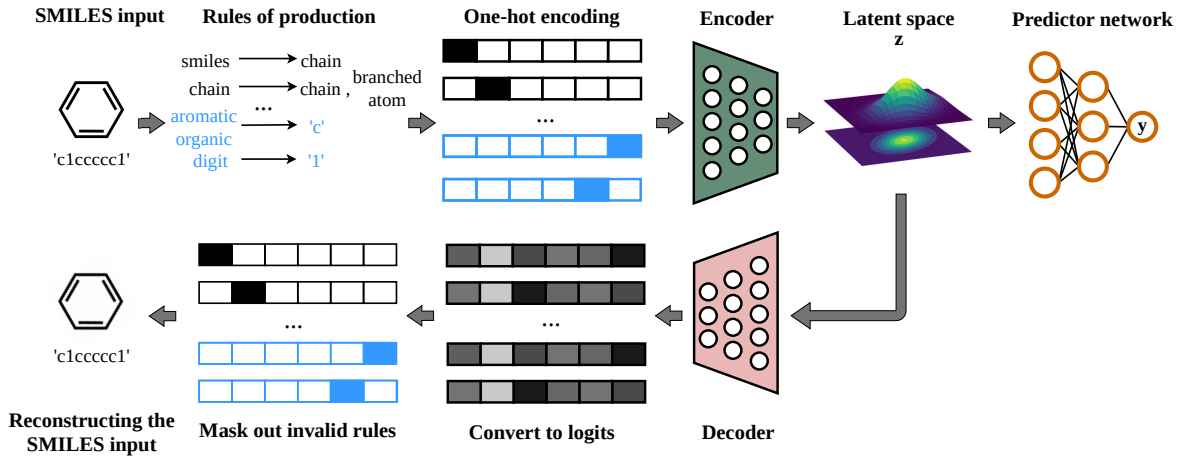

Figure 2: Model's architecture. The production rules of each SMILES input are mapped into the latent space in the form of a one-hot encoding matrix. These latent representations z are fed into the predictor network that learns to map such representations to a property value y.

Table 1: Parameters and hyperparameters used to tune the FNN for molecular property prediction of unseen data

Parameter/Hyperparameter	Values
number of layers	2, 3
number of neurons per layer	64, 128, 256
drop out rate	0.2, 0.3, 0.5
batch size	256
activation function	ReLU

GVAE's architecture has 3 fully connected hidden layers with 70 neurons on each and a batch normalization layer preceding the output layer, which is composed by 1 neuron with linear activation. All hidden layers have the rectified linear unit (ReLU) as activation function. As for the FNN to predict the molecular properties of unseen molecules based on their latent representation, herein called predictor model, its parameters and hyperparameters were optimized for each property by using a search strategy known as grid search. Table 1 shows the set of parameters and hyperparameters used to tune such FNN.

Following previous implementations, we randomly selected 5000 molecules as hold-out set from each dataset to perform reconstruction accuracy tests and analyze the prediction ability of the predictor model. To train the SGVAE, we split the remaining data into 90 % training and 10 % validation. Regarding the predictor model, we used the k-fold cross validation technique with k=10 to train the model. The same splitting percentages were used and the hold-out set was employed to test the predictor model on unseen data. We report the mean and standard deviation of the prediction error over five trials for each property. It's also important to point out that before being fed to the model, we first divided the ground-state properties values by the number of atoms of each molecule and then scaled the input data to have a mean of zero and a standard deviation equals to unit. In the Supporting Information material we show the molecular property distribution of both training and test set.

After training the SGVAE on each property we measured both its reconstruction accuracy and prior validity. To compute the reconstruction accuracy, for each molecule in the hold-out set we encoded it 10 times and decoded it 10 times, as encoding and decoding are stochastic. This resulted in 100 decoded molecules for each of the 5000 molecules in the hold-out set. We then computed the average of these 100 decodings that are identical to the input molecule. Finally, we average these averages across all 5000 inputs to get the percentage of molecules that correctly reconstruct out of 500 000 attempts.

To compute the percentage of prior validity, 1000 latent points were sampled from the prior $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. Each of these points were decoded 100 times and the validity of the resulting SMILES strings where checked using the RDKit. We averaged across all 1000 points and 100 trails to evaluate the percentage of valid SMILES strings in a sample size of $N_s = 100\,000$ decoded molecules. Both evaluation were performed considering all 7

chosen properties for the QM9 dataset and the $\varepsilon_{\text{LUMO}}$ for the PubChemQC dataset, as a way to directly compare how changes in the latent space configuration may affect the ability of the model to correctly reconstruction an input and also sample for valid molecules. Finally, we also measured the number of novel and unique molecules generated by the model. The number of novel molecules n_l was defined as the number of valid decoded molecules from the sample size N_s that were not present in the original dataset, while the number of unique molecules corresponded to the set of unique SMILES strings from n_l, *i.e.*, $unique = set(n_l)/n_l$. We analyzed the performance of the vanilla GVAE using the same metrics.

3 Results and Discussion

Here we reported the performance of the SGVAE model on both, property prediction and molecular design tasks. Comparison with the literature is also provided together with an analysis on how training the model with different properties can affect its results.

3.1 Property Prediction

3.1.1 QM9 Dataset

To compare the effects of jointly training a feedforward neural network within the GVAE architecture, Figure 3 shows a two-dimensional principal components analysis (PCA) on the latent space generate by both network architectures and colored by the values of the $\varepsilon_{\text{LUMO}}$ property for 30 000 randomly selected molecules. Differences in the distribution are evident and the latent space generated by the grammar variational autoencoder jointly trained with the feed forward neural network shows a distinct gradient by property value. In fact, this gradient allows for a linear mapping between each molecule and its respective property. Similar comparison for the other ground-state properties considered in this work can be found in the Supporting Information material. Such changes in the latent space configuration may indicate better regions where the sampling for new molecules is likely to yield preferable candidates to maximize or minimize a target property. Moreover, the property prediction task is expected to achieve better results, as close regions in the latent space are likely to have molecules with similar property values and, possibly, similar molecular structures that could be identified by a machine learning algorithm.

To test the latter hypothesis, the predictor model was trained to predict the ground-state properties based on the latent space generated by both trained models. Table 2 compares the resulting MAE for the GVAE and SGVAE models with other methods in the literature which also use SMILES as molecular representation. Best results for each property are presented in bold. The SGVAE model scores better in all properties in comparison to the standard GVAE. This is expected since, as seen in Figure 3, the "shaped" latent space may facilitates a map between a point and a property value.

Better results, however, are mostly achieved by a FNN model[14] directly trained on molecular descriptors derived from the SMILES representation and obtained using the mordred[44] python package. Even so, the SGVAE model scored lower MAE values for three out of seven properties even when compared to ML models directly engineered for property prediction using SMILES representation as the backbone. Not only that, the MAE for the remaining properties scored fairly close in comparison to the FNN model. As for the character VAE (CVAE) model,[18] the authors also incorporated a FNN to a VAE's architecture to predict molecular properties. However the model offers inferior results both on MAE and in metrics related to molecular design, probably due to the lack of constraints in the process of generating new SMILES strings and also because how the incorporation of property information was performed.

