Article/Review

FragNet, a contrastive learning-based transformer model for clustering, interpreting, visualising and navigating chemical space

Aditya Divyakant Shrivastava [1] and Douglas B. Kell [2,3,4]*

[1] Dept of Computer Science and Engineering, Nirma University, Ahmedabad, India; 17bit014@nirmauni.ac.in
[2] Department of Biochemistry and Systems Biology, Institute of Systems, Molecular and Integrative Biology, University of Liverpool, Crown St, Liverpool L69 7ZB, UK; dbk@liv.ac.uk
[3] Novo Nordisk Foundation Centre for Biosustainability, Technical University of Denmark, Building 220, Kemitorvet, 2800 Kgs Lyngby, Denmark
[4] Mellizyme Ltd, Liverpool Science Park, IC1, 131 Mount Pleasant, Liverpool L3 5TF, UK
* Correspondence: dbk@liv.ac.uk

Abstract: The question of molecular similarity is core in cheminformatics, and is usually assessed via a *pairwise* comparison based on vectors of properties or molecular fingerprints. We recently exploited variational autoencoders to embed 6M molecules in a chemical space, such that their (Euclidean) distance within the latent space so formed could be assessed within the framework of the entire molecular set. However, the standard objective function used did not seek to manipulate the latent space so as to cluster the molecules based on any perceived similarity. Using a set of some 160,000 molecules of biological relevance, we here bring together three modern elements of deep learning to create a novel and disentangled latent space, viz transformers, contrastive learning, and an embedded autoencoder. The effective dimensionality of the latent space was varied such that clear separation of individual types of molecules could be observed within individual dimensions of the latent space. The capacity of the network was such that many dimensions were not populated at all. As before, we assessed the utility of the representation by comparing clozapine with its near neighbours, and did the same for various antibiotics related to flucloxacillin. Transformers, especially when as here coupled with contrastive learning, effectively provide one-shot learning, and lead to a successful and disentangled representation of molecular latent spaces that at once uses the entire training set in their construction while allowing 'similar' molecules to cluster together in an effective and interpretable way.

Keywords: Deep learning – artificial intelligence – generative methods – chemical space – neural networks – transformers – attention – cheminformatics.

1. Introduction

The relatively recent development and success of 'deep learning' methods involving 'large', artificial neural networks (e.g. [1,2]) has brought into focus a number of important features that can serve to improve them further, in particular with regard to the 'latent spaces' that they encode internally. One particular recognition is that the much greater availability of unlabelled than labelled (supervised learning) data can be exploited in the creation of such deep nets (whatever their architecture), for instance in variational autoencoders [3-7] or in transformers [8-10].

A second trend involves the recognition that the internal workings of deep nets can be rather opaque, and especially in medicine there is a desire for systems that explain precisely the features they are using in order to solve classification or regression problems. This is often referred to as 'explainable AI' [11-18]. The most obviously explainable networks are those in which individual

dimensions of the latent space more or less directly reflect or represent identifiable features of the inputs; in the case of images of faces, for example, this would occur when the value of a feature in one dimension varies smoothly with, and thus can be seen to represent, an input feature such as hair colour, the presence or type of spectacles, the presence or type of a moustache, and so on [19-22]. This is known as a disentangled representation (e.g. [23-34]). To this end, it is worth commenting that the ability to generate more or less realistic facial image structures using orthogonal features extracted from a database or collection of relevant objects that can be parametrized has been known for decades [35-38].

Given an initialisation, the objective function of a deep network necessarily determines the structure of its latent space. Typical variational autoencoders seek to minimise the evidence lower bound (ELBO) of the Kulback-Leibler divergence between the desired and calculated output distributions [39-42], although many other variants with different objective functions have been suggested (e.g. [41,43-50]). However, a third development is the recognition that training with such unlabelled data can also be used to optimise the (self-) organisation of the latent space itself. A particular objective of one kind of self-organisation is one in which individual inputs are used to create a structure in which similar input examples are also closer to each other in the latent space; this is commonly referred to as self-supervised [10,51-53] or contrastive [54-62] learning. In image processing this is often performed by augmenting training data with rotated or otherwise distorted versions of a given image, which then retain the same class membership or 'similarity' despite appearing very different [57,63-65]. Our interests here are in molecular similarity.

Molecular similarity

Molecular (as with any other kind of) similarity [66-68] is a somewhat elusive but, importantly, unsupervised concept in which we seek a metric to describe, in some sense, how closely related two entities are to each other from their structure or appearance alone. The set of all small molecules of possible interest for some purpose, subject to constraints such as commercial availability [69], synthetic accessibility [70], or 'drug-likeness' [71], is commonly referred to 'chemical space', and it is very large [72-88]. In cheminformatics the concept of similarity is widely used to prioritise the choice of molecules 'similar' to an initial molecule (usually a 'hit' with a given property or activity in an assay of interest) from this chemical space or by comparison with those in a database, on the grounds that 'similar' molecular structures tend to have 'similar' bioactivities [89].

The problem with this is that the usual range of typical metrics of similarity, whether using molecular fingerprints or vectors of the values of property descriptors, tend to give quite different values for the similarity of a given pair of molecules (e.g. [90]). In addition, and importantly, such pairwise evaluations are done individually, and their construction takes no account of the overall structure and population of the chemical space.

Deep learning for molecular similarity

In a recent paper [3], we have constructed a subset of chemical space using six million molecules taken from the ZINC database [91] (www.zincdocking.org/), employing a variational autoencoder to construct the latent space used to represent 2D chemical structures. A brief survey [3] implied that molecules near each other in this chemical space did indeed tend to exhibit evident and useful structural similarities, though no attempt was made there either to exploit contrastive learning or to assess degrees of similarity systematically. Thus, it is correspondingly unlikely that we had optimised the latent space from the points of view of either optimal feature extraction or explainability

The most obvious disentanglement for small molecules, that is equivalent to feature extraction in images, is surely the extraction of molecular fragments or substructures, that can then simply be 'bolted together' in different ways to create any other larger molecule(s). Thus, it is reasonable that a successful disentangled representation would involve the principled extraction of useful

substructures (or small molecules) taken from the molecules used in the training. In this case we have an additional advantage over those interested in image processing, because we have other effective means for assessing molecular similarity, and these do tend to work for molecules whose Tanimoto similarity (TS) is greater than about 0.8 or so [90]; such molecules can then be said to be similar, providing positive examples for contrastive learning (although in this case we use a different encoding strategy). Pairwise comparisons returning TS values lower than say 0.3 may similarly be considered to represent negative examples.

Nowadays, transformer architectures (e.g. [9,10,92-100]) are seen as the state of the art for deep learning of the type of present interest. As per the definition of the contrastive learning framework mentioned in [57,62], , we add an extra autoencoder in which the encoder behaves as a projection head. The output of the transformer encoder, which we regard as representations, is to be of a higher dimension. Consequently, it can still take a relatively large computational effort to compute the similarity between the representations. To this end, we add a simple encoder network that maps the representations to a lower dimensional latent space on which the contrastive loss is computationally easier to define. Then, to again convert the latent vector to the appropriate representations to feed into the transformer decoder network, we add a simple decoder network.

In sum, therefore, it seemed sensible to bring together both contrastive learning and transformer architectures so as to seek a latent space optimised for substructure or molecular fragment extraction. The purpose of the present paper is to describe our implementation of this. During the preparation of this paper, a related approach also appeared [101], but used graphs rather than a SMILES encoding of the structures.

2. Results

Figure 1 shows the basic architecture chosen, essentially as set down by [102]. It is based on [102] and is described in detail in Materials and Methods. Pseudocode for the algorithm used is given in Table 1.

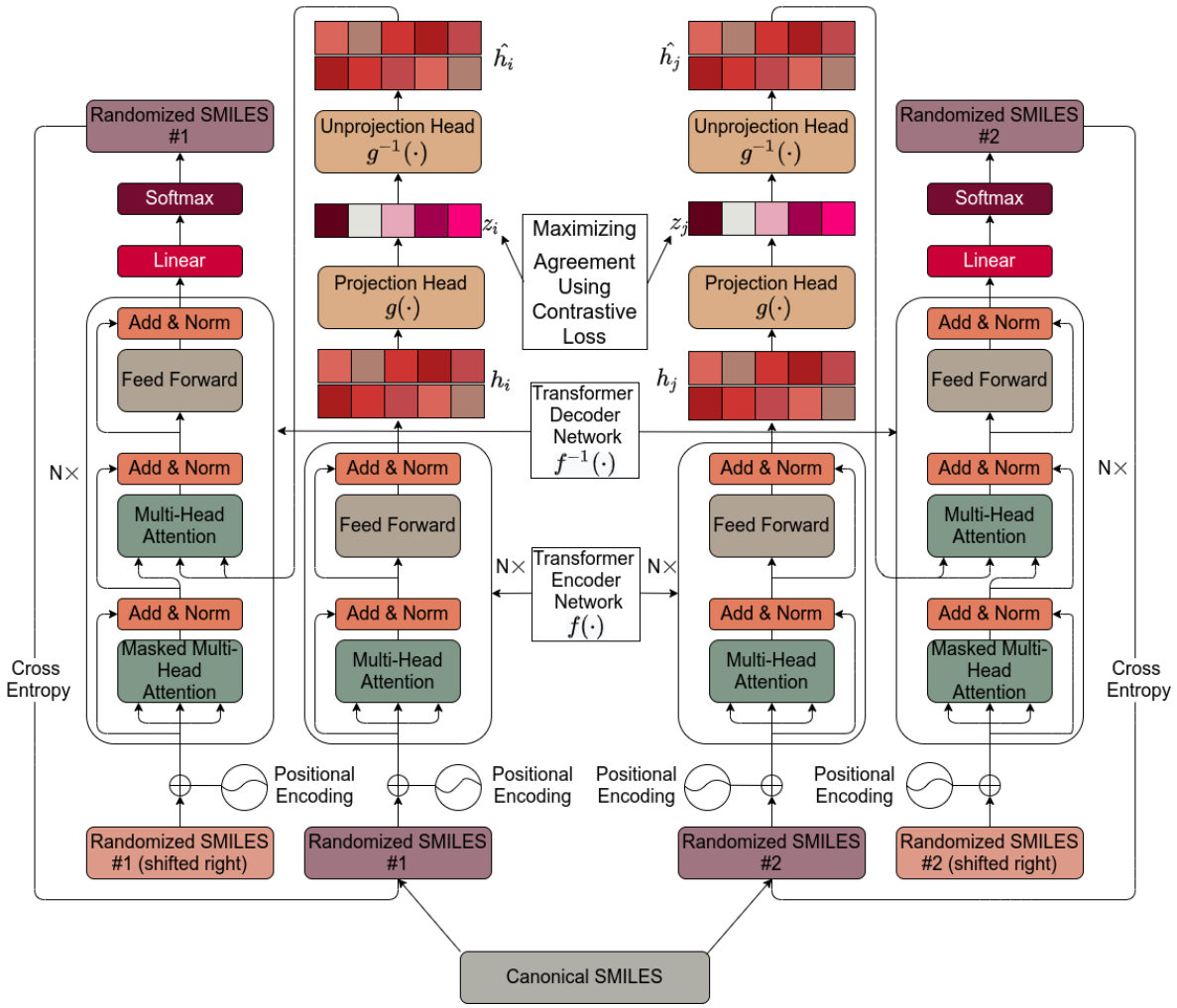


Fig 1. The transformer-based architecture used in the present work. The internals are described in Materials and Methods.

Table 1. Pseudocode for the transformer algorithm as implemented here

Input : Batch size N, constant τ, SMILES Enumeration Augmentation function T, transformer encoder network f, projection head g, projection head g^{-1}, transformer decoder network f^{-1}.

foreach *batch of samples* $\{x_k\}_{k=1}^{N}$ **do**
  foreach $k \in \{1, ..., N\}$ **do**
    `// For first augmentation`
    $\tilde{x}_{2k-1} = T(x_k)$
    $h_{2k-1} = f(\tilde{x}_{2k-1})$
    $z_{2k-1} = g(h_{2k-1})$
    $\hat{h}_{2k-1} = g^{-1}(z_{2k-1})$
    $\tilde{x}'_{2k-1} = f^{-1}(\hat{h}_{2k-1})$
    `// For second augmentation`
    $\tilde{x}_{2k} = T(x_k)$
    $h_{2k} = f(\tilde{x}_{2k})$
    $z_{2k} = g(h_{2k})$
    $\hat{h}_{2k} = g^{-1}(z_{2k})$
    $\tilde{x}'_{2k} = f^{-1}(\hat{h}_{2k})$
  Procedure `CrossEntropyLoss()`
    $L = -\sum_{k=1}^{N}[\tilde{x}_{2k-1}\log(\tilde{x}'_{2k-1}) + \tilde{x}_{2k}\log(\tilde{x}'_{2k})]$
    Update the networks f, g, g^{-1}, f^{-1} to minimize the above L.
  Procedure `ContrastiveLoss()`
    foreach $i, j \in \{1, ..., 2N\}$ **do**
      $\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_j) = z_i^{\top} z_j / (\| z_i \| \| z_j \|)$
      $\ell_{i,j} = -\log \frac{\exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{k \neq i} \exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}$
    $L = \frac{1}{2N}\sum_{k=1}^{2N}[\ell(2k-1, 2k) + \ell(2k, 2k-1)]$
    Update the networks f, g to minimize the above L.

Transformers are computationally demanding (our largest network had some 4.68M parameters), and so (as described in Materials and Methods) instead of using 6M ZINC molecules (that the memory available in our computational resources could not accommodate) we studied datasets consisting overall of ~160,000 natural products, fluorophores, endogenous metabolites and marketed drugs (the dataset is provided at [103]). We compared contrastive learning with the conventional objective function in which we used the Evidence Lower Bound of the K-L divergence. The first dataset (Materials & Methods) consisted of ~5,000 (actually 4,643) drugs, metabolites, fluorophores and 2,000 UNPD natural products molecules, while the second consisted of the full set of ~150k natural products. In appropriate circumstances, transformers can act as few-shot [98,104,105] or (as here) even one-shot learners [106,107]. We thus first compare the learning curves of transformers trained using cross entropy versus those trained using contrastive loss (Fig 2). In each case, the transformer-based learning essentially amounts to one-shot learning, especially for the contrastive case, and so the learning curve is given in terms of the effective fraction of the training set. We note that recent studies happily imply that large networks of the present type are indeed surprisingly resistant to overtraining [108]. In Fig 2A the optimal temperature used seemed to be 0.05 and this was

used for the larger dataset (Fig 2B). The clock time for training an epoch on a single NVIDIA-V100-GPU system was ca 30s and 23 min for the two datasets illustrated in Figs 2A and 2B, respectively.

