Predicting Binding from Screening Assays with Transformer Network Embeddings

Paul Morris,* Rachel St. Clair, Elan Barenholtz, and William Edward Hahn

Center for Complex Systems and Brain Sciences, Florida Atlantic University, Boca Raton, Florida 33431, United States

E-mail: pmorris2012@fau.edu

Abstract

Cheminformatics aims to assist in chemistry applications that depend on molecular interactions, structural characteristics, and functional properties. The arrival of deep learning and the abundance of easily accessible chemical data from repositories like PubChem have enabled advancements in computer-aided drug discovery. Virtual High-Throughput Screening (vHTS) is one such technique that integrates chemical domain knowledge to perform in silico biomolecular simulations, but prediction of binding affinity is restricted due to limited availability of ground-truth binding assay results. Here, text representations of 83,000,000 molecules are leveraged to enable single-target binding affinity prediction directly on the outcome of screening assays. The embedding of an end-to-end Transformer neural network, trained to encode the structural characteristics of a molecule via a text-based translation task, is repurposed through transfer learning to classify binding affinity to a single target. Classifiers trained on the embedding outperform those trained on SMILES strings for multiple tasks, receiving between 0.67-0.99 AUC. Visualization reveals organization of structural and functional properties in the learned embedding useful for binding prediction. The proposed model is suitable for parallel computing, enabling rapid screening as a complement to virtual screening techniques when limited data is available.

Introduction

Cheminformatics aims to assist in chemistry applications that depend on molecular interactions, structural characteristics, and functional properties. The arrival of powerful computational techniques and the abundance of easily accessible chemical data from repositories have enabled dramatic recent advancements in computer-aided drug discovery. The domain of computer-aided drug design ranges from quantitative structure activity relationship,[1] drug induced liver injury,[2] toxicity modeling,[3] virtual screening,[4] among others. All of these tasks have been aided by models that make use of computational techniques that leverage large datasets and human expertise to encode molecular features in order to predict biochemical activity.[5] Such techniques seek to expedite drug-discovery pipelines by increasing the quantity and quality of active compounds identified, potentially resulting in new drug leads. Computational approaches are also advantaged in their ability to integrate features from many sources describing different chemical properties to approximate chemical function without the limitations of traditional wet-lab approaches.[6]

Virtual High-Throughput Screening (vHTS) is one such technique that integrates chemical domain knowledge to perform in silico biomolecular simulations. While binding assays are generally more accurate than traditional virtual screening approaches, they can only identify drug leads from a set of com-

pounds which are easy and cost-efficient to synthesize. Computational techniques which simulate or approximate physical models of chemistry are not constrained by real-world limitations, as molecules do not need to be synthesized and resources for wet-lab experiments are not required. Models or algorithms which are driven by chemical data and expert knowledge have been used to estimate structural and functional properties and aid in scoring of existing molecules,[7,8] as well as in de novo drug design.[9]

Earlier approaches to the application of machine learning in cheminformatics involved more traditional techniques such as support vector machines (SVM), random-forest decision tree ensembles, markov models, and linear regression.[10] However, the advent of deep learning on parallel computing resources has increased the power and utility of computational models, leading to new opportunities to leverage the wealth of available machine-readable chemical information. Deep neural networks (DNNs) trained to classify molecular representations have reportedly been highly effective for cheminformatic tasks in computer-aided drug design, computational structural biology, quantum chemistry, and computational material design.

The recent success of deep learning can be attributed in part to the availability of large, labeled datasets.[11] Repositories such as PubChem[12] which compile information on molecular structure and properties have enabled the application of deep learning vision and natural language processing (NLP) techniques to many molecular property prediction tasks. These include training convolutional neural networks (CNN) on raw SMILES strings,[13,14] adapting CNNs to atom graphs and connectivity matrices,[15–18] and using neural networks to classify molecules from fingerprints or other hand-designed molecular descriptors.[19]

A number of approaches attempt to replace the hand-designed scoring function of traditional molecular docking algorithms with a learned scoring function.[16,20–22] Another class of deep learning applications for drug discovery attempts to simulate molecular docking. Large databases such as PDBBind[23] contain 3D conformations of molecules bound to relevant sites on thousands of target structures. Deep learning approaches encode this 3D information to learn a model of physics and identify molecules with low-energy conformations and high likelihood of binding.[24] Though physics-based molecular docking models are less constrained than wet-lab screening approaches, they can still be computationally expensive and require significant time and/or resources.

As an alternative to docking, other deep learning approaches attempt to improve the quality of virtual screening predictions by learning to represent molecules with automatically selected features.[25,26] In particular, translation between distinct molecular representations have previously been shown as an effective technique for learning useful representations of molecular properties.[9] By learning from existing representations and other information which describe structural patterns, these techniques develop custom chemical feature sets which can match or increase performance on molecular classification/prediction tasks compared to existing representations. Learning new representations expands the scope of cheminformatics applications by allowing prediction of molecular function, as improved representations can increase the predictive quality of models trained on limited amounts of data.

While the application of deep learning to prediction of molecular properties and other tasks has shown promise in aiding drug discovery, the direct application of deep learning to prediction of screening assay results has been made difficult by the limited quantity of available data. Molecules screened against a particular target likely constitute a much less representative sample of chemical space than is typical of dataset samples from vision or NLP populations, where deep learning has been most successful. The application of deep learning is especially difficult for datasets that are not primarily hand-engineered.[5]

To address these limitations, we leverage