Although the results have improved, using SMILES as molecular representation for predicting ground-state properties of a set of molecules is not common in the literature.[14] In fact, it is arguably more beneficial to derive molecular descriptors by using graph representation and thus consider the 3D information of a molecule.[6,45,46] As an example, the dipole moment property is regarded as a difficult property to correctly be predicted by methods using the SMILES as molecular representation due to its strong dependence on the atomic spatial arrangement, which is

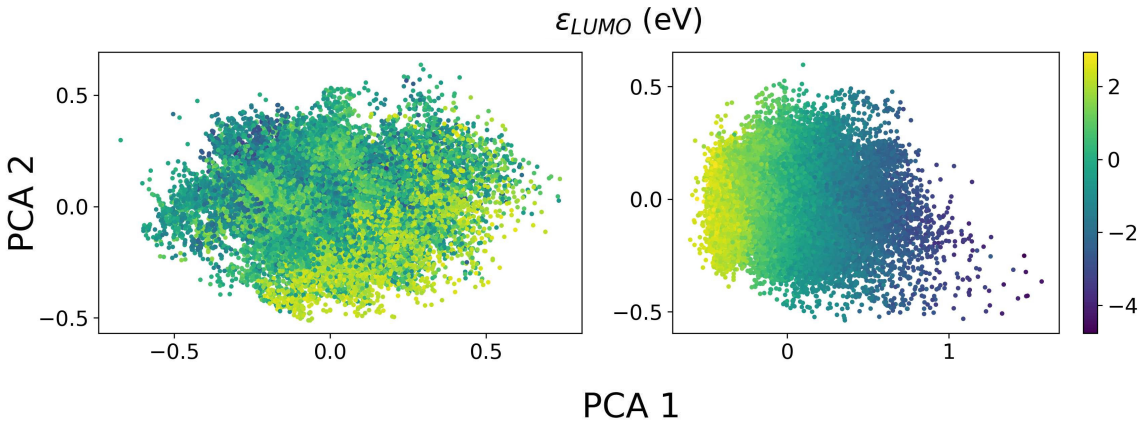

Figure 3: Two-dimensional PCA analysis of the latent space generated by the GVAE on the left, and the SGVAE on the right for the molecules on the QM9 dataset. Colors indicate the values of the $\varepsilon_{\text{LUMO}}$ property for each point in the latent space and the color bar shows the range of $\varepsilon_{\text{LUMO}}$ values.

Table 2: Comparison of MAE for molecular property prediction using different ML algorithms

Properties	GVAE	SGVAE	CVAE[18]	FNN[14]
$\varepsilon_{\text{LUMO}}$ (eV)	0.244±0.007	**0.098±0.002**	0.16	—
$\varepsilon_{\text{HOMO}}$ (eV)	0.260±0.005	0.125±0.002	0.16	**0.0952±0.0007**
Δ_ε (eV)	0.327±0.009	0.186±0.005	0.21	**0.137±0.001**
μ (Debye)	0.441±0.007	**0.291±0.002**	—	0.523±0.006
U_0 (eV)	0.239±0.008	0.066±0.002	—	**0.0573±0.0004**
U (eV)	0.232±0.006	**0.056±0.003**	—	0.0582±0.0005
H (eV)	0.241±0.006	0.066±0.006	—	**0.0575±0.0007**

not described by the SMILES.[14] Even though we have achieved a relatively good result in predicting this particular property, coming close to the chemical accuracy of ∼0.1 Debye, state-of-the-art graph methods have presented MAE's well below this threshold, yielding values of, for example, 0.101 Debye,[9] 0.04 Debye,[47] 0.033 Debye,[48] and 0.03 Debye.[11]

Yet, such improvements often come with a higher price in terms of computational resource, as training graph-based models can take orders of magnitude longer than simpler models.[9] Nonetheless, it is interesting to point out that the change in the GVAE's architecture yielded results that still came close to the chemical accuracy of ∼0.1 Debye for the dipole moment and also of ∼0.05 eV for the atomization energies, where the chemical accuracy represents the desired target errors for the chemical community.[14] This is specially remarkable when considering that the GVAE is, in essence, a generative model. The Supporting Information material also offers a comparison between the Pearson correlation coefficient for predicting property values considering both the GVAE and SGVAE models. As expected, the SGVAE model achieved the best results.

3.1.2 PubChemQC Dataset

Figure 4 shows the changes in the latent space configuration with the use of the $\varepsilon_{\text{LUMO}}$ property information during the SGVAE training on the PubChemQC dataset. Although the latent space configuration by property value is not as visible as in Figure 3, it is still possible to identify a shift in the overall values of the $\varepsilon_{\text{LUMO}}$ property in the PCA 1 direction in Figure 4. Interestingly, such change in configuration does not seem to take place when property information is embedded in the objective function of the VAE via a condition vector as done in the work of Lim et al.,[49] where the authors simply concatenate the one-hot vectors representing each molecule with the condition vector representing the corresponding property.

The modification of the latent space was consistent across all properties and datasets used to test our model and, as discussed previously,

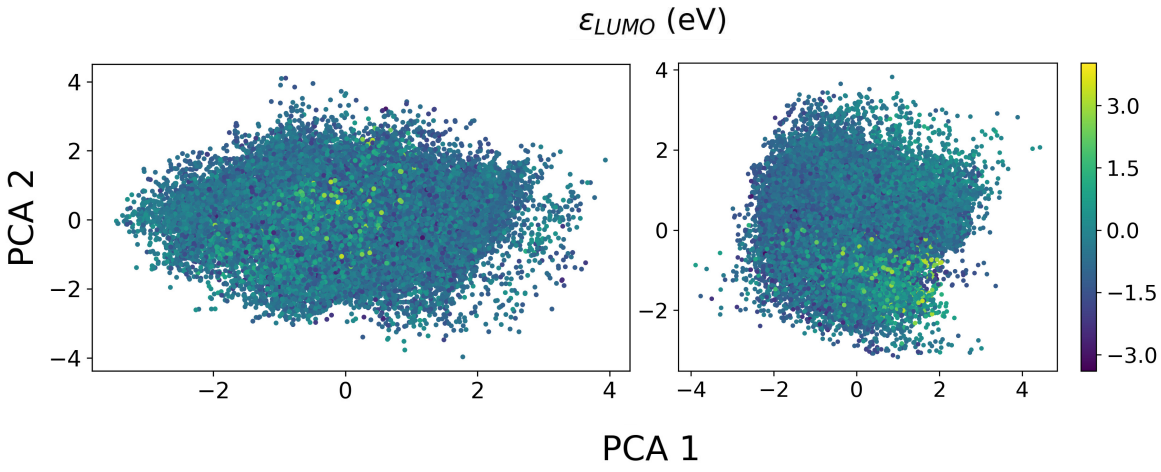

Figure 4: Two-dimensional PCA analysis of the latent space generated by the GVAE on the left, and the SGVAE on the right for the molecules on the PubChemQC. Colors indicate the values of the $\varepsilon_{\text{LUMO}}$ property for each point in the latent space and the color bar shows the range of $\varepsilon_{\text{LUMO}}$ values.

this should be enough to enhance the performance of downstream task such as molecular property prediction. Indeed, using the latent representation generated by the SGVAE, the MAE for predicting the $\varepsilon_{\text{LUMO}}$ of the molecules in the hold-out was (0.111 ± 0.002) eV, while it reached as high as (0.569 ± 0.005) eV using the latent features generated by the GVAE as molecular representation. Unfortunately, for being a recent proposed dataset, there are no study up to this point which uses the PubChemQC for property prediction as it is commonly done with the QM9 dataset. As such, we are not able to directly compare our results with values outside the scope of this work. Nonetheless, the improvement in the MAE value show is consistent to which was observed using the QM9 dataset and thus it demonstrates that the proposed model can indeed be used to improve the predictive performance of a generative model.