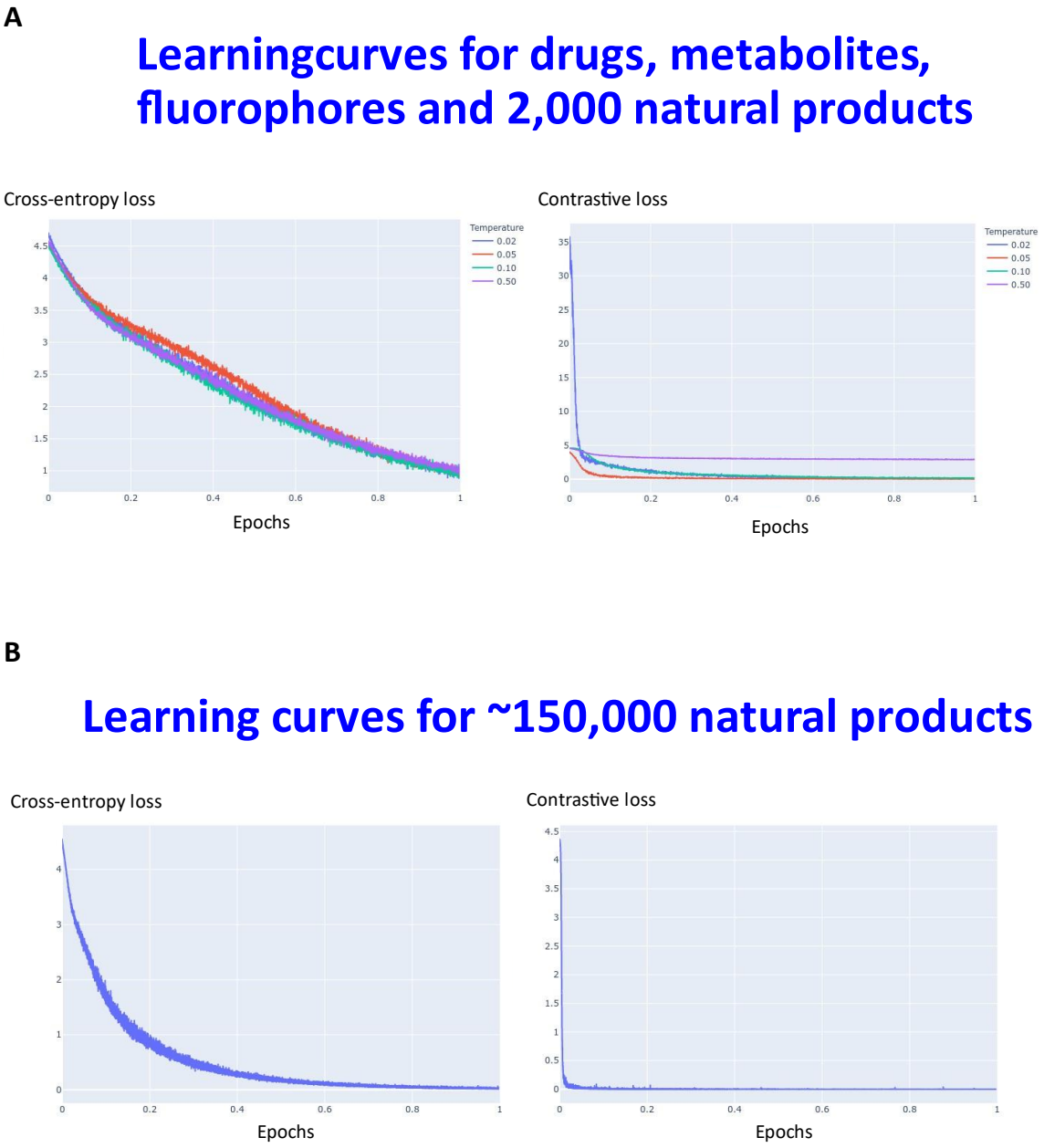
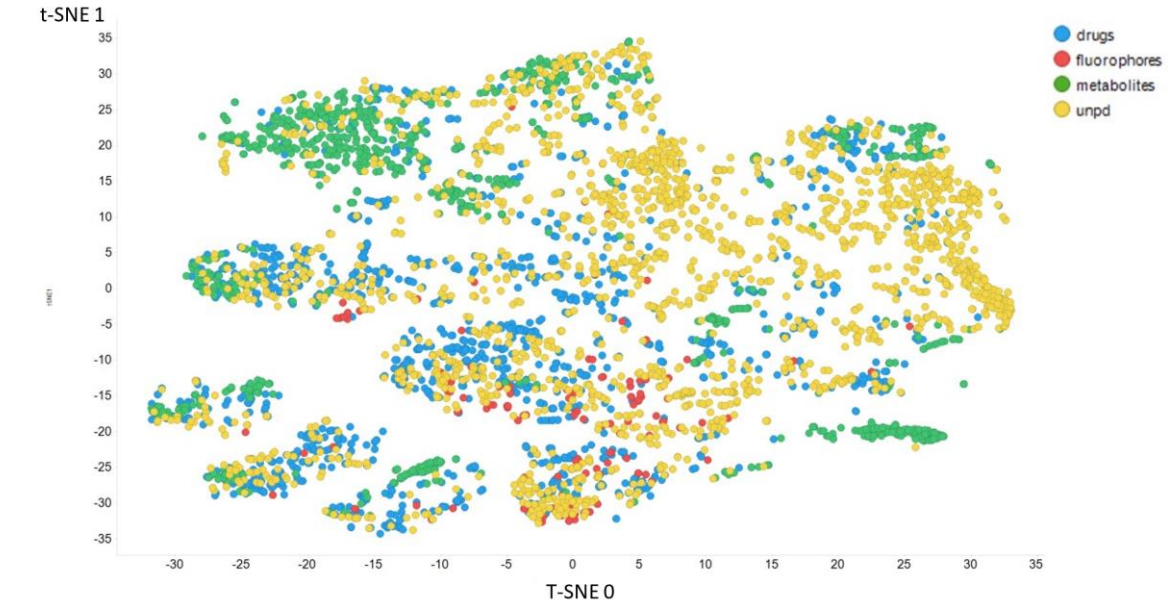


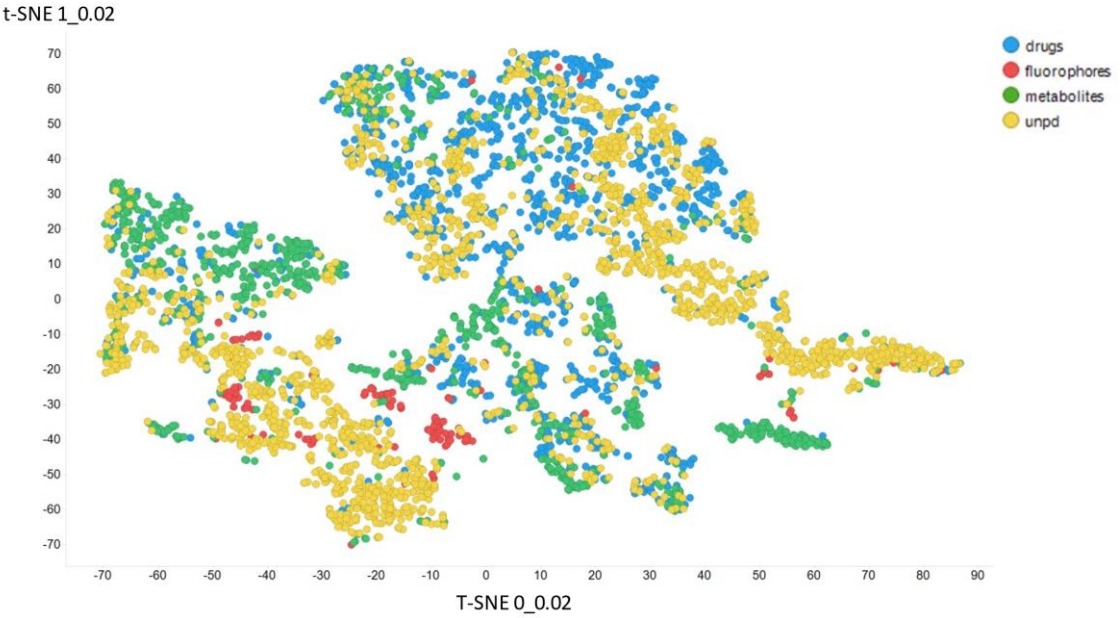
Fig 2. Learning curve for training our transformers on (A) drugs, metabolites, fluorophores and 2,000 natural products, (B) full set of natural products. Because the transformer is effectively a one-shot learner, and the batch size varied, the abscissa is shown as a single epoch. The batch size was varied as described in Materials and Methods, and was (A) 50 (latent space of 64 dimensions) and (B) 20 (latent space of 256 dimensions), leading to an actual number of batches of (A) 92 and (B) 7500.

Fig 3 gives an overall picture using t-SNE [109,110] of the dataset used. Fig 3A recapitulates that published previously, using standard VAE-type ELBO/K-L divergence learning alone, while panels B – E show the considerable effect of varying the temperature scalar (as in [102]).

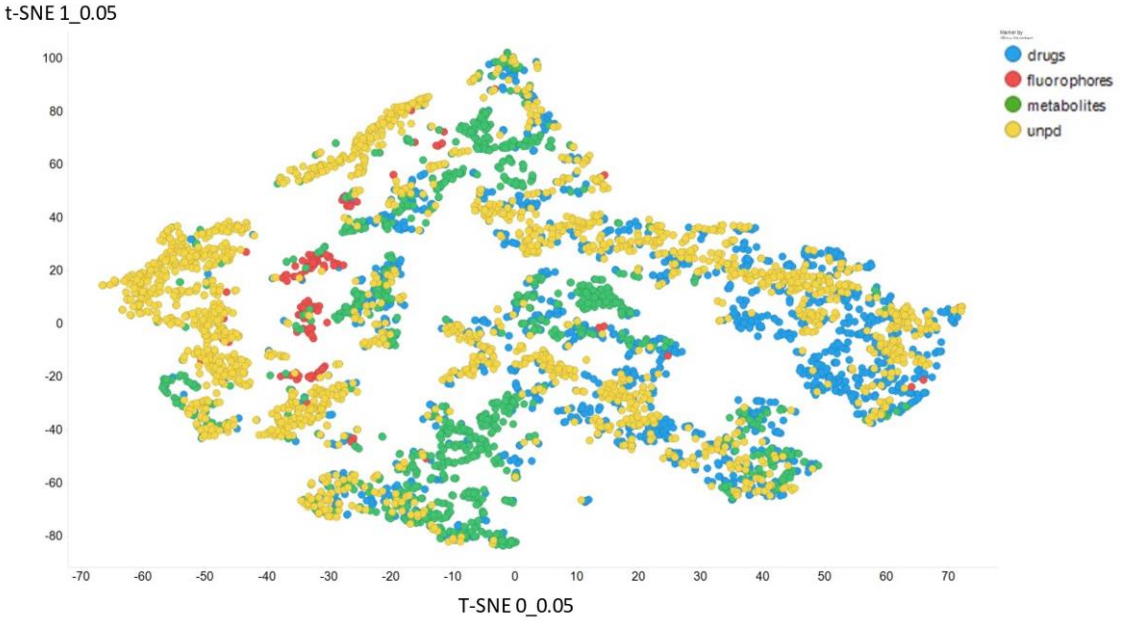
A



B



C



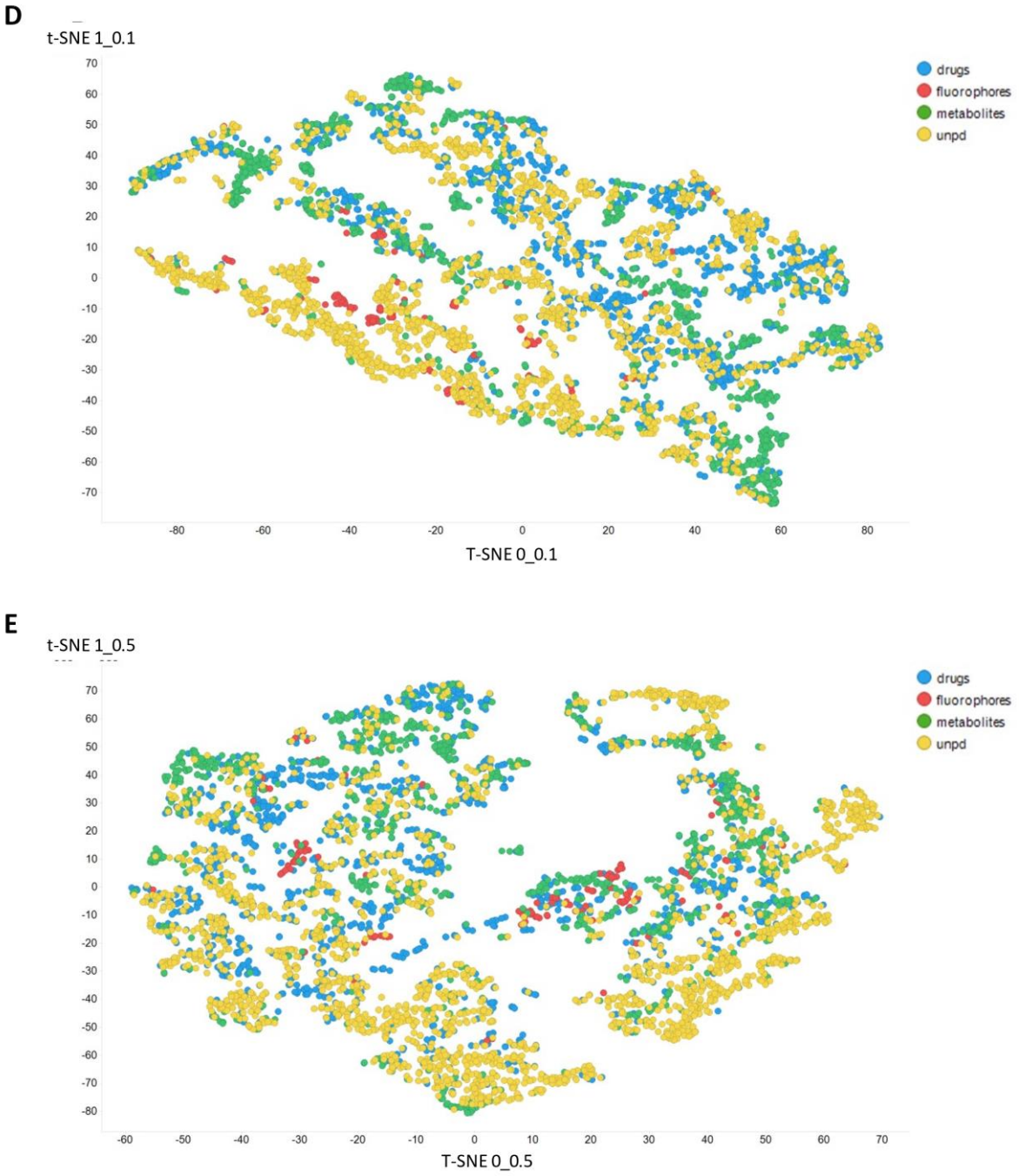


Figure 3. Effect of contrastive learning on the distribution of molecules in latent space as visualized via the t-SNE algorithm. For clarity, only a random subset of 2000 natural products is shown. (A) Learning based purely on the cross-entropy objective function. (B) - (E) The temperature scalar (as in [102]) was varied between 0.02 and 0.5 as indicated. All drugs, fluorophores and Recon2 metabolites are plotted, along with a randomly chosen 2000 natural products (as in [103]).