3.2 Molecular Design

3.2.1 QM9 Dataset

As a generative model, we performed several analyses to evaluate the ability of the GVAE to generate valid molecules when jointly trained with a FNN. To visually inspect the smoothness of the latent space, we first encoded a molecule in the latent space, generating the vector $\boldsymbol{z}_0$. Following previous implementations, two random orthogonal unit vectors, $\boldsymbol{x}$ and $\boldsymbol{y}$, were then generated, defining a grid over the latent space. New vectors in this grid can be access by combining $\boldsymbol{z}_0$, $\boldsymbol{x}$ and $\boldsymbol{y}$ using a equation in the form,

$$\boldsymbol{z} = \boldsymbol{z}_0 + \boldsymbol{x} \times dx + \boldsymbol{y} \times dy\,, \tag{2}$$

where dx and dy scale the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ to only search in the neighborhood of the encoded molecule. For each vector in this grid we decoded it 100 times and only kept the most commonly generated molecule. Figure 5 shows the resulting grid of molecules generated by the standard GVAE, left, and with the addition of the FNN trained using the $\varepsilon_{\text{LUMO}}$ property on the right. Even though the same molecule with SMILES *O=CC1(COC=N1)C#C* was initially defined as $\boldsymbol{z}_0$, a slight different molecule was sampled as $\boldsymbol{z}_0$ in the right side of Figure 5. Due to the stochasticity of the decoder, such result is likely to occur, specially when the generated molecule only differs by one atom from the original encoded molecule. This shows the smoothness of the latent space created by the model, and the obtained grid of molecules is similar to what is commonly found in works on molecular design.[50]

To make use of the possibility of predicting the property of novel molecules, Figure 6 shows random molecules sampled from the prior for the model trained with the internal energy at room temperature (U) property. All molecules are novel and their predicted U values in electron-volt are also shown. Notice that similar molecules are predicted to have similar property values, as it would be expected by the changes in latent space that are likely to occur as depicted in Figure 3. This is perhaps the greatest benefit

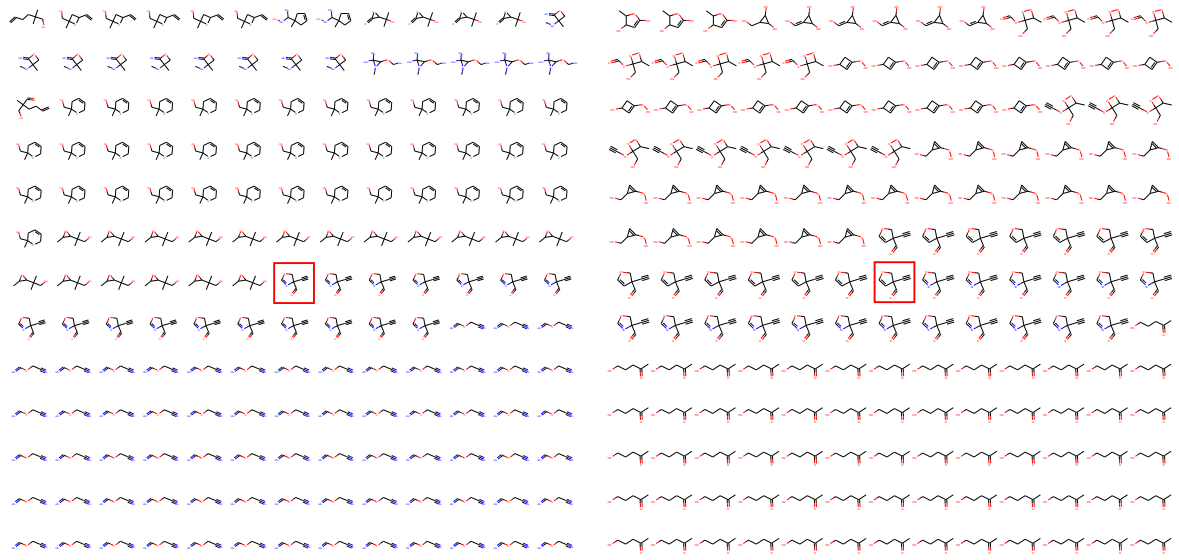

Figure 5: Local neighborhood from the molecule highlighted in the center. **Left**: Generated using the standard GVAE. **Right**: Generated using the model SGVAE when trained with the energy of LUMO property.

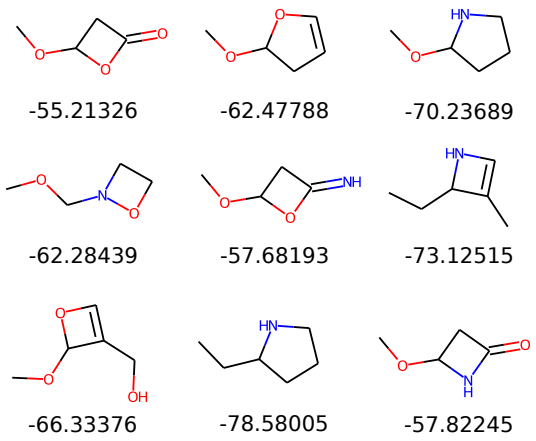

Figure 6: Random molecules sampled from the prior with the model trained using the internal energy at 298.15 K (U) property. Values under each molecule indicate the predicted U values in eV.

of incorporating property prediction tasks into a generative model, as one can use it to not only generate novel molecules, but also have a good estimate on its properties.

However, from the fact that the neural network gives good predictions of properties on the hold-out set, it may not follow that predictions of the same quality will be achieved on the novel generated molecules. To test it, we randomly selected 50 molecules from the hold-out set, for which the properties are known, and mapped them in the latent space using the model trained with the $\varepsilon_{\text{LUMO}}$ property. For each molecule, we then sampled a total of 100 new molecules using equation 2 and used the property prediction model to predict the $\varepsilon_{\text{LUMO}}$ of these molecules. Having these predictions we could check the correspondence of the properties of the generated molecules to the desired values represented by the molecules from the hold-out set that were mapped in the latent space. According to Figure 3, points in the neighborhood of an encoded molecule should be translated into molecules with similar property values.

Results indicate a less accurate prediction, with an average MAE value of (0.347 ± 0.029) eV and an average Root Mean Squared Error (RMSE) value of (0.199 ± 0.033) eV when comparing the predicted $\varepsilon_{\text{LUMO}}$ values with the true values of the 50 molecules from the hold-out set. This reduction in the accuracy of the predictions might be explained by the difficulty of the model in predicting the properties of molecules from low dense regions in the latent space. For example, using the molecule with SMILES *CC1(C#N)C(=O)N2CC21* and $\varepsilon_{\text{LUMO}}$ of -1.1537 eV as seed, the resulting MAE and RMSE were much lower, 0.146 eV and 0.046 eV respectively, and the mean $\varepsilon_{\text{LUMO}}$ value of the 100 newly generated molecules in its neighborhood were (-1.120 ± 0.048) eV. Judging by Figure 3, there is a considerable amount of molecules with $\varepsilon_{\text{LUMO}}$ values between -2 eV and 0 eV that the model had access during training and thus it makes sense that better predictions would be achieved in such range.

On the other hand, using the molecule with SMILES *Cn1ncnc(F)c1=O* and $\varepsilon_{\text{LUMO}}$ of -2.0871 eV as seed, both the MAE and RMSE were much higher, 0.656 eV and 0.492 eV respectively, and the mean $\varepsilon_{\text{LUMO}}$ value of the generated molecules were (-2.743 ± 0.060) eV. Figure 3 shows that there are indeed fewer molecules in the range from -2 eV up to -4 eV when considering the $\varepsilon_{\text{LUMO}}$ energy, probably being the reason for a much poorer performance in such regions. Therefore, one should be cautious when predicting properties of novel molecules originated in low populated regions over the latent space when using the SGVAE. Although the mean value of the predicted properties of the novel molecules generated in such regions might still be in an acceptable range, it is much more difficult for the model to pin point a molecule with the exact desired property value. The same behavior where observed for the remaining properties.

Random sampled molecules and their predicted properties for the model trained with all other properties, similar to Figure 6, can be seen in the Supporting Information material. It's interesting to notice how different these molecules are depending on which property the model was trained on. For example, the model tended to sample molecules with a four-member ring structures when trained with the $\varepsilon_{\text{HOMO}}$ energy (e.g., *CC1N=C(CN)C1=N*) and with a chain like structure when trained with the dipole moment (e.g., *CC=CC=NC=CN*). Different tendencies can also be seeing for the remaining properties, meaning that, even though we didn't specifically applied an optimization algorithm over the latent space, incorporating molecular properties information was already enough to bias the latent space to the point where a specific type of molecule is more likely to be sampled by the model.

Table 3 summarizes the performance of the model measured by reconstruction accuracy, percentage of novel molecules and percentage of unique molecules by chosen property. Bold values indicate the best value by property and metric. In respect to the vanilla GVAE, the reconstruction accuracy reached a value of 59.88 %, similar to the value of 53.7 % found in the original paper. However, the biggest difference between measures was to be found on prior validity. While it reached 40.65 % is this work, it was only 7.2 % in the original GVAE paper. Such discrepancy might be caused by the different datasets used for testing the machine learning models, as the authors used the ZINC database[51] in their work. The molecules in such database are more complex in comparison to the ones in the QM9, thus having a longer SMILES sequence that needs to be correctly generated. Even graph-based variational autoencoders may scores rather low on the ZINC database.[52] As for the percentage of novel and unique molecules, we obtained values of 50.17 % and 8.63 % respectively. It means that, although the number of novel molecules was reasonably high, only a fraction of these new molecules were unique, i.e., the model sampled the same molecules repeatedly from the prior.

Values on Table 3 show that the addition of the FNN into the GVAE architecture actually had an overall positive effect in three out of four metrics, namely, the reconstruction accuracy, prior validity and percentage of unique molecules. The percentage of novel molecules was the only metric to present lower results for the SGVAE model in comparison to the standard GVAE. It measures the amount of new molecules in the fraction of valid molecules sampled from the prior. As an example, in a trial of 100 000 sampling attempts from the prior, a total of 63 510 (63.51 %) points were decoded into valid SMILES strings according to the RDKit python package for the model trained with the Δ_ε property, as shown in Table 3. From this total, 33 947 molecules were novel (53.45 %) and, from the latter, 8745 were unique (35.76 %), meaning that the majority of the novel molecules were sampled repeatedly. Similar analysis can be done for the remaining properties. However, even though the number of novel molecules was higher for the standard GVAE, the percentage of unique molecules was by far the lowest. In a total of 20 394 novel molecules, only 1760 (8.63 %) were unique, making the vanilla GVAE the worst in terms of expanding the sample space beyond the original dataset.

Original generative studies usually do not evaluate their models considering the influence of ground-state properties in its training procedure nor commonly test their algorithms using the QM9 dataset. Thus, as a way to compare the results from Table 3 with reported values, we use the baseline values of different VAE-based

Table 3: SGVAE performance measured by the metrics reconstruction accuracy, prior validity, novel and unique molecules for each of the chosen properties. All values are in percentages

Properties	Reconstruction	Prior validity	Novel	Unique
$\varepsilon_{\text{LUMO}}$ (eV)	82.27	57.89	30.36	20.85
$\varepsilon_{\text{HOMO}}$ (eV)	47.28	62.04	16.24	18.19
Δ_ε (eV)	73.33	63.51	**53.45**	25.76
μ (Debye)	73.42	67.24	18.20	31.31
U_0 (eV)	**92.70**	47.62	35.30	43.88
U (eV)	58.27	66.20	22.46	27.08
H (eV)	55.43	**75.75**	9.07	**49.01**

Table 4: Compilation of results for validity, novelty and uniqueness using different VAE-based algorithms and the QM9 dataset. Values for the models GraphVAE, CVAE, Syntax Directed VAE, Regularized Graph VAE, Junction Tree VAE and Constrained Graph VAE are reported by refs[52,53]. All values are in percentages.

Model	Prior validity	Novel	Unique
GraphVAE	55.70	61.60	76.00
CVAE	10.30	90.00	67.50
GVAE	40.65	50.17	8.63
Syntax Directed VAE	15.00	**100**	**100**
Regularized Graph VAE	87.71	41.26	83.13
Junction Tree VAE	99.95	91.14	90.27
Constrained Graph VAE	**100**	92.82	98.86
SGVAE (Δ_ε – DFT-B3LYP)	11.65	93.33	96.42

algorithms trained on the QM9 dataset and obtained by Simonovsky and Komodakis (2018) and Rigoni et al. (2020).[52,53] Values for the GVAE were obtained in this work. As a way to compare the effects of using SMILES of different lengths, we also report the results for the SGVAE model trained using the Gap energy (Δ_ε) property but with the SMILES from the DFT-B3LYP/6-31G-(2df,p) calculation instead, as reported in Table S1 on the Supporting Information. Table 4 shows the results. Best values are presented in bold. Even though the SGVAE presented high values for both novelty and uniqueness when trained using the SMILES from the DFT-B3LYP/6-31G-(2df,p) calculations, the overall best values for all metrics are achieved by graph-based algorithms, further showing their abilities to encode relevant information.

Except for the prior validity of the SGVAE trained on the U_0 property, all other values of such metric were higher in comparison to 5 out of 7 algorithms on Table 4, further showing the ability of the model to generate a high number of valid molecules. Notice that the best values for prior validity are all achieved by graph-based models. However, the number of novel and unique molecules were not equally high, reducing the relevance of the high validity values. Using the $\varepsilon_{\text{HOMO}}$ data from Table 3 as an example, from a total of 100 000 sampling trials, 62 040 (62.04 %) points were decoded into valid SMILES strings parsed by the RDKit, but only 10 075 (16.24 %) were novel and 1832 were unique. Performing the same calculations using the numbers for the GraphVAE model in Table 4, in a trial of 100 000 molecules, 55 700 (55.70 %) were valid molecules and, from this, 34 311 (61.60 %) were novel and 26 076 were unique, a number 14 times higher.