While there is no ideal metric for assessing similarity when considering it solely as an unsupervised problem, a value of 0.05 seemed to lead to the tightest clusters.

We also varied the number of dimensions used in the latent space, which served to provide some interesting insights into the effectiveness of the disentanglement and the capacity of the transformer (Fig 4).

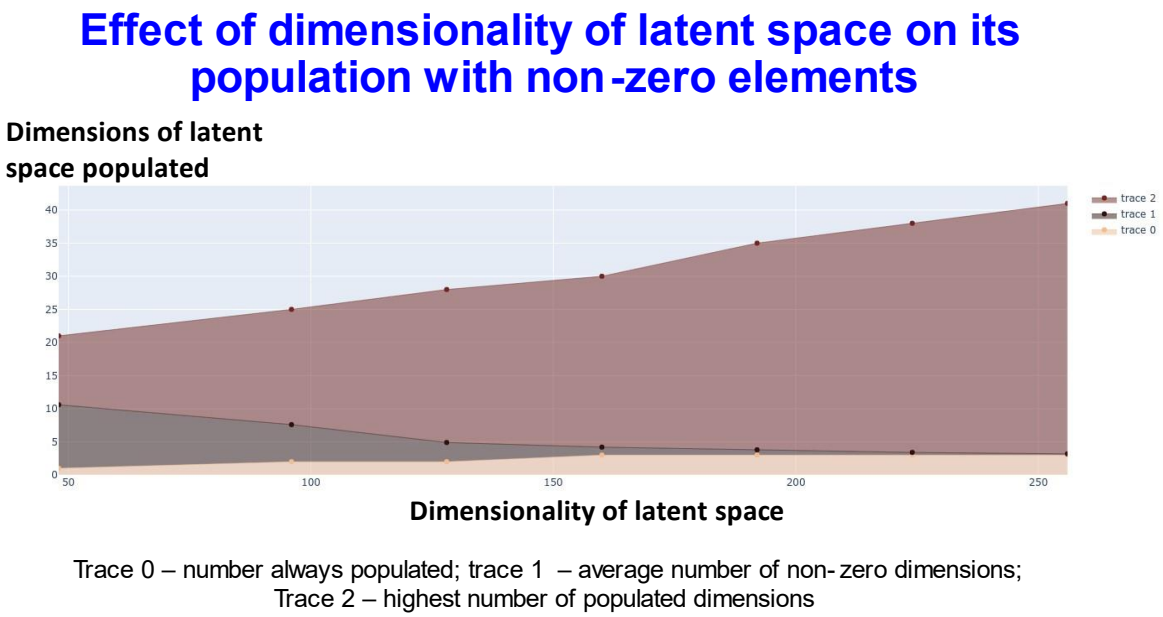


Fig 4. Relationship between the extent of population of different dimensions and the dimensionality of the latent space using transformers with contrastive learning.

In Fig 4, trace0 means that the elements of this number of dimensions was always nonzero. In other words, for every molecule, the value of at least that number of dimensions (the value on the y-axis) will be always non zero. Thus, for the 256-dimensional latent space 3 dimensions were always non-zero). trace1 means the average of the number of dimensions that were non zero for the dataset. Finally trace2 gives the highest number of dimensions recorded as populated for that specific dimensional latent space. This shows (and see below) that while GPU memory requirements meant that we were limited to a comparatively small number of molecules in our ability to train a batch of molecules, the capacity of the network was very far from being exceeded, and in many cases some of the dimensions were not populated with non-zero values at all. At one level this might be seen as obvious: if we have 256 dimensions *and each could take only two values*, there are 2^{256} positions in this space (~10^{77}). This large dimensionality at once explains the power and the storage capacity of large neural networks of this type.

We illustrate this further by showing the population of just three of the dimensions (for the 256-dimension case), viz dimensions 254 (Fig 5), 182 (Fig 6) and 25 (Fig 7) (these three were always populated with non-zero values).

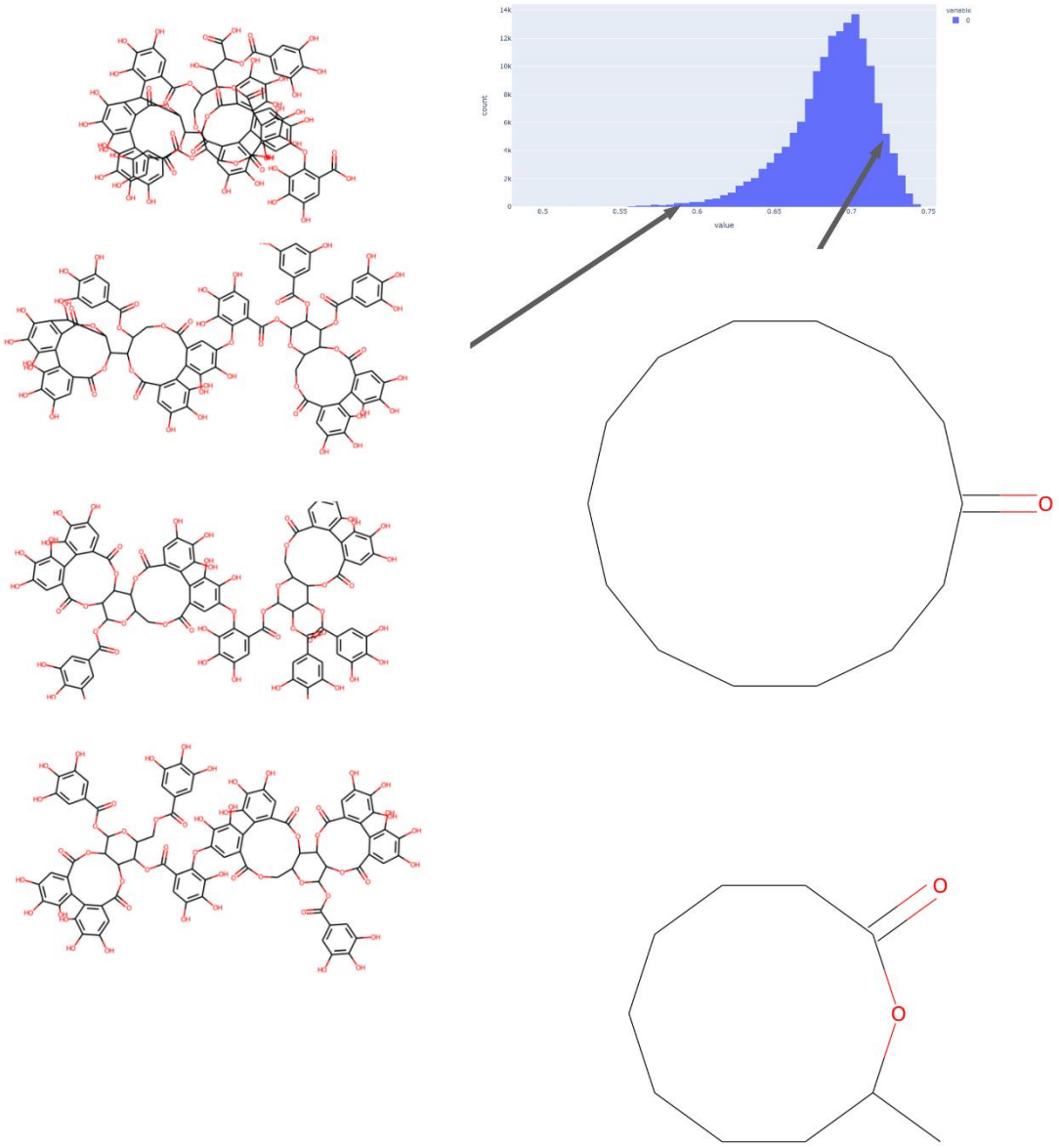


Fig 5. Values adopted in dimension 254 of the trained 256-D transformer, showing the values of various tri-hydroxy-benzene-containing compounds (left) ca 0.59 and two lactones (ca 0.73)

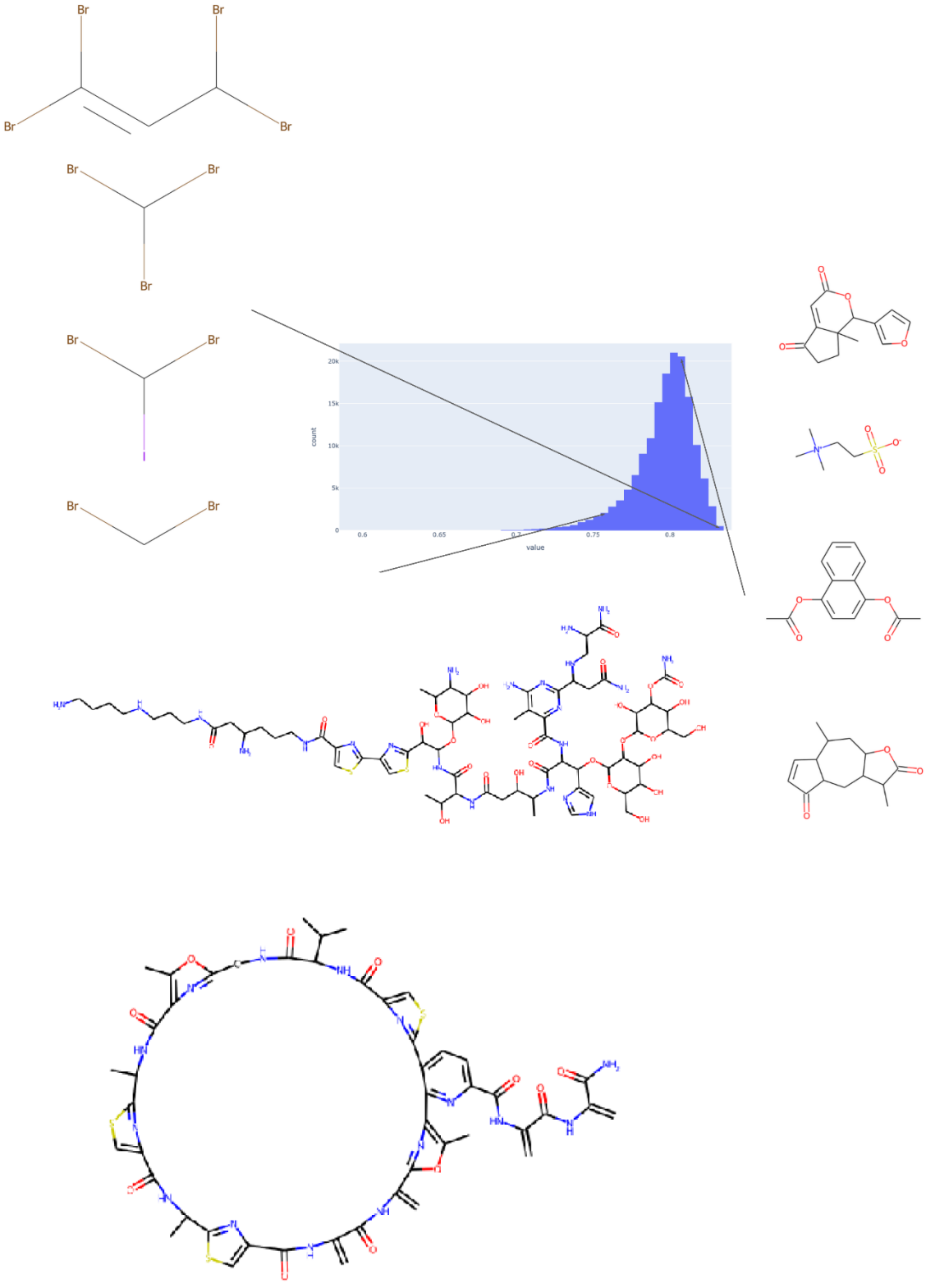
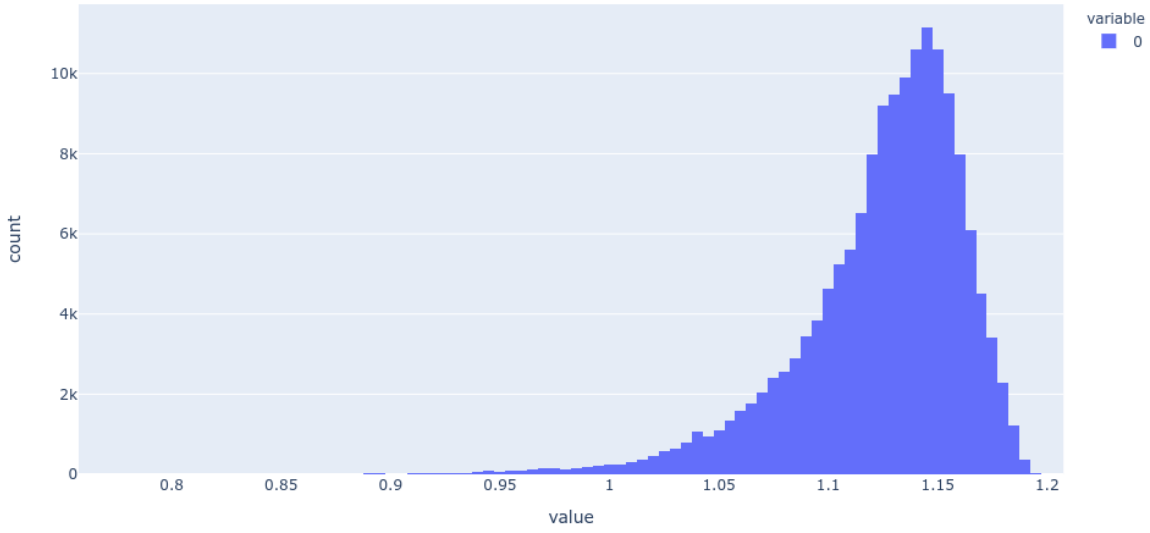


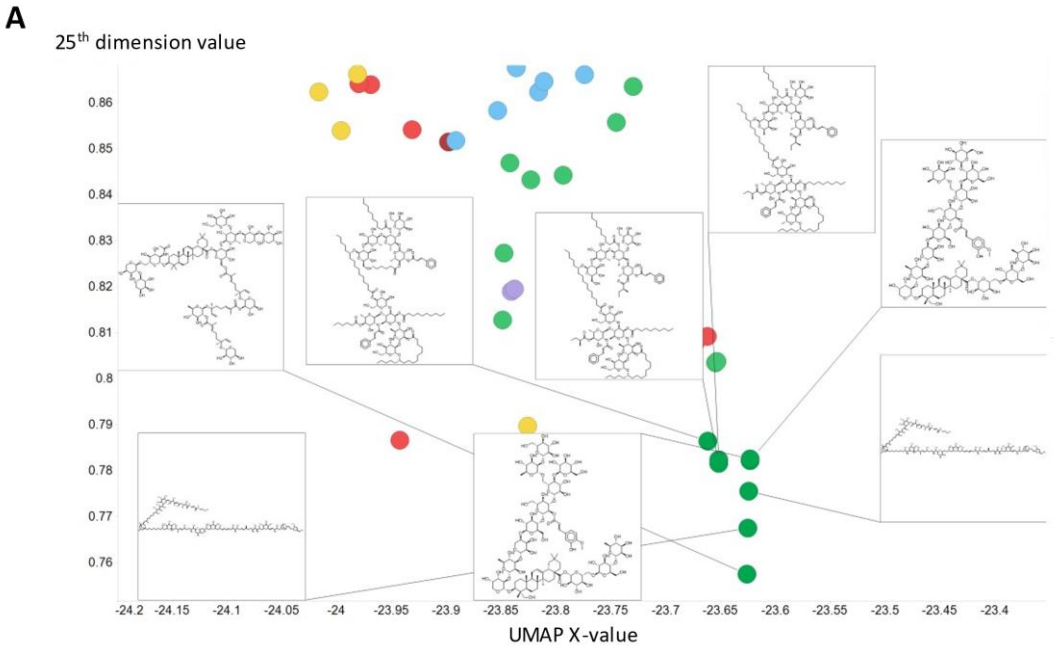
Fig 6. Values adopted in dimension 182 of the trained 256-D transformer, showing the values of various halides (~0.835) and other molecules



.