This result, by itself, is not a concerning problem for two reasons. First, it is simple and cheap to sample a new point from the prior and check its validity in a repeatedly manner. Thus, creating novel and unique molecules is just a matter of new trials. Second, as implied by Table S1 in the Supporting Information material, by just using the SMILES from relaxed geometries, the results on both novelty and uniqueness can be massively improved, generating ten times more unique molecules in comparison to the SGVAE model trained using the $\varepsilon_{\text{HOMO}}$ property and SMILES from the GDB-17 as discussed earlier. Moreover, the SGVAE model offers the possibility of accurately predict the ground-state properties of the newly generated molecules, also giving a sense of how incorporating property information to bias the training procedure can affect a model overall performance.

Finally, Figure 7 shows another possible sample strategy to generate molecules with a specific range of property values, but now using the two-dimensional PCA projection of the latent space and taking advantage of the fact that there is an approximate linear correlation between a point and a property value. Dashed black arrows show the locations and directions where the sampling of novel molecules occurred, while the colored arrows show the range of $\varepsilon_{\text{LUMO}}$ values over these locations. For each direction we sampled 100 points and decoded each point 100 times, as decoding is stochastic, and only kept the most common molecule. Figure 7 then shows 5 examples of novel molecules sampled on each direction with equal distance between each one of them. Their predicted $\varepsilon_{\text{LUMO}}$ values, in eV, are shown below each molecule. Notice that the molecules in the PCA 2 direction have similar $\varepsilon_{\text{LUMO}}$ values, as it would be expected according to the configuration of the latent space. On the other hand, and exploiting the linear mapping between the molecular latent representation and the property value, the molecules in the PCA 1 direction show a decreasing $\varepsilon_{\text{LUMO}}$ value the further right the sampling occurred. The red dot exhibit the intersection point between the dashed black arrows, and the highlighted molecule was the most common sample molecule after 100 trials in this particular location. The predicted $\varepsilon_{\text{LUMO}}$ value for this molecule is also shown.

3.2.2 PubChemQC Dataset

Table 5 shows the performance of the model measured by reconstruction accuracy, percentage of novel molecules and percentage of unique molecules when trained using the $\varepsilon_{\text{LUMO}}$ in comparison to the standard GVAE. Contrary to what was previously seem with the QM9 dataset, the metrics didn't presented much disparity between the models, both having overall high values of unique molecules but a poor performance in validity. The same behavior was observed for when the model was trained using the SMILES from the DFT-B3LYP/6-31G-(2df,p) relaxed geometries of the QM9 dataset. Larger SMILES sequences with a greater variety of characters to construct a valid sequence might be the reason for such a high variety of novel and unique molecules, and yet, it could represent a more difficult task for the model to correctly assemble a valid SMILES.

Table 5: Comparison between the metrics of reconstruction accuracy, prior validity, novel and unique molecules for the GVAE and the SGVAE trained using the $\varepsilon_{\text{LUMO}}$ property for the PubChemQC dataset. All values are in percentages

Model	Reconstruction	Prior validity	Novel	Unique
SGVAE	60.93	12.29	72.66	93.06
GVAE	57.48	10.98	69.22	97.76

As in the case of the QM9 dataset, we also analyzed the correspondence between the properties of newly generated molecules to the desired properties represented by a molecule with target property value mapped into the latent space. Once again, 50 random molecules from the hold-out set were mapped to the latent space and equation 2 was used to generate a total of 100 novel molecules in their neighborhood. Their $\varepsilon_{\text{LUMO}}$ was then predicted by the model and compared to the values of the seed molecules. Results once again show a reduction in the accuracy of predictions when comparing the target property value represented by the seed molecule and that of the novel molecules. The MAE grew to (0.326 ± 0.036) eV and the RMSE was (0.194 ± 0.046) eV.

High and low populated regions of the latent space were again responsible for the disparity between most of the MAE results depending on the seed molecule. For example, using the molecule with SMILES *CC(C)N(C(=O)Nc1ccc(F)cc1)c1ccccc1* and $\varepsilon_{\text{LUMO}}$ equals to -0.3172 eV as seed, the resulting MAE and RMSE were 0.067 eV and 0.008 eV respectively, considerably lower than the MAE for the hold-out set. As can be seen in Figure 8, which shows the distribution of $\varepsilon_{\text{LUMO}}$ in the portion of PubChemQC dataset used for training, the region comprising the aforementioned molecule contains the highest number of examples for which the model could be trained on. This explains such a low value for the MAE when using such seed molecule to generate novel molecules. On the other hand, when using the molecule with SMILES *CN(C)C(=S)N=c1ssc(=S)n1C* and $\varepsilon_{\text{LUMO}}$ equals to -2.1140 eV as seed, the MAE and RMSE grew to 0.817 eV and 0.833 eV respectively,

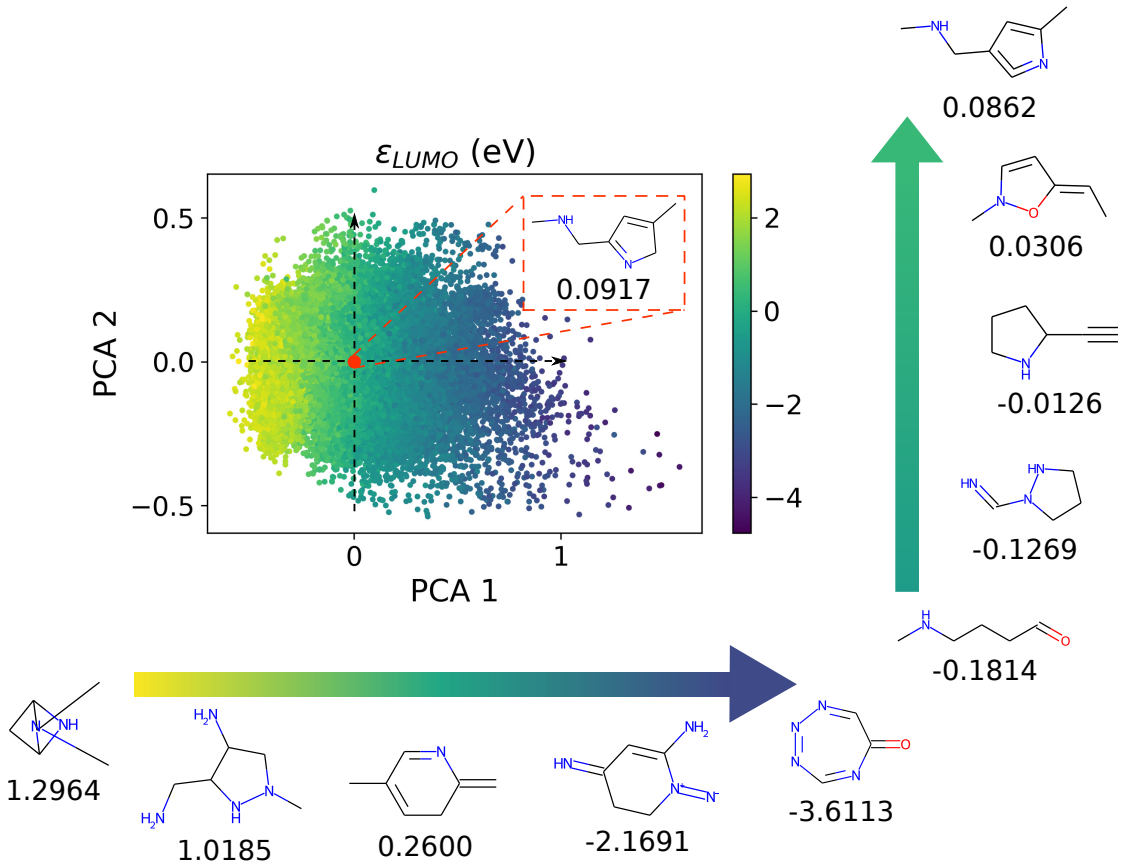

Figure 7: Novel molecules sampled in two particular directions over the latent space organized by the values of the $\varepsilon_{\text{LUMO}}$ property as shown in Figure 3. The dashed black arrows show the location where the sampling occurred and the colored arrows show the range of $\varepsilon_{\text{LUMO}}$ values over theses locations. All molecules are novel and their predicted $\varepsilon_{\text{LUMO}}$ value are shown below each molecule. The red dot shows the intersection between the two sampling directions and the highlighted molecule, alongside its predicted $\varepsilon_{\text{LUMO}}$ value, was the one most commonly sampled molecule in this point after 100 trials. All values are in eV

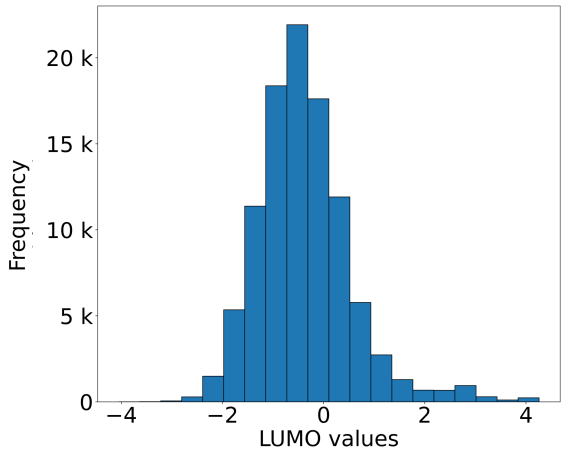

Figure 8: Distribution of the $\varepsilon_{\text{LUMO}}$ property in the training set for the PubChemQC dataset.

showing again the high uncertainty of the model when predicting the properties of molecules sampled from low populated regions in the latent space.

3.3 From Property Prediction to Molecular Design

Incorporating property information during the GVAE's training allowed the creation of a biased latent space which was used to do both, the accurate prediction of molecular properties from a hold-out set and the design of novel molecules with some particular property. Not only this, but one key aspect of such approach is that using the trained prediction and generative models it was also possible to estimate the properties of the newly generated molecules, as shown in Figures 6 and 7. Figure 7 also showed a possible sample strategy, where one could sample new molecules over a particular direction on the biased latent space starting from a specific molecule to either find molecules with increasing or decreasing values of a target property, or find molecules with similar property values.

Results on property prediction showed that

the Mean Absolute Error (MAE) between the true and the predicted property values from a hold-out set came close to chemical accuracy for the atomization energies and dipole moment, even outperforming models specifically designed for the prediction of molecular properties using the SMILES as molecular representation. As for the generative task, the biased latent space improved most of the common metrics employed to evaluate generative models, namely, reconstruction accuracy, prior validity and percentage of novel and unique molecules, in comparison to the standard GVAE and other generative models in the literature. However, we also showed that these metrics significantly varied depending on the property the model was trained on, indicating that one should be careful and investigate how impactful the use of property information would be in the final result to not hinder the ability the model in generating novel and valid molecules.

The length of the SMILES representation was another source of variance on the evaluation of the generative portion of the model, as shown in Table 4 and in the Supporting Information material. While using the QM9 dataset and its SMILES from the GDB-17 presented reasonable results across all generative modeling metrics, using the SMILES from the DFT-B3LYP/6-31G-(2df,p) calculations induced poor results on prior validity but it also presented by far the best results in percentage of novel and unique molecules, further showing how many factors one needs to consider when evaluating a model's performance. Nonetheless, the SGVAE model was indeed able to present good results in the two proposed tasks, showing the possibility of combine the molecular property prediction and molecular design of novel molecules under a single model.

4 Conclusions

We presented a modification in the GVAE's model that allowed the used of property information during its training to create a biased latent space representation which could then be used for molecular property prediction and molecular design of molecules with some desirable property. In order to do so, we added a FNN into the GVAE's architecture to predict the molecular properties of the input molecule based on its latent representation. The error of both networks, FNN and GVAE, was added together and the resulting latent space is organized by property value as shown in Figure 3.

Such framework has allowed the generation of new molecules with some desired property as it was exemplified in Figure 7, where the sampling of novel molecules can occur in any given direction over the latent space to either generate molecules with increasing or decreasing values of a particular property, or to generate molecules with similar property values. Results on the generative modeling metrics also indicated that such framework improved the amount of novel and unique molecules that can be generated by the SGVAE in comparison to other models in the literature, specially when using the SMILES from the DFT-B3LYP/6-31G-(2df,p) calculations as molecular representation when using the QM9 dataset.

The results also indicate a heavy influence of the molecular property on the configuration of the latent space itself, as all generative modeling metrics significantly varied depending on the property the model was trained on. Not only that, but we also observed that less populated regions over the latent space tended to yield worse results overall, as the model hadn't access to a sufficiently large amount of data points from those regions during training. Still, using the proposed approach we were able to accurately predict the properties values of the input molecules, generate new molecules with the desired property value and also estimate the property values for all sampled molecules over the latent space, whether they were generated from a specific point or randomly sampled. Thus, the SGVAE proved to be a viable way to unite two distinct applications of ML in the materials science field under a single framework by taking advantage of the molecular property information of the input molecules.

5 Data and Software Availability

The data and code that support the findings of this study are openly available at https://github.com/Monge88/SGVAE.

Acknowledgement The authors gratefully acknowledge support from FAPESP (São Paulo

Research Foundation) and Shell, projects No. 2017/11631 – 2 and 2018/21401 – 7, and the strategic importance of the support given by ANP (Brazil's National Oil, Natural Gas and Biofuels Agency) through the R&D levy regulation. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) – Finance Code 001.

Supporting Information Available

Complementary figures of the latent space configuration for all properties, the Pearson correlation coefficient between the true and predicted property values and novel random molecules sampled from the prior, together with their predicted properties for the QM9 dataset, are reported within the Supporting Information (SI). We also present the results of the SGVAE using the QM7-X dataset and briefly discuss the results of DFT calculations to access the reliability of the model in predicting accurate molecular properties. SI is available free, in the online version at http://www.rsc.org/

References

1 Liu, Y.; Zhao, T.; Ju, W.; Shi, S. Materials Discovery and Design Using Machine Learning. *J. Materiomics* **2017**, 159–177, DOI: 10.1016/j.jmat.2017.08.002.