Fig 7. Histogram of the population of dimension 25 for the 256-D dataset. It is evident that most molecules adopt only a small range of non-zero values in this dimension.

To illustrate in more detail the effectiveness of the disentanglement, we illustrate a small fraction of the values of the 25th dimension alone, as plotted against a UMAP [111,112] X-coordinate. Despite the *tiny* part of the space involved (shown on the y-axis), it is clear that this dimension alone has extracted features that involve tri-hydroxylated cyclohexane Fig 8A) or halide-containing moieties (Fig 8B).



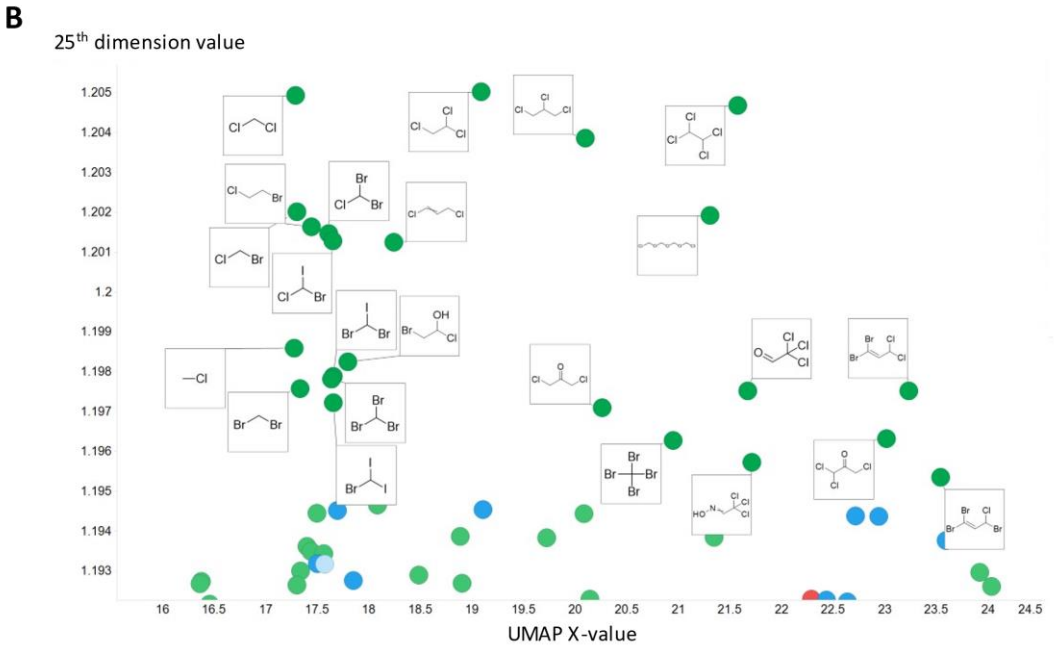


Fig 8. Effective disentangelement of molecular features into individual dimensions, using the indicated values of 25th dimension of the latent space of the 2nd dataset. In this case we used a latent space of 256 dimensions and a temperature τ of 0.05. A. trihydroxycyclohexane derivatives, B. halide-containing moieties.

Another feature of this kind of chemical similarity analysis involves picking a molecule of interest and assessing what is 'near' to it in the high-dimensional latent space, as judged by conventional measures of vector distance. We variously use the cosine or the Euclidean distance. As before [3], we choose clozapine as our first 'target' molecule, and use it to illustrate different feature of our method.

Fig 9 illustrates the relationship (using a temperature factor of 0.05) between the cosine similarity and the Tanimoto similarity for clozapine (using RDKit's RDKfingerprint encoding (https://www.rdkit.org/docs/source/rdkit.Chem.rdmolops.html),.

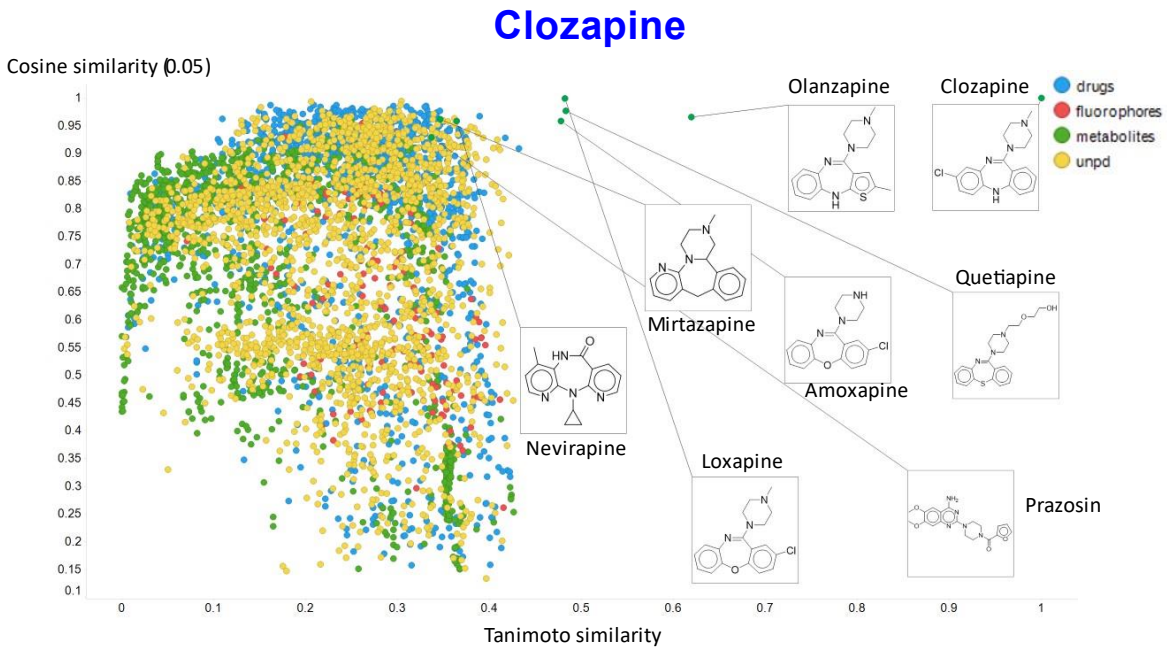


Fig 9. Relationship between cosine similarity and Tanimoto similarity for clozapine in our chemical space, using a temperature of 0.05.

It is clear that (i) very few molecules show up as being similar to clozapine in Tanimoto space, while prazosin (which competes with it for transport [113]) has a high cosine similarity despite having a very low Tanimoto similarity. In particular, none of the molecules with a high Tanimoto similarityhas a low cosine similarity, indicating that our method does recognise molecular similarities effectively.

To show other features, Fig 10 plots the cosine similarity against the Euclidean distance; they are tolerably well correlated, with an interesting bifurcation, implying that the cosine similarity is probably to be preferred.

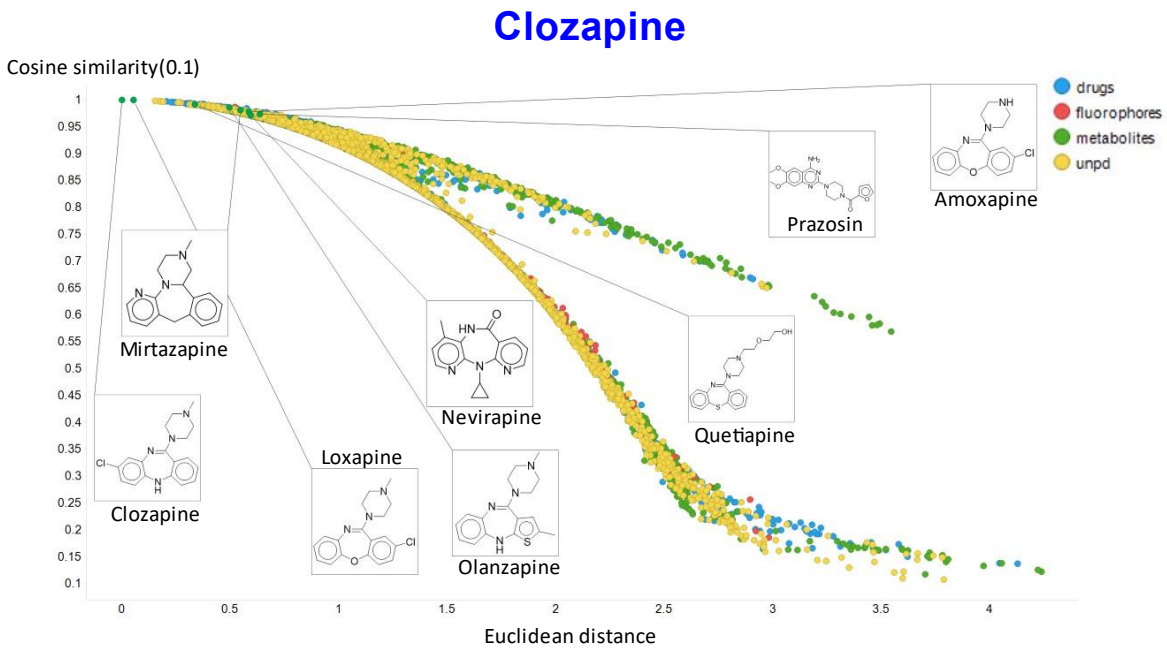


Fig 10. Relationship between cosine similarity and Euclidean disance for clozapine in our chemical space using a temperature of 0.1.

In a similar vein, varying the temperature scalar causes significant differences in the values of the cosine similarities for clozapine vs the rest of the dataset (Fig 11):

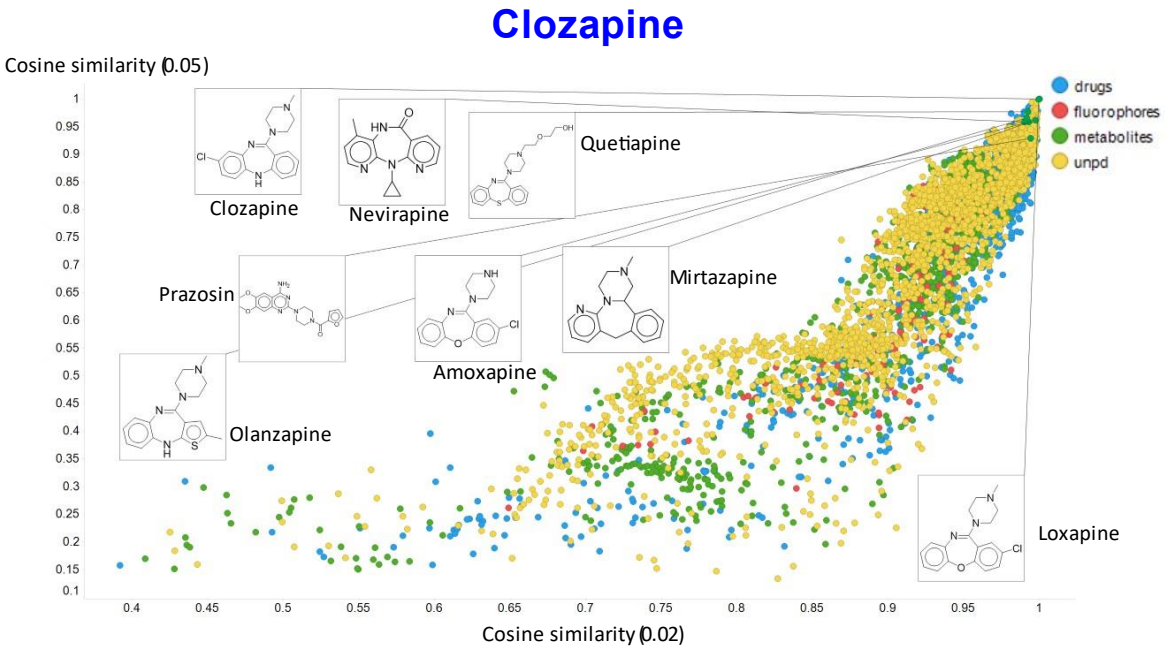


Fig 11. Relationship between cosine similarity for values of the temperature parameter of 0.05 and 0.02 for clozapine in our chemical space.

A similar plot is shown, at a higher resolution, for the cosine similarities with temperature scalars of 0.05 and 0.1 (Fig 12) and 0.05 vs 0.5 (Fig 13). The closeness of clozapine to the other 'apines' as judged by cosine similarity did vary somewhat with the value of the temperature. However, the latter value brings prazosin to be very close to clozapine, indicating the substantial effects that the choice of the temperature scalar can exert.

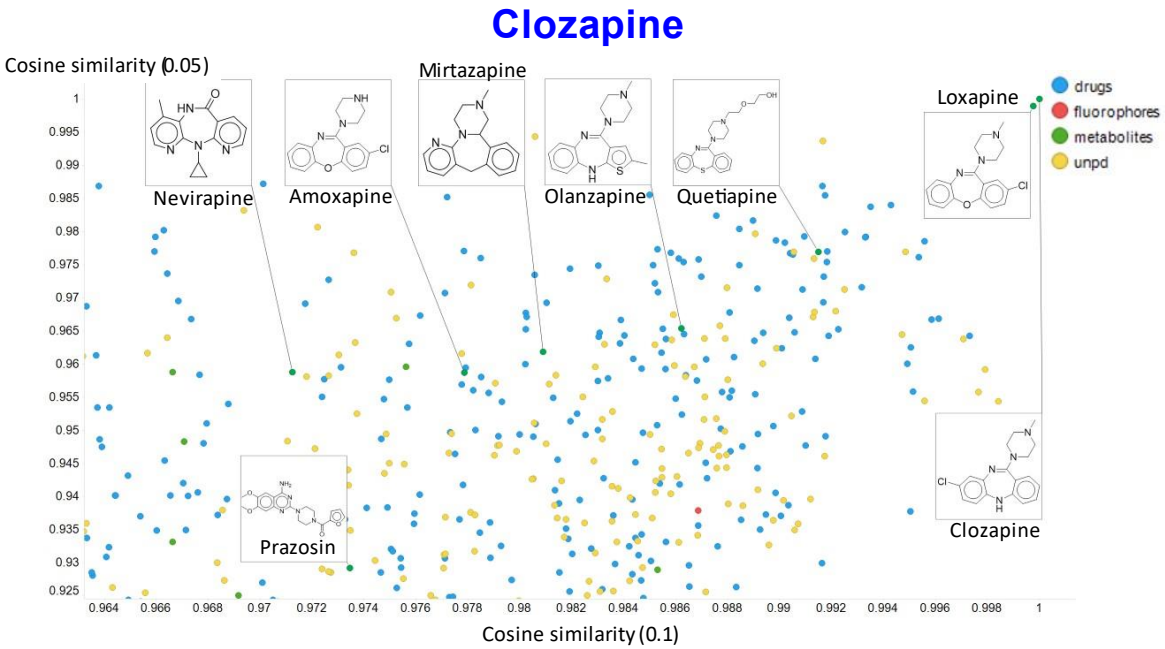


Fig 12. Relationship between cosine similarity for values of the temperature parameter of 0.05 and 0.1 for clozapine in our chemical space.

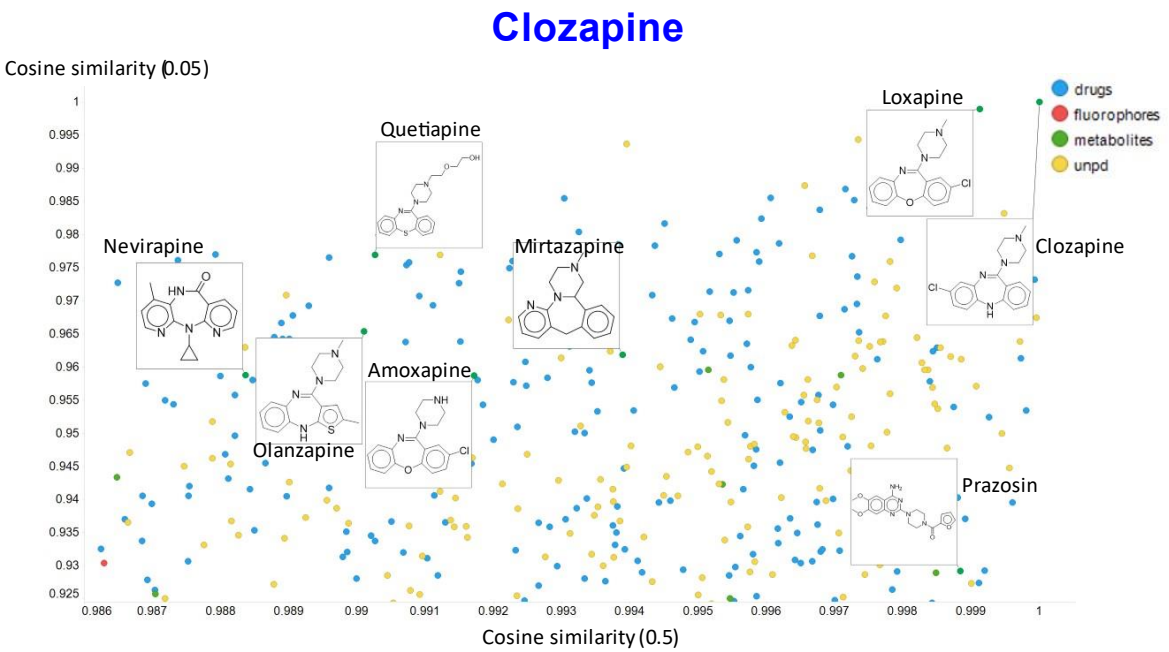


Fig 13. Relationship between cosine similarity for values of the temperature parameter of 0.05 and 0.5 for clozapine in our chemical space.

A similar exercise was undertaken for 'acillin'-type antibiotics based on flucloxacillin, with the results illustrated in Figures 14 to 18.

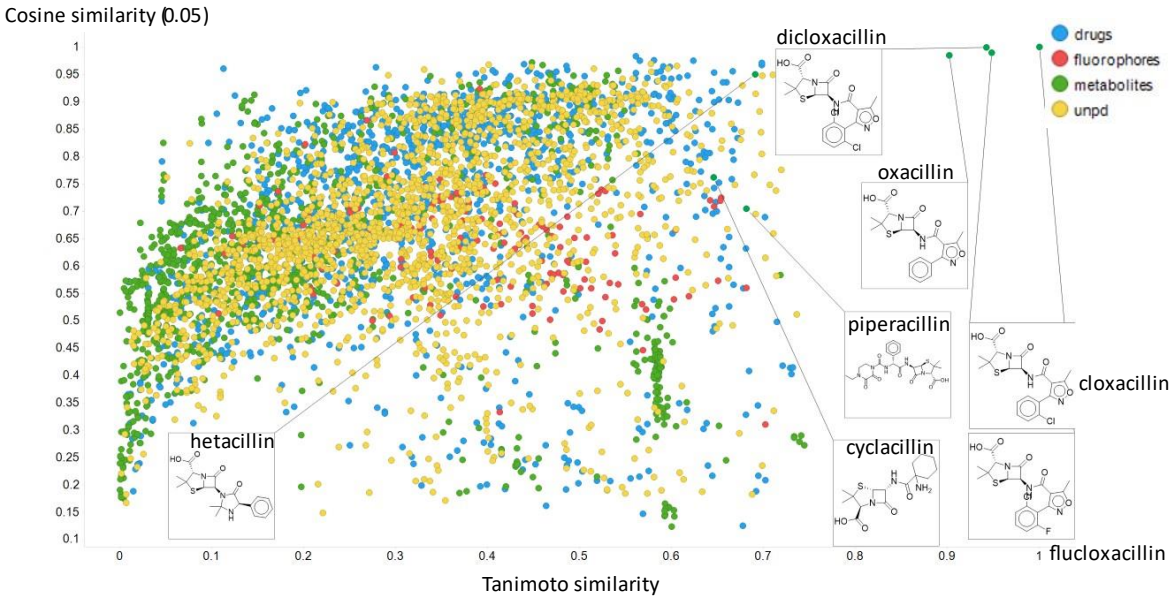


Fig 14. Relationship between cosine similarity and Tanimoto similarity (temperature = 0.05) for flucloxacillin in our chemical space.

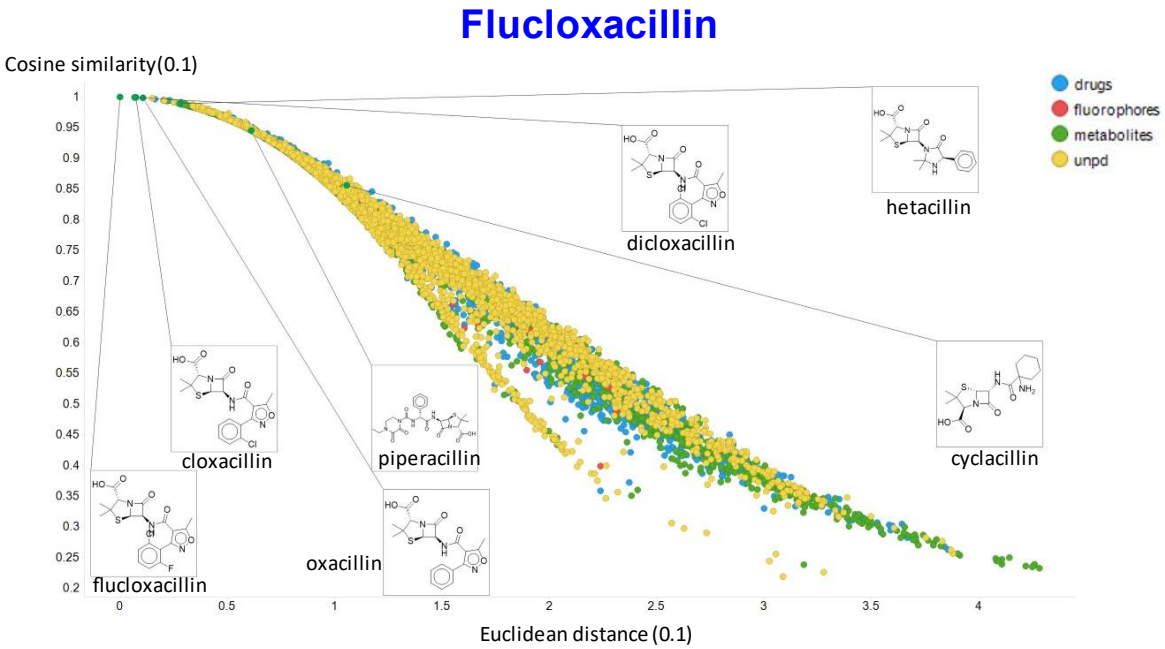


Fig 15. Relationship between cosine similarity and Euclidean distance for flucloxacillin in our chemical space, with a temperature parameter of 0.1.

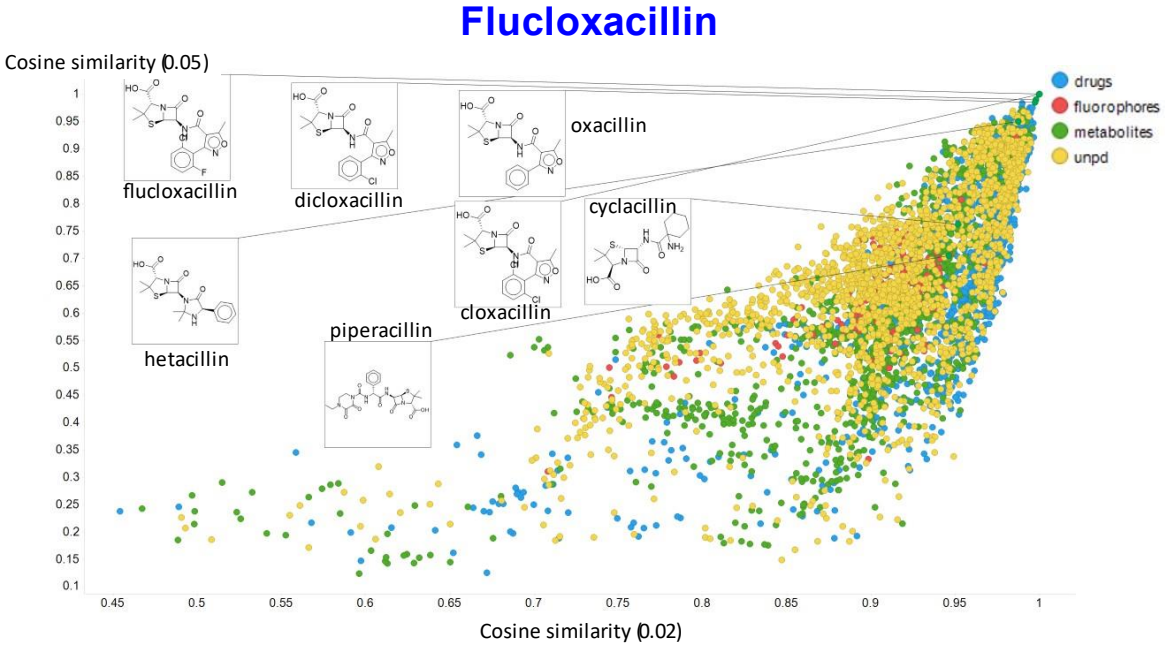


Fig 16. Relationship between cosine similarity for values of the temperature parameter of 0.05 and 0.02 for flucloxacillin in our chemical space.

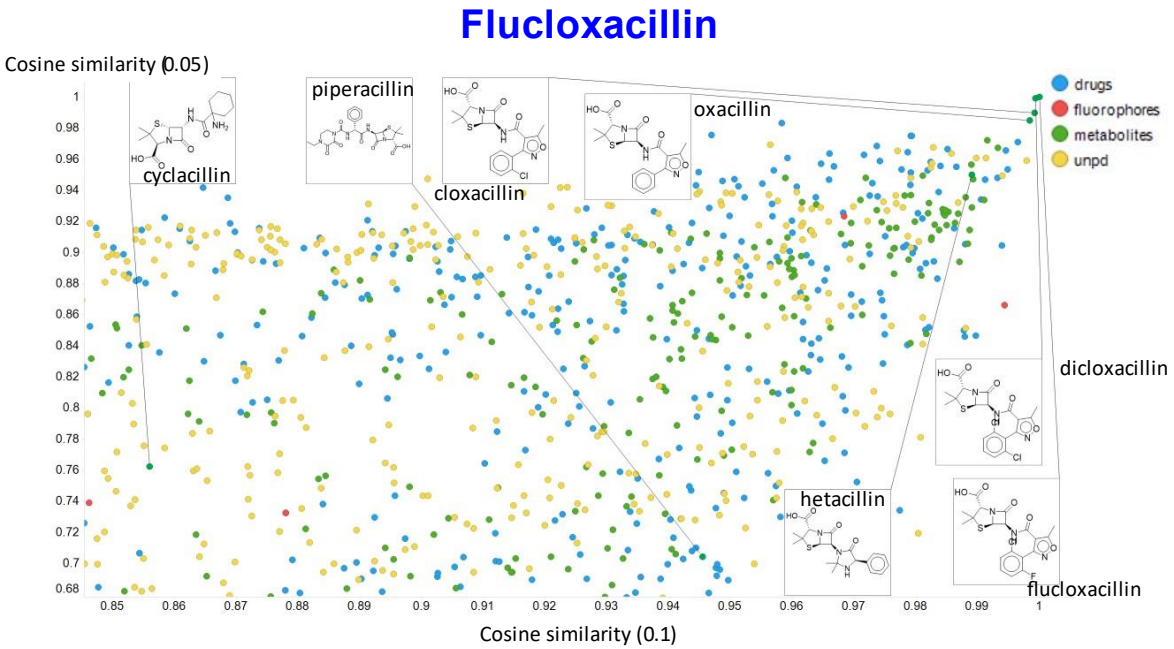


Fig 17. Relationship between cosine similarity for values of the temperature parameter of 0.05 and 0.1 for flucloxacillin in our chemical space.

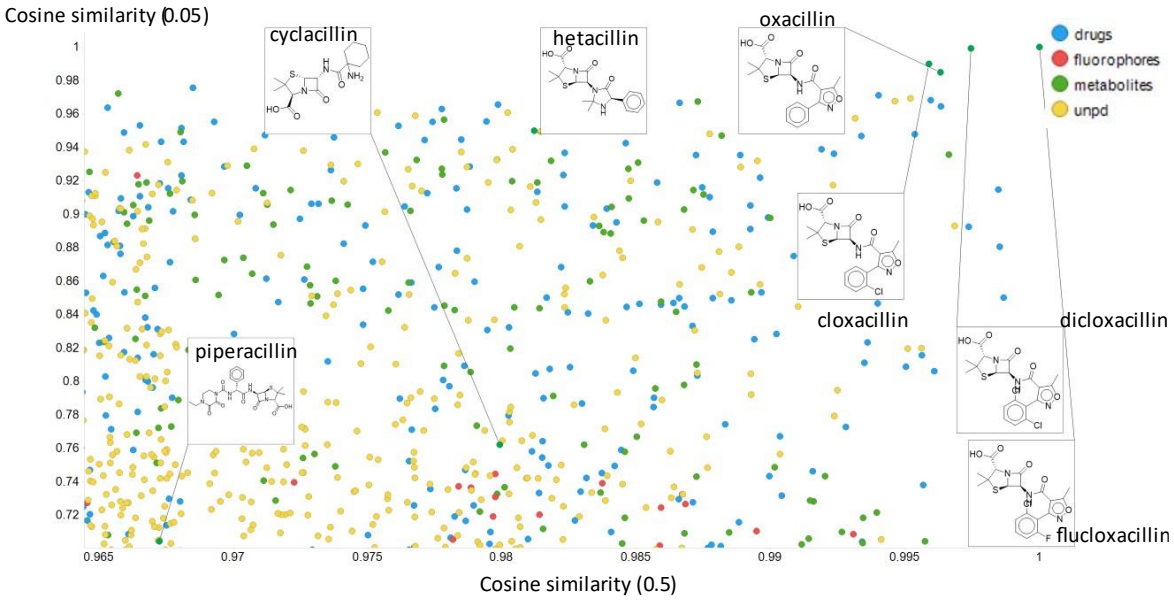


Fig 18. Relationship between cosine similarity for values of the temperature parameter of 0.05 and 0.5 for flucloxacillin in our chemical space.