2 Butler, K. T.; Davies, D. W.; Cartwright, H.; Isayev, O.; Walsh, A. Machine Learning for Molecular and Materials Science. *Nature* **2018**, *559*, 547–555, DOI: 10.1038/s41586-018-0337-2.

3 Kang, S.; Cho, K. Conditional Molecular Design With Deep Generative Models. *J. Chem. Inf. Model.* **2019**, *59*, 43–52, DOI: 10.1021/acs.jcim.8b00263.

4 Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: A Benchmark for Molecular Machine Learning. arXiv, 2018; 1703.00564.

5 LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444, DOI: 10.1038/nature14539.

6 Elton, D. C.; Boukouvalas, Z.; Fuge, M. D.; Chung, P. W. Deep Learning for Molecular Design — A Review of the State of the Art. *Mol. Syst. Des. Eng.* **2019**, *4*, 828–849, DOI: 10.1039/C9ME00039A.

7 Shen, J.; Nicolaou, C. A. Molecular Property Prediction: Recent Trends in the Era of Artificial Intelligence. *Drug Discovery Today: Technologies* **2019**, *32–33*, 29–36, DOI: 10.1016/j.ddtec.2020.05.001.

8 Lejaeghere, K.; Bihlmayer, G.; Bjorkman, T.; Blaha, P.; Blügel, S.; Blum, V.; Caliste, D.; Castelli, I. E.; Clark, S. J.; Dal Corso, A.; de Gironcoli, S.; Deutsch, T.; Dewhurst, J. K.; Di Marco, I.; Draxl, C.; Dułak, M.; Eriksson, O.; Flores-Livas, J. A.; Garrity, K. F.; Genovese, L.; Giannozzi, P.; Giantomassi, M.; Goedecker, S.; Gonze, X.; Grånäs, O.; Gross, E. K. U.; Gulans, A.; Gygi, F.; Hamann, D. R.; Hasnip, P. J.; Holzwarth, N. A. W.; Iuşan, D.; Jochym, D. B.; Jollet, F.; Jones, D.; Kresse, G.; Koepernik, K.; Küçükbenli, E.; Kvashnin, Y. O.; Locht, I. L. M.; Lubeck, S.; Marsman, M.; Marzari, N.; Nitzsche, U.; Nordström, L.; Ozaki, T.; Paulatto, L.; Pickard, C. J.; Poelmans, W.; Probert, M. I. J.; Refson, K.; Richter, M.; Rignanese, G.-M.; Saha, S.; Scheffler, M.; Schlipf, M.; Schwarz, K.; Sharma, S.; Tavazza, F.; Thunström, P.; Tkatchenko, A.; Torrent, M.; Vanderbilt, D.; van Setten, M. J.; Van Speybroeck, V.; Wills, J. M.; Yates, J. R.; Zhang, G.-X.; Cottenier, S. Reproducibility in Density Functional Theory Calculations of Solids. *Science* **2016**, *351*, aad3000, DOI: 10.1126/science.aad3000.

9 Faber, F. A.; Hutchison, L.; Huang, B.; Gilmer, J.; Schoenholz, S. S.; Dahl, G. E.; Vinyals, O.; Kearnes, S.; Riley, P. F.; Lilienfeld, O. A. v. Prediction Errors of Molecular Machine Learning Models Lower Than Hybrid DFT Error. *J. Chem. Theory Comput.* **2017**, *13*, 5255–5264, DOI: 10.1021/acs.jctc.7b00577.

10 Sinitskiy, A. V.; Pande, V. S. Deep Neural

Network Computes Electron Densities and Energies of a Large Set of Organic Molecules Faster than Density Functional Theory (DFT). arXiv, 2018; 1703.00564.

11 Elton, D. C.; Boukouvalas, Z.; Fuge, M. D.; Chung, P. W. Neural Message Passing for Quantum Chemistry. Proceedings of the 34th International Conference on Machine Learning. 2017; pp 1263–1272, DOI: 10.5555/3305381.3305512.

12 Unke, O. T.; Meuwly, M. PhysNet: A Neural Network for Predicting Energies, Forces, Dipole Moments, and Partial Charges. *J. Chem. Theory Comput.* **2019**, *15*, 3678–3693, DOI: 10.1021/acs.jctc.9b00181.

13 Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. Analyzing Learned Molecular Representations for Property Prediction. *Journal of Chemical Information and Modeling* **2019**, *59*, 3370–3388, DOI: 10.1021/acs.jcim.9b00237.

14 Pinheiro, G. A.; Mucelini, J.; Soares, M. D.; Prati, R. C.; F., S. J. L.; Quiles, M. G. Machine Learning Prediction of Nine Molecular Properties Based on the SMILES Representation of the QM9 Quantum-Chemistry Dataset. *J. Phys. Chem. A* **2020**, *124*, 9854–9866, DOI: 10.1021/acs.jpca.0c05969.

15 Shen, W. X.; Zeng, X.; Zhu, F.; Wang, Y. L.; Qin, C.; Tan, Y.; Jiang, Y. Y.; Chen, Y. Z. Out-of-the-box deep learning prediction of pharmaceutical properties by broadly learned knowledge-based molecular representation. *Nat Mach Intell* **2021**, *3*, 334–343, DOI: 10.1038/s42256-021-00301-6.

16 Chen, D.; Gao, K.; Nguyen, D. D.; Chen, X.; Jiang, Y.; Wei, G.-W.; Pan, F. Algebraic graph-assisted bidirectional transformers for molecular property prediction. *Nat Commun* **2021**, *12*, DOI: 10.1038/s41467-021-23720-w.

17 Kingma, D. P.; Welling, M. Auto-Encoding Variational Bayes. arXiv, 2014; 1703.00564.

18 Bombarelli, R. G.; Wei, J. N.; Duvenaud, D.; Lobato, J. M.; Lengeling, B. S.; Sheberla, D.; Iparraguirre, J. A.; Hirzel, T. D.; Adams, R. P.; Guzik, A. A. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Cent. Sci.* **2018**, *4*, 268–276, DOI: 10.1021/acs.jpca.0c05969.

19 Blaschke, T.; Olivecrona, M.; Engkvist, O.; Bajorath, J.; Chen, H. Application of Generative Autoencoder in De Novo Molecular Design. *Mol. Inform.* **2018**, *37*, 1700123, DOI: 10.1002/minf.201700123.

20 Harel, S.; Radinsky, K. Prototype-Based Compound Discovery Using Deep Generative Models. *Mol. Pharm.* **2018**, *15*, 4406–4416, DOI: 10.1021/acs.molpharmaceut.8b00474.

21 Dai, H.; Tian, Y.; Dai, B.; Skiena, S.; Song, L. Syntax-Directed Variational Autoencoder for Structured Data. arXiv, 2018; 1703.00564.

22 Bjerrum, E. J.; Threlfall, R. Molecular Generation with Recurrent Neural Networks (RNNs). arXiv, 2017; 1703.00564.

23 Popova, M.; Isayev, O.; Tropsha, A. Deep Reinforcement Learning for De Novo Drug Design. *Science Advances* **2018**, *4*, eaap7885, DOI: 10.1126/sciadv.aap7885.