In the case of flucloxacillin, the closeness of the other 'acillins' varied more or less monotonically with the value of the temperature parameter. Thus for particular drugs of interest it is likely best to fine tune the temperature parameter accordingly.

Finally, here, we show (using for clarity drugs and fluorophores only (Fig 19)), the closeness of chlorpromazine and prazosin in UMAP space when the NT-Xent temperature factor is 0.1.

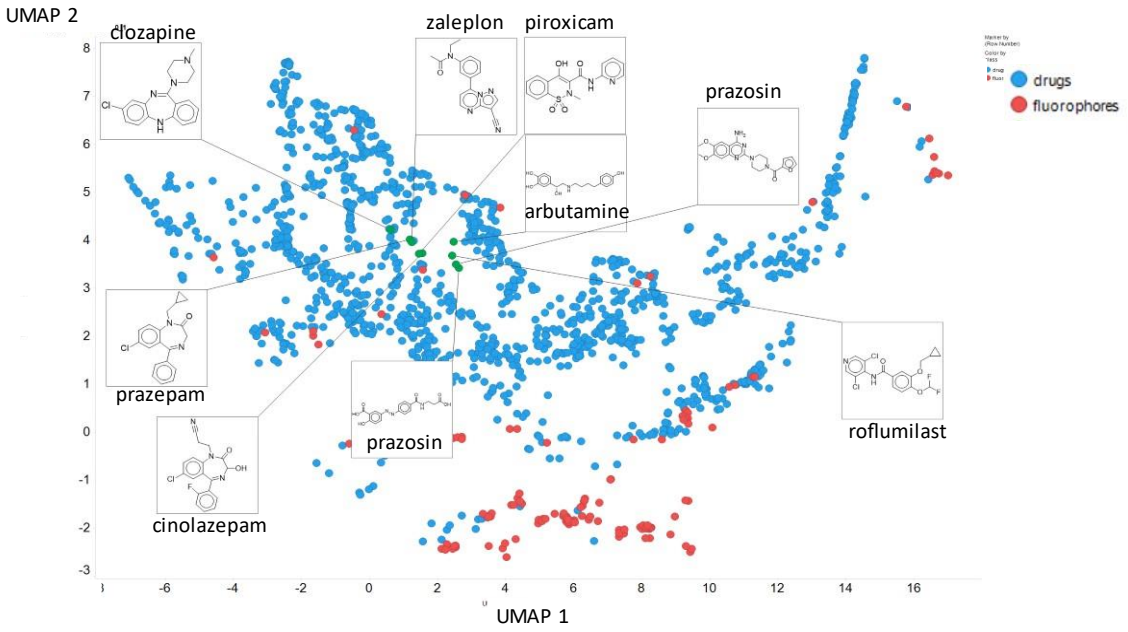


Fig 19. Positions of chlorpromazine, prazosin and some other molecules in UMAP space when the NT-Xent temperature factor is 0.1.

3. Discussion

The concept of molecular similarity is at the core of much of cheminformatics, on the simple grounds that structures that are more similar to each other tend to have more similar bioeffects, an elementary idea typically referred to as the 'molecular similarity principle' (e.g. [89,114-116]). Its particular importance commonly comes in circumstances where one has a 'hit' in a bioassay and wishes to select from a library of available molecules of known structure which ones to prioritise for further assays that might detect a more potent hit. The usual means of assessing molecular similarity are based on encoding the molecules as vectors of numbers based either on a list of measured or calculated biophysical or structural properties, or via the use of so-called molecular fingerprinting methods (e.g. [117-124]). We ourselves have used a variety of these methods in comparing the 'similarity' between marketed drugs, endogenous metabolites and vitamins, natural products and certain fluorophores [83,90,103,125-130].

At one level, the biggest problem with these kinds of methods is that all comparisons are done pairwise, and no attempt is thereby made to understand chemical space 'as a whole'. In a previous paper [3], based in part on other 'deep learning' strategies (e.g. [72,88,131-141]) we used a variational autoencoder (VAE) [4] to project some 6M molecules into a latent chemical space of some 192 dimensions. It was then possible to assess molecular similarity as a simple Euclidean distance.

A popular and more powerful alternative to the VAE is the transformer. Originally proposed by Vaswani and colleagues [9], transformers have come to dominate the list of preferred methods, especially those used with strings such as those involved in natural language processing [97,142]. Since chemical structures can be encoded as strings such as SMILES [143], it is clear that transformers might be used with success to attach problems involving small molecules, and they have indeed been so exploited (e.g. [8,10,95,144-147]). In the present work, we have adopted and refined the transformer architecture.

A second point is that in the previous work [3] we made no real attempt to manipulate the latent space so as to 'disentangle' the input representations, and if one is to begin to understand the working of such 'deep' neural networks it is necessary to do so. Of the various strategies available, those using

contrastive learning [9,58,62,148-150] seem to be the most apposite. In contrastive learning, one informs the learning algorithm whether two (or more) individual examples come from the same of different classes. Since in the present case we do know the structures, it is relatively straightforward to assign 'similarities', and we used a SMILES augmentation method for this.

The standard transformer does not have an obvious latent space of the type generated by autoencoders (variational or otherwise). However, the SimCLR architecture admits its production using one of the transformer heads. To this end, we added a simple autoencoder to our transformer such that we could create a latent space with which to assess molecular similarity more easily. In the present case, we used cosine similarity, Tanimoto similarity, and Euclidean distance.

There is no 'correct' answer for similarity methods, and as Everitt [151] points out, results are best assessed in relation to their utility. In this sense, it is clear that our method returns very sensible groupings of molecules that may be seen as similar by the trained chemical eye, and which in the cases illustrated (clozapine and flucloxacillin) clearly group molecules containing the base scaffold that contributes to both their activity and to their family membership ('apines' and 'acillins', respectively.

There has long been a general recognition (possibly as part of the search for 'artificial general intelligence' (e.g. [152-158]) that one reason that human brains are more powerful than are artificial neural networks may be – at least in part – simply because the former contain vastly more neurons. What is now definitely increasingly clear is that very large transformer networks can both act as few-shot learners (e.g. [98,100]) and are indeed able to demonstrate extremely powerful generative properties, albeit within somewhat restricted domains. Even though the limitations on the GPU memory that we could access meant that we studied only some 160,000 molecules, our analysis of the contents of the largest transformer trained with contrastive learning indicated that it was nonetheless very sparsely populated. This both illustrates the capacity of these large networks and leads necessarily to an extremely efficient means of training.

Looking to the future, as more computational resources become available (with transformers using larger networks for their function), we can anticipate the ability to address and segment much larger chemical spaces, and to use our disentangled transformer-based representation for the encoding of molecular structures for a variety of both supervised and unsupervised problem domains.

4. Materials and Methods

We develop a novel hybrid framework by combining three things namely: transformers, an auto-encoder, and a contrastive learning framework. The complete framework is shown in Fig 1. The architecture chosen is based on the SimCLR framework of Hinton and colleagues [57,102], to which we have added an autoencoder so as to provide a convenient latent space for analysis and extraction. Programs were written in PyTorch within an Anaconda environment. They were mostly run one GPU of a 4-GPU (NVIDIA V100) system. The dataset used included ~150,000 natural products [83,90,130], plus fluorophores [103], Recon2 endogenous human metabolites [125,126,128,129], and FDA-approved drugs [90,125-127], as previously described. Visualisation tools such as t-SNE [109,110] and UMAP [111,112] were implemented as previously described [103]. The dataset was split into training and validation and test sets as described below.

We here develop a novel hybrid framework upon the contrastive learning framework using transformers. We explain the complete framework with each of the components as below:

Molecular SMILES Augmentation

Contrastive learning is all about uniting positive pairs and discriminating between negative pairs. The first objective is thus to develop an efficient way of determining positive and negative data

pairs for the model. We adopt the SMILES Enumeration data augmentation technique from Bjerrum [159] that any given Canonical SMILES data example can generate multiple SMILES strings that basically represent the same molecule. We use this technique to sample two different SMILES strings x_i and x_j from every Canonical SMILES string from the dataset which we regard as positive pairs.

Base Encoder

Once we have received the augmented, randomised SMILES, they are added with their respective positional encoding. The positional encoding is a sine or cosine function defined according to the position of a token in the input sequence length. It is done in order also to take into consideration the order of the sequence. The next component of the framework is the Encoder network that takes in the summation of the input sequence and its positional encoding, and extracts the representation vectors for those samples. As stated by Chen and colleagues [102], there is complete freedom when it comes to the choice of architecture for the encoder network. Therefore, we used a Transformer Encoder Network which has in recent years become the state-of-the-art for language modelling tasks and has been subsequently significantly extended to chemical domains as well.

As set down in the original transformers paper, the transformer encoder basically comprises two sub-blocks. The first sub-block has a Multi-Head Attention Layer followed by a Layer Normalization layer. The first multi-head attention layer makes the model pay attention to the values at neighbours' positions when encoding a representation for one particular position. Then, the Layer Normalization layer normalizes the sum of inputs obtained from the residual connection and the outputs of the multi-head attention layer. The second block consists of a feed forward network, one for every position. Then, similar to the previous case, Layer Norm is defined on the position-wise sum of the outputs from the feed forward layer and the residually connected output from the previous block.

The output of the Transformer Encoder Network is an array of feature-embedding vectors which we call the representation (h_i). The representation obtained from the network is of the dimension-Sequential Length x d_{model} . This means that the transformer encoder network generates feature embedding vectors for every position in the input sequence length. Normally, these transformer encoder network blocks are repeated N times and the output representation of one encoder is an input of another. Here, we employ 4 transformer encoder blocks.

Projection Head

The projection head is a simple encoder neural network to project the feature embedding representation vector of shape (Input Sequence Length x d_{model}) down to a lower dimension representation of shape (1x d_{model}). Here, we use an artificial neural network of 4 layers with the ReLu activation function. This gives an output projection vector z_i which is then used for defining the contrastive loss.

Contrastive Loss

As the choice of contrastive loss for our experiments, we use the normalized temperature-scaled cross entropy (NT-Xent) loss [60,102,160,161].

$$\mathcal{L}_{i,j} = \log \frac{\exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}\{k \neq i\} \exp(\mathrm{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}, \quad \text{.....(1)}$$

where, z_i and z_j are positive pair projection vectors when two transformer models are run in parallel. $\mathbb{1}\{k \neq i\}$ is a Boolean evaluating to 1 if k is not the same as I, and τ is the temperature

parameter. Lastly, sim() is the similarity metric for estimating the similarity between z_i and z_j. The idea behind using this loss function is that when sampling a sample batch of data of size N for training, each sample is augmented as per subsection "Molecular SMILES Augmentation" and the total would then be 2N samples. Therefore, for every sample there is one other sample from the same Canonical SMILES and 2N-2 other samples. Therefore, we consider for every sample one other sample generated from the same canonical SMILES as a positive pair and each of the other 2N-2 samples as a negative pair.

Unprojection Head

Unlike SimCLR or any other previous contrastive learning framework, we also opt to include a simple decoder network and then a transformer decoder network through which we also teach the model to generate a Molecular SMILES representation whenever queried with latent space vectors. With this architecture, we thus develop a novel framework which can not only build nicely clustered latent spaces based on the structural similarities of molecules but also has the capability of doing some intelligent navigation of those latent spaces to generate some other highly similar molecules.

Base Decoder

This final component of our architecture, the base decoder, consists of a transformer decoder network, a final linear layer, and a softmax layer. The transformer decoder network adds one more block of multi-head attention which takes in the attention vectors K and V from the output of the unprojection. Moreover, the masking mechanism is infused in the first attention block to mask the 1 position shifted right output embedding. With this, the model is only allowed to take into consideration the feature embeddings from the previous positions. Then the final linear layer is a simple neural network to convert position vector outputs from the transformer decoder network into a logit vector which is then followed by softmax layer to convert this array of logit values into a probability score, and the atom or bond corresponding to the index with highest probability is produced as an output. Once the complete sequence of molecules is generated, it is compared with the original input sequence with cross-entropy as a loss function.

Default Settings

We refer to the first dataset of ~5k molecules containing natural products, drugs, fluorophores and metabolites as SI1 and that of ~150k natural product molecules as SI2.

For both the datasets, the choice of optimizer is Adam [162], the learning rate is 10^{-5} and dropout [163] is 20%. Our model has 4 encoder and decoder blocks and each transformer block has 4 attention heads in its multi-head attention layer. For the SI1 dataset, the maximum sequence length of the molecule (in its SMILES encoding) was found to be of length 412. Therefore, we choose the optimal input sequence length post data preprocessing to be 450. The vocabulary size was 79 and the dmodel is set to 64. With these settings the total number of parameters in our model was 342,674 and we chose the maximum possible batch-size to fit on our GPU set-up, which was 40. We randomly split the dataset in the ratio 3:2 for training and validation. However, in this particular scenario we augment the canonical SMILES and train *only* on the augmented SMILES. Our model is shown none of the original canonical SMILES during training and validation. Canonical SMILES are used only for obtaining the projection vectors during testing and the analyses of the latent space.

For the SI2 dataset, the maximum molecule length was 619 and therefore we chose to train the model with input sequence length of 650. The total vocabulary size of the dataset is 69. The dimensionality dmodel of the model is varied for this dataset from around 48 to 256. For most of our analysis however, we choose 256 dimensional latent space or dmodel = 256. Therefore, we focus on the settings for this case only. The batch size was set to 20 and the model had a total of 4,678,864 training parameters. In this case, the dataset was split such that 125,000 molecules were used for training and 25,000 reserved for validation.

5. Conclusions

The combination of transformers, contrastive learning and an autoencoder head allows the production of a powerful and disentangled learning system that we have applied to the problem of small molecule similarity. It also admitted a clear understanding of the sparseness with which the space was populated even by over 150,000 molecules, giving optimism that these methods, when scaled to greater numbers of molecules, can learn many molecular properties of interest to the computational chemical biologist.

Supplementary Materials: There are no supplementary materials.

Author Contributions: Conceptualization, all authors; methodology, all authors; software, A.D.S.; resources, D,B.K.; data curation, A.D.S.; writing—original draft preparation, D.B.K.; writing—review and editing, all authors;; funding acquisition, D.B.K. All authors have read and agreed to the published version of the manuscript

Funding: ADS was an intern sponsored by the University of Liverpool. DBK is also funded by the Novo Nordisk Foundation (grant NNF10CC1016517).

Acknowledgments: We thanks Drs Soumitra Samanta and Neil Swainston for useful discussions.

Conflicts of Interest: The authors declare that they have no conflicts of interest. "The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results".

References

1. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436-444.
2. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw* **2015**, *61*, 85-117.
3. Samanta, S.; O'Hagan, S.; Swainston, N.; Roberts, T.J.; Kell, D.B. Vae-sim: A novel molecular similarity measure based on a variational autoencoder. *Molecules* **2020**, *25*, 3446.
4. Kingma, D.; Welling, M. Auto-encoding variational bayes. *arXiv* **2014**, 1312.6114v1310.
5. Kingma, D.P.; Welling, M. An introduction to variational autoencoders. *arXiv* **2019**, 1906.02691v02691.
6. Wei, R.; Mahmood, A. Recent advances in variational autoencoders with representation learning for biomedical informatics: A survey. *Ieee Access* **2021**, *9*, 4939-4956.
7. Wei, R.; Garcia, C.; El-Sayed, A.; Peterson, V.; Mahmood, A. Variations in variational autoencoders - a comparative evaluation. *Ieee Access* **2020**, *8*, 153651-153670.
8. van Deursen, R.; Tetko, I.V.; Godin, G. Beyond chemical 1d knowledge using transformers. *arXiv* **2020**, 2010.01027.
9. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, 1706.03762.
10. Chithrananda, S.; Grand, G.; Ramsundar, B. Chemberta: Large-scale self-supervised pretraining for molecular property prediction. *arXiv* **2020**, 2010.09885.
11. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. *arXiv* **2017**, 1703.01365.
12. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv* **2013**, 1312.6034.
13. Azodi, C.B.; Tang, J.; Shiu, S.H. Opening the black box: Interpretable machine learning for geneticists. *Trends Genet* **2020**, *36*, 442-455.
14. Core, M.G.; Lane, H.C.; van Lent, M.; Gomboc, D.; Solomon, S.; Rosenberg, M. Building explainable artificial intelligence systems. *AAAI* **2006**, 1766-1773.
15. Holzinger, A.; Biemann, C.; Pattichis, C.S.; Kell, D.B. What do we need to build explainable ai systems for the medical domain? *arXiv* **2017**, 1712.09923v09921.

16. Samek, W.; Montavon, G.; Vedaldi, A.; Hansen, L.K.; Müller, K.-R. *Explainable ai: Interpreting, explaining and visualizing deep learning*. Springer: Berlin, 2019.
17. Singh, A.; Sengupta, S.; Lakshminarayanan, V. Explainable deep learning models in medical image analysis. *arXiv* **2020**, 2005.13799.
18. Tjoa, E.; Guan, C. A survey on explainable artificial intelligence (xai): Towards medical xai. *arXiv* **2019**, 1907.07374.
19. Parmar, G.; Li, D.; Lee, K.; Tu, Z. Dual contradistinctive generative autoencoder. *arXiv* **2020**, 2011.10063.
20. Peis, I.; Olmos, P.M.; Artés-Rodríguez, A. Unsupervised learning of global factors in deep generative models. *arXiv* **2020**, 2012.08234.
21. Klys, J.; Snell, J.; Zemel, R. Learning latent subspaces in variational autoencoders. *arXiv* **2018**, 1812.06190.
22. He, Z.; Kan, M.; Zhang, J.; Shan, S. Pa-gan: Progressive attention generative adversarial network for facial attribute editing. *arXiv* **2020**, 2007.05892.
23. Shen, X.; Liu, F.; Dong, H.; Lian, Q.; Chen, Z.; Zhang, T. Disentangled generative causal representation learning. *arXiv* **2020**, 2010.02637.
24. Esser, P.; Rombach, R.; Ommer, B. A note on data biases in generative models. *arXiv* **2020**, 2012.02516.
25. Kumar, A.; Sattigeri, P.; Balakrishnan, A. Variational inference of disentangled latent concepts from unlabeled observations. *arXiv* **2017**, 1711.00848.
26. Kim, H.; Mnih, A. Disentangling by factorising. *arXiv* **2018**, 1802.05983.
27. Locatello, F.; Bauer, S.; Lucic, M.; Rätsch, G.; Gelly, S.; Schölkopf, B.; Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. *arXiv* **2018**, 1811.12359.
28. Locatello, F.; Tschannen, M.; Bauer, S.; Rätsch, G.; Schölkopf, B.; Bachem, O. Disentangling factors of variation using few labels. *arXiv* **2019**, 1905.01258v01251
29. Locatello, F.; Poole, B.; Rätsch, G.; Schölkopf, B.; Bachem, O.; Tschannen, M. Weakly-supervised disentanglement without compromises. *arXiv* **2020**, 2002.02886.
30. Oldfield, J.; Panagakis, Y.; Nicolaou, M.A. Adversarial learning of disentangled and generalizable representations of visual attributes. *IEEE Trans Neural Netw Learn Syst* **2021**, *PP*.
31. Pandey, A.; Schreurs, J.; Suykens, J.A.K. Generative restricted kernel machines: A framework for multi-view generation and disentangled feature learning. *Neural Netw* **2021**, *135*, 177-191.
32. Hao, Z.; Lv, D.; Li, Z.; Cai, R.; Wen, W.; Xu, B. Semi-supervised disentangled framework for transferable named entity recognition. *Neural Netw* **2021**, *135*, 127-138.
33. Shen, Y.; Yang, C.; Tang, X.; Zhou, B. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE Trans Pattern Anal Mach Intell* **2020**, *PP*.
34. Tang, Y.; Tang, Y.; Zhu, Y.; Xiao, J.; Summers, R.M. A disentangled generative model for disease decomposition in chest x-rays via normal image synthesis. *Med Image Anal* **2021**, *67*, 101839.
35. Cootes, T.F.; Edwards, G.J.; Taylor, C.J. Active appearance models. *IEEE Trans Pattern Anal Mach Intell* **2001**, *23*, 681-685.
36. Cootes, T.F.; Taylor, C.J.; Cooper, D.H.; Graham, J. Active shape models - their training and application. *Comput Vis Image Und* **1995**, *61*, 38-59.
37. Hill, A.; Cootes, T.F.; Taylor, C.J. Active shape models and the shape approximation problem. *Image Vision Comput* **1996**, *14*, 601-607.
38. Salam, H.; Seguier, R. A survey on face modeling: Building a bridge between face analysis and synthesis. *Visual Comput* **2018**, *34*, 289-319.

39. Bozkurt, A.; Esmaeili, B.; Brooks, D.H.; Dy, J.G.; van de Meent, J.-W. Evaluating combinatorial generalization in variational autoencoders. *arXiv* **2019**, 1911.04594v04591.
40. Alemi, A.A.; Poole, B.; Fischer, I.; Dillon, J.V.; Saurous, R.A.; Murphy, K. Fixing a broken elbo. *arXiv* **2019**, 1711.00464.
41. Zhao, S.; Song, J.; 1, S.E. Infovae: Balancing learning and inference in variational autoencoders. *arXiv* **2017**, 1706.02262v02263.
42. Leibfried, F.; Dutordoir, V.; John, S.T.; Durrande, N. A tutorial on sparse gaussian processes and variational inference. *arXiv* **2020**, 2012.13962.
43. Rezende, D.J.; Viola, F. Taming vaes. *arXiv* **2018**, 1810.00597v00591.
44. Dai, B.; Wipf, D. Diagnosing and enhancing vae models. *arXiv* **2019**, 1903.05789v05782.
45. Li, Y.; Yu, S.; Principe, J.C.; Li, X.; Wu, D. Pri-vae: Principle-of-relevant-information variational autoencoders. *arXiv* **2020**, 2007.06503.
46. Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; Lerchner, A. B-vae: Learning basic visual concepts with a constrained variational framework. *Proc ICLR* **2017**.
47. Burgess, C.P.; Higgins, I.; Pal, A.; Matthey, L.; Watters, N.; Desjardins, G.; Lerchner, A. Understanding disentangling in β-vae. *arXiv* **2018**, 1804.03599.
48. Havtorn, J.D.; Frellsen, J.; Hauberg, S.; Maaløe, L. Hierarchical vaes know what they don't know. *arXiv* **2021**, 2102.08248.
49. Kumar, A.; Poole, B. On implicit regularization in β-vaes. *arXiv* **2021**, 2002.00041.
50. Yang, T.; Ren, X.; Wang, Y.; Zeng, W.; Zheng, N.; Ren, P. Groupifyvae: From group-based definition to vae-based unsupervised representation disentanglement. *arXiv* **2021**, 2102.10303.
51. Gatopoulos, I.; Tomczak, J.M. Self-supervised variational auto-encoders. *arXiv* **2020**, 2010.02014.
52. Rong, Y.; Bian, Y.; Xu, T.; Xie, W.; Wei, Y.; Huang, W.; Huang, J. Self-supervised graph transformer on large-scale molecular data. *arXiv* **2020**, 2007.02835.
53. Saeed, A.; Grangier, D.; Zeghidour, N. Contrastive learning of general-purpose audio representations. *arXiv* **2020**, 2010.10915.
54. Aneja, J.; Schwing, A.; Kautz, J.; Vahdat, A. Ncp-vae: Variational autoencoders with noise contrastive priors. *arXiv* **2020**, 2010.02917.
55. Artelt, A.; Hammer, B. Efficient computation of contrastive explanations. *arXiv* **2020**, 2010.02647.
56. Ciga, O.; Martel, A.L.; Xu, T. Self supervised contrastive learning for digital histopathology. *arXiv* **2020**, 2011.13971.
57. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv* **2020**, 2002.05709.
58. Jaiswal, A.; Babu, A.R.; Zadeh, M.Z.; Banerjee, D.; Makedon, F. A survey on contrastive self-supervised learning. *arXiv* **2020**, 2011.00362.
59. Purushwalkam, S.; Gupta, A. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *arXiv* **2020**, 2007.13916.
60. van den Oord, A.; Li, Y.; Vinyals, O. Representation learning with contrastive predictive coding. *arXiv* **2018**, 1807.03748v03742.
61. Verma, V.; Luong, M.-T.; Kawaguchi, K.; Pham, H.; Le, Q.V. Towards domain-agnostic contrastive learning. *arXiv* **2020**, 2011.04419.
62. Le-Khac, P.H.; Healy, G.; Smeaton, A.F. Contrastive representation learning: A framework and review. *arXiv* **2020**, 2010.05113.

63. Wang, Q.; Meng, F.; Breckon, T.P. Data augmentation with norm-vae for unsupervised domain adaptation. *arXiv* **2020**, 2012.00848.
64. Li, H.; Zhang, X.; Sun, R.; Xiong, H.; Tian, Q. Center-wise local image mixture for contrastive representation learning. *arXiv* **2020**, 2011.02697.
65. You, Y.; Chen, T.; Sui, Y.; Chen, T.; Wang, Z.; Shen, Y. Graph contrastive learning with augmentations. *arXiv* **2020**, 2010.13902.
66. Willett, P. Similarity-based data mining in files of two-dimensional chemical structures using fingerprint measures of molecular resemblance. *Wires Data Min Knowl* **2011**, *1*, 241-251.
67. Stumpfe, D.; Bajorath, J. Similarity searching. *Wires Comput Mol Sci* **2011**, *1*, 260-282.
68. Maggiora, G.; Vogt, M.; Stumpfe, D.; Bajorath, J. Molecular similarity in medicinal chemistry. *J Med Chem* **2014**, *57*, 3186-3204.
69. Irwin, J.J.; Shoichet, B.K. Zinc--a free database of commercially available compounds for virtual screening. *J Chem Inf Model* **2005**, *45*, 177-182.
70. Ertl, P.; Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J Cheminform* **2009**, *1*, 8.
71. Bickerton, G.R.; Paolini, G.V.; Besnard, J.; Muresan, S.; Hopkins, A.L. Quantifying the chemical beauty of drugs. *Nat Chem* **2012**, *4*, 90-98.
72. Arús-Pous, J.; Awale, M.; Probst, D.; Reymond, J.L. Exploring chemical space with machine learning. *Chimia (Aarau)* **2019**, *73*, 1018-1023.
73. Awale, M.; Probst, D.; Reymond, J.L. Webmolcs: A web-based interface for visualizing molecules in three-dimensional chemical spaces. *J Chem Inf Model* **2017**, *57*, 643-649.
74. Baldi, P.; Muller, K.R.; Schneider, G. Charting chemical space: Challenges and opportunities for artificial intelligence and machine learning. *Mol Inform* **2011**, *30*, 751-752.
75. Chen, Y.; Garcia de Lomana, M.; Friedrich, N.O.; Kirchmair, J. Characterization of the chemical space of known and readily obtainable natural products. *J Chem Inf Model* **2018**, *58*, 1518-1532.
76. Drew, K.L.M.; Baiman, H.; Khwaounjoo, P.; Yu, B.; Reynisson, J. Size estimation of chemical space: How big is it? *J Pharm Pharmacol* **2012**, *64*, 490-495.
77. Ertl, P. Visualization of chemical space for medicinal chemists. *J Cheminform* **2014**, *6*, O4.
78. Gonzalez-Medina, M.; Prieto-Martinez, F.D.; Naveja, J.J.; Mendez-Lucio, O.; El-Elimat, T.; Pearce, C.J.; Oberlies, N.H.; Figueroa, M.; Medina-Franco, J.L. Chemoinformatic expedition of the chemical space of fungal products. *Future Med Chem* **2016**, *8*, 1399-1412.
79. Klimenko, K.; Marcou, G.; Horvath, D.; Varnek, A. Chemical space mapping and structure-activity analysis of the chembl antiviral compound set. *J Chem Inf Model* **2016**, *56*, 1438-1454.
80. Lin, A.; Horvath, D.; Afonina, V.; Marcou, G.; Reymond, J.L.; Varnek, A. Mapping of the available chemical space versus the chemical universe of lead-like compounds. *ChemMedChem* **2018**, *13*, 540-554.
81. Lucas, X.; Gruning, B.A.; Bleher, S.; Günther, S. The purchasable chemical space: A detailed picture. *J Chem Inf Model* **2015**, *55*, 915-924.
82. Nigam, A.; Friederich, P.; Krenn, M.; Aspuru-Guzik, A. Augmenting genetic algorithms with deep neural networks for exploring the chemical space. *arXiv* **2019**, 1909.11655.
83. O'Hagan, S.; Kell, D.B. Generation of a small library of natural products designed to cover chemical space inexpensively. *Pharm Front* **2019**, *1*, e190005.
84. Polishchuk, P.G.; Madzhidov, T.I.; Varnek, A. Estimation of the size of drug-like chemical space based on gdb-17 data. *J Comput Aided Mol Des* **2013**, *27*, 675-679.
85. Reymond, J.L. The chemical space project. *Acc Chem Res* **2015**, *48*, 722-730.