24 Gupta, A.; Müller, A. T.; Huisman, B. J. H.; Fuchs, J. A.; Schneider, P.; Schneider, G. Generative Recurrent Networks for De Novo Drug Design. *Mol. Inform.* **2018**, *37*, 1700111, DOI: 10.1002/minf.201700111.

25 Zheng, S.; Yan, X.; Gu, Q.; Yang, Y.; Yunfei, D.; Yutong, L.; Xu, J. QBMG: Quasi-Biogenic Molecule Generator with Deep Recurrent Neural Network. *J. Cheminform.* **2019**, *11*, DOI: 10.1186/s13321-019-0328-9.

26 Brown, N.; Fiscato, M.; Segler, M. H. S.; Vacher, A. C. GuacaMol: Benchmarking Models for De Novo Molecular Design. *J. Chem. Inf. Model* **2019**, *59*, 1096–1108, DOI: 10.1021/acs.jcim.8b00839.

27 David, L.; Thakkar, A.; Engvist, O. Molecular Representations in AI-Driven Drug Discovery: A Review and Practical Guide. *J. Chem. Inf. Model* **2020**, *12*, 1–22, DOI: 10.1186/s13321-020-00460-5.

28 Kusner, M. J.; Paige, B.; Hernández-Lobato, J. M. Grammar Variational Autoencoder. arXiv, 2017; 1703.00564.

29 Weininger, D. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31–36, DOI: 10.1021/ci00057a005.

30 Kingma, D. P.; Rezende, D. J.; Mohamed, S.; Welling, M. Semi-Supervised Learning with Deep Generative Models. arXiv, 2014; 1406.5298.

31 Siddharth, N.; Paige, B.; van de Meent, J.-W.; Desmaison, A.; Goodman, N. D.; Kohli, P.; Wood, F.; Torr, P. H. S. Learning Disentangled Representations with Semi-Supervised Deep Generative Models. arXiv, 2017; 1706.00400.

32 Ramakrishnan, R. P.; Rupp, M.; von Lilienfeld, O. A. Quantum Chemistry Structures and Properties of 134 Kilo Molecules. *Scientific Data 1* **2014**, *1*, 140022, DOI: 10.1038/sdata.2014.22.

33 Nakata, M.; Shimazaki, T.; Hashimoto, M.; Maeda, T. PubChemQC PM6: Data Sets of 221 Million Molecules with Optimized Molecular Geometries and Electronic Properties. *Journal of Chemical Information and Modeling* **2020**, *60*, 5891–5899, DOI: 10.1021/acs.jcim.0c00740.

34 Johannes Hoja, L. M. S.; Ernst, B. G.; Vazquez-Mayagoitia, A.; Distasio, R. A.; Tkatchenko, A. QM7-X, a comprehensive dataset of quantum-mechanical properties spanning the chemical space of small organic molecules. *Nature* **2021**, *8*, DOI: 10.1038/s41597-021-00812-2.

35 de Azevedo, L. C.; Pinheiro, G. A.; Quiles, M. G.; silva, J. L. F. D.; Prati, R. C. Systematic Investigation of Error Distribution in Machine Learning Algorithms Applied to the Quantum-Chemistry QM9 Data Set Using the Bias and Variance Decomposition. *Journal of Chemical Information and Modeling* **2021**, *61*, 4210–4223, DOI: 10.1021/acs.jcim.1c00503.

36 Gokcan, H.; Isayev, O. Learning molecular potentials with neural networks. *WIREs Computational Molecular Science* **2021**, e1564, DOI: 10.1002/wcms.1564.

37 Stohr, M.; Sandonas, L. M.; Tkatchenko, A. Accurate Many-Body Repulsive Potentials for Density-Functional Tight Binding from Deep Tensor Neural Networks. *The Journal of Physical Chemistry Letters* **2020**, *11*, 6835–6843, DOI: 10.1021/acs.jpclett.0c01307.

38 Chen, C.; Ye, W.; Zheng, C.; Ong, S. P. Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chem. Mater.* **2019**, *31*, 3564–3572, DOI: 10.1021/acs.chemmater.9b01294.

39 Ruddigkeit, L.; Deursen, R. v.; Blum, L. C.; Reymond, J.-L. Enumeration of 166 Billion Organic Small Molecules in the Chemical Universe Database GDB-17. *J. Chem. Inf. Model.* **2012**, *52*, 2864–2875, DOI: 10.1021/ci300415d.

40 Landrum, G. RDKit: Open-Source Cheminformatics. http://www.rdkit.org/, 2012; Acessed: 2021-04-14.

41 Wang, Y.; Bryant, S. H.; Cheng, T.; Wang, J.; Gindulyte, A.; Shoemaker, B. A.; Thiessen, P. A.; He, S.; Zhang, J. PubChem BioAssay: 2017 update. *Nucleic Acids Research* **2016**, *45*, D955–D963, DOI: 10.1093/nar/gkw1118.

42 Hopcroft, J.; Ullman, J. *Introduction to Automata Theory, Languages and Computation*; Pearson, 1979.

43 Cho, K.; van Merrienboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. arXiv, 2014; 1703.00564.

44 Moriwaki, H.; Tian, Y.-S.; Kawashita, N.; Takagi, T. Mordred: A Molecular

Descriptor Calculator. *Journal of Cheminformatics* **2017**, *10*, DOI: 10.1186/s13321-018-0258-y.

45 Brown, N.; Fiscato, M.; Segler, M. H. S.; Vaucher, A. C. GuacaMol: Benchmarking Models for De Novo Molecular Design. *J. Chem. Inf. Model.* **2019**, *59*, 1096–1108, DOI: 10.1021/acs.jcim.8b00839.

46 Langer, M. F.; Goeßmann, A.; Rupp, M. Representations of Molecules and Materials for Interpolation of Quantum-Mechanical Simulations via Machine Learning. arXiv, 2021; 1703.00564.

47 Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S. P. Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *Chem. Mater.* **2019**, *31*, 3564–3572, DOI: 10.1021/acs.chemmater.9b01294.

48 Schütt, K. T.; Sauceda, H. E.; Kindermans, P. J.; Tkatchenko, A.; Müller, K. R. SchNet – A Deep Learning Architecture for Molecules and Materials. *J. Chem. Phys.* **2017**, *148*, 3564–3572, DOI: 10.1063/1.5019779.

49 Lim, J.; Ryu, S.; Kim, J. W. K. W. Y. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *Journal of Cheminformatics* **2018**, *10*, DOI: 10.1186/s13321-018-0286-7.

50 Jin, W.; Barzilay, R.; Jaakkola, T. Junction Tree Variational Autoencoder for Molecular Graph Generation. arXiv, 2019; 1703.00564.

51 Sterling, T.; Irwin, J. J. ZINC 15 - Ligand Discovery for Everyone. *J. Chem. Inf. Model.* **2015**, *55*, 2324–2337, DOI: 10.1021/acs.jcim.5b00559.

52 Simonovsky, M.; Komodakis, N. GraphVAE: Towards Generation of Small Graphs Using Variational Autoencoders. arXiv, 2018; 1703.00564.

53 Rigoni, D.; Navarin, N.; Sperduti, A. A Systematic Assessment of Deep Learning Models for Molecule Generation. arXiv, 2020; 2008.09168.

TOC Graphic