86. Rosén, J.; Gottfries, J.; Muresan, S.; Backlund, A.; Oprea, T.I. Novel chemical space exploration via natural products. *J Med Chem* **2009**, *52*, 1953-1962.
87. Thakkar, A.; Selmi, N.; Reymond, J.L.; Engkvist, O.; Bjerrum, E. 'Ring breaker': Neural network driven synthesis prediction of the ring system chemical space. *J Med Chem* **2020**.
88. Thiede, L.A.; Krenn, M.; Nigam, A.; Aspuru-Guzik, A. Curiosity in exploring chemical space: Intrinsic rewards for deep molecular reinforcement learning. *ARxIV* **2020**, 2012.11293.
89. Bender, A.; Glen, R.C. Molecular similarity: A key technique in molecular informatics. *Org Biomol Chem* **2004**, *2*, 3204-3218.
90. O'Hagan, S.; Kell, D.B. Consensus rank orderings of molecular fingerprints illustrate the 'most genuine' similarities between marketed drugs and small endogenous human metabolites, but highlight exogenous natural products as the most important 'natural' drug transporter substrates. *ADMET & DMPK* **2017**, *5*, 85-125.
91. Sterling, T.; Irwin, J.J. Zinc 15 - ligand discovery for everyone. *J Chem Inf Model* **2015**, *55*, 2324-2337.
92. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv* **2019**, 1910.10683.
93. Rives, A.; Goyal, S.; Meier, J.; Guo, D.; Ott, M.; Zitnick, C.L.; Ma, J.; Fergus, R. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *bioRxiv* **2019**, 622803.
94. So, D.R.; Liang, C.; Le, Q.V. The evolved transformer. *arXiv* **2019**, 1901.11117.
95. Grechishnikova, D. Transformer neural network for protein specific de novo drug generation as machine translation problem. *bioRxiv* **2020**, 10.1101/863415v863411.full.
96. Choromanski, K.; Likhosherstov, V.; Dohan, D.; Song, X.; Gane, A.; Sarlos, T.; Hawkins, P.; Davis, J.; Mohiuddin, A.; Kaiser, L., *et al.* Rethinking attention with performers. *arXiv* **2020**, 2009.14794.
97. Yun, C.; Bhojanapalli, S.; Rawat, A.S.; Reddi, S.J.; Kumar, S. Are transformers universal approximators of sequence-to-sequence functions? *arXiv* **2019**, 1912.10077.
98. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A., *et al.* Language models are few-shot learners. *arXiv* **2020**, 2005.14165.
99. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S., *et al.* An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, 2010.11929.
100. Fedus, W.; Zoph, B.; Shazeer, N. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv* **2021**, 2101.03961.
101. Wang, Y.; Wang, J.; Cao, Z.; Farimani, A.B. Molclr: Molecular contrastive learning of representations via graph neural networks. *arXiv* **2021**, 2102.10056.
102. Chen, T.; Kornblith, S.; Swersky, K.; Norouzi, M.; Hinton, G. Big self-supervised models are strong semi-supervised learners. *arXiv* **2020**, 2006.10029.
103. O'Hagan, S.; Kell, D.B. Structural similarities between some common fluorophores used in biology, marketed drugs, endogenous metabolites, and natural products. *Marine Drugs* **2020**, *18*, 582.
104. Chochlakis, G.; Georgiou, E.; Potamianos, A. End-to-end generative zero-shot learning via few-shot learning. *arXiv* **2021**, 2102.04379.
105. Majumder, O.; Ravichandran, A.; Maji, S.; Polito, M.; Bhotika, R.; Soatto, S. Revisiting contrastive learning for few-shot classification. *arXiv* **2021**, 2101.11058.
106. Dasari, S.; Gupta, A. Transformers for one-shot visual imitation. *arXiv* **2020**, 2011.05970.

107. Logeswaran, L.; Lee, A.; Ott, M.; Lee, H.; Ranzato, M.A.; Szlam, A. Few-shot sequence learning with transformers. *arXiv* **2020**, 2012.09543.
108. Belkin, M.; Hsu, D.; Ma, S.; Mandal, S. Reconciling modern machine-learning practice and the classical bias-variance trade-off. *Proc Natl Acad Sci U S A* **2019**, *116*, 15849-15854.
109. van der Maaten, L.; Hinton, G. Visualizing data using t-sne. *J Machine Learning Res* **2008**, *9*, 2579-2605.
110. van der Maaten, L. Learning a parametric embedding by preserving local structure. *Proc AISTATS* **2009**, 384-391.
111. McInnes, L.; Healy, J.; Melville, J. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv* **2018**, 1802.03426v03422.
112. McInnes, L.; Healy, J.; Saul, N.; Großberger, L. Umap: Uniform manifold approximation and projection. *J Open Source Software* **2018**, *DOI 10.21105/joss.00861*.
113. Dickens, D.; Rädisch, S.; Chiduza, G.N.; Giannoudis, A.; Cross, M.J.; Malik, H.; Schaeffeler, E.; Sison-Young, R.L.; Wilkinson, E.L.; Goldring, C.E., *et al.* Cellular uptake of the atypical antipsychotic clozapine is a carrier-mediated process. *Mol Pharm* **2018**, *15*, 3557-3572.
114. Horvath, D.; Jeandenans, C. Neighborhood behavior of *in silico* structural spaces with respect to in vitro activity spaces-a novel understanding of the molecular similarity principle in the context of multiple receptor binding profiles. *J Chem Inf Comput Sci* **2003**, *43*, 680-690.
115. Bender, A.; Jenkins, J.L.; Li, Q.L.; Adams, S.E.; Cannon, E.O.; Glen, R.C. Molecular similarity: Advances in methods, applications and validations in virtual screening and qsar. *Annual Reports in Computational Chemistry, Vol 2* **2006**, *2*, 141-168.
116. Horvath, D.; Koch, C.; Schneider, G.; Marcou, G.; Varnek, A. Local neighborhood behavior in a combinatorial library context. *J Comput Aid Mol Des* **2011**, *25*, 237-252.
117. Gasteiger, J. *Handbook of chemoinformatics: From data to knowledge*. Wiley/VCH: Weinheim, 2003.
118. Bajorath, J. *Chemoinformatics: Concepts, methods and tools for drug discovery*. Humana Press: Totowa, NJ, 2004.
119. Sutherland, J.J.; Raymond, J.W.; Stevens, J.L.; Baker, T.K.; Watson, D.E. Relating molecular properties and *in vitro* assay results to *in vivo* drug disposition and toxicity outcomes. *J Med Chem* **2012**, *55*, 6455-6466.
120. Capecchi, A.; Probst, D.; Reymond, J.L. One molecular fingerprint to rule them all: Drugs, biomolecules, and the metabolome. *J Cheminform* **2020**, *12*, 43.
121. Muegge, I.; Mukherjee, P. An overview of molecular fingerprint similarity search in virtual screening. *Expert Opin Drug Discov* **2016**, *11*, 137-148.
122. Nisius, B.; Bajorath, J. Rendering conventional molecular fingerprints for virtual screening independent of molecular complexity and size effects. *ChemMedChem* **2010**, *5*, 859-868.
123. Riniker, S.; Landrum, G.A. Similarity maps - a visualization strategy for molecular fingerprints and machine-learning methods. *J Cheminform* **2013**, *5*, 43.
124. Vogt, I.; Stumpfe, D.; Ahmed, H.E.; Bajorath, J. Methods for computer-aided chemical biology. Part 2: Evaluation of compound selectivity using 2d molecular fingerprints. *Chem Biol Drug Des* **2007**, *70*, 195-205.
125. O'Hagan, S.; Swainston, N.; Handl, J.; Kell, D.B. A 'rule of 0.5' for the metabolite-likeness of approved pharmaceutical drugs. *Metabolomics* **2015**, *11*, 323-339.
126. O'Hagan, S.; Kell, D.B. Understanding the foundations of the structural similarities between marketed drugs and endogenous human metabolites. *Front Pharmacol* **2015**, *6*, 105.

127. O'Hagan, S.; Kell, D.B. The apparent permeabilities of caco-2 cells to marketed drugs: Magnitude, and independence from both biophysical properties and endogenite similarities *PeerJ* **2015**, *3*, e1405.
128. O'Hagan, S.; Kell, D.B. Metmaxstruct: A tversky-similarity-based strategy for analysing the (sub)structural similarities of drugs and endogenous metabolites. *Front Pharmacol* **2016**, *7*, 266.
129. O'Hagan, S.; Kell, D.B. Analysis of drug-endogenous human metabolite similarities in terms of their maximum common substructures. *J Cheminform* **2017**, *9*, 18.
130. O'Hagan, S.; Kell, D.B. Analysing and navigating natural products space for generating small, diverse, but representative chemical libraries. *Biotechnol J* **2018**, *13*, 1700503.
131. Gawehn, E.; Hiss, J.A.; Schneider, G. Deep learning in drug discovery. *Mol Inform* **2016**, *35*, 3-14.
132. 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.; Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent Sci* **2018**, *4*, 268-276.
133. Sanchez-Lengeling, B.; Aspuru-Guzik, A. Inverse molecular design using machine learning: Generative models for matter engineering. *Science* **2018**, *361*, 360-365.
134. Arús-Pous, J.; Probst, D.; Reymond, J.L. Deep learning invades drug design and synthesis. *Chimia (Aarau)* **2018**, *72*, 70-71.
135. Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M., *et al.* Analyzing learned molecular representations for property prediction. *J Chem Inf Model* **2019**, *59*, 3370-3388.
136. Zhavoronkov, A.; Ivanenkov, Y.A.; Aliper, A.; Veselov, M.S.; Aladinskiy, V.A.; Aladinskaya, A.V.; Terentiev, V.A.; Polykovskiy, D.A.; Kuznetsov, M.D.; Asadulaev, A., *et al.* Deep learning enables rapid identification of potent ddr1 kinase inhibitors. *Nat Biotechnol* **2019**.
137. Khemchandani, Y.; O'Hagan, S.; Samanta, S.; Swainston, N.; Roberts, T.J.; Bollegala, D.; Kell, D.B. Deepgraphmolgen, a multiobjective, computational strategy for generating molecules with desirable properties: A graph convolution and reinforcement learning approach. *J Cheminform* **2020**, *12*, 53.
138. Shen, C.; Krenn, M.; Eppel, S.; Aspuru-Guzik, A. Deep molecular dreaming: Inverse machine learning for de-novo molecular design and interpretability with surjective representations. *arXiv* **2020**, 2012.09712.
139. Moret, M.; Friedrich, L.; Grisoni, F.; Merk, D.; Schneider, G. Generative molecular design in low data regimes. *Nat Mach Intell* **2020**, *2*, 171-180.
140. Kell, D.B.; Samanta, S.; Swainston, N. Deep learning and generative methods in cheminformatics and chemical biology: Navigating small molecule space intelligently *Biochem J* **2020**, *477*, 4559-4580.
141. Walters, W.P.; Barzilay, R. Applications of deep learning in molecule generation and molecular property prediction. *Acc Chem Res* **2021**, *54*, 263-270.
142. Zaheer, M.; Guruganesh, G.; Dubey, A.; Ainslie, J.; Alberti, C.; Ontanon, S.; Pham, P.; Ravula, A.; Wang, Q.; Yang, L., *et al.* Big bird: Transformers for longer sequences. *arXiv* **2020**, 2007.14062.
143. 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.
144. Tetko, I.V.; Karpov, P.; Van Deursen, R.; Godin, G. State-of-the-art augmented nlp transformer models for direct and single-step retrosynthesis. *Nature Communications* **2020**, *11*.
145. Lim, S.; Lee, Y.O. Predicting chemical properties using self-attention multi-task learning based on smiles representation. *arXiv* **2020**, 2010.11272.
146. Pflüger, P.M.; Glorius, F. Molecular machine learning: The future of synthetic chemistry? *Angew Chem Int Ed Engl* **2020**.

147. Shin, B.; Park, S.; Bak, J.; Ho, J.C. Controlled molecule generator for optimizing multiple chemical properties. *arXiv* **2020**, 2010.13908.
148. Liu, X.; Zhang, F.; Hou, Z.; Mian, L.; Wang, Z.; Zhang, J.; Tang, J. Self-supervised learning: Generative or contrastive. *arXiv* **2020**, 2006.08218v08214.
149. Wanyan, T.; Honarvar, H.; Jaladanki, S.K.; Zang, C.; Naik, N.; Somani, S.; Freitas, J.K.D.; Paranjpe, I.; Vaid, A.; Miotto, R., *et al.* Contrastive learning improves critical event prediction in covid-19 patients. *arXiv* **2021**, 2101.04013.
150. Kostas, D.; Aroca-Ouellette, S.; Rudzicz, F. Bendr: Using transformers and a contrastive self-supervised learning task to learn from massive amounts of eeg data. *arXiv* **2021**, 2101.12037.
151. Everitt, B.S. *Cluster analysis*. Edward Arnold: London, 1993.
152. Botvinick, M.; Barrett, D.G.T.; Battaglia, P.; de Freitas, N.; Kumaran, D.; Leibo, J.Z.; Lillicrap, T.; Modayil, J.; Mohamed, S.; Rabinowitz, N.C., *et al.* Building machines that learn and think for themselves. *Behav Brain Sci* **2017**, *40*, e255.
153. Hassabis, D.; Kumaran, D.; Summerfield, C.; Botvinick, M. Neuroscience-inspired artificial intelligence. *Neuron* **2017**, *95*, 245-258.
154. Shevlin, H.; Vold, K.; Crosby, M.; Halina, M. The limits of machine intelligence despite progress in machine intelligence, artificial general intelligence is still a major challenge. *EMBO Rep* **2019**, *20*.
155. Pei, J.; Deng, L.; Song, S.; Zhao, M.; Zhang, Y.; Wu, S.; Wang, G.; Zou, Z.; Wu, Z.; He, W., *et al.* Towards artificial general intelligence with hybrid tianjic chip architecture. *Nature* **2019**, *572*, 106-111.
156. Stanley, K.O.; Clune, J.; Lehman, J.; Miikkulainen, R. Designing neural networks through neuroevolution. *Nat Mach Intell* **2019**, *1*, 24-35.
157. Zhang, Y.; Qu, P.; Ji, Y.; Zhang, W.; Gao, G.; Wang, G.; Song, S.; Li, G.; Chen, W.; Zheng, W., *et al.* A system hierarchy for brain-inspired computing. *Nature* **2020**, *586*, 378-384.
158. Nadji-Tehrani, M.; Eslami, A. A brain-inspired framework for evolutionary artificial general intelligence. *IEEE Trans Neural Netw Learn Syst* **2020**, *31*, 5257-5271.
159. Bjerrum, E.J. Smiles enumeration as data augmentation for neural network modeling of molecules. *arXiv* **2017**, 1703.07076.
160. Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. . *NIPS* **2016**, *30*, 1857–1865.
161. Wu, Z.; Xiong, Y.; Yu, S.; Lin, D. Unsupervised feature learning via non-parametric instance-level discrimination. *arXiv* **2018**, 1805.01978.
162. Kingma, D.P.; Ba, J.L. Adam: A method for stochastic optimization. *arXiv* **2015**, 1412.6980v1418.
163. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J Machine Learning Res* **2014**, *15*, 1929-1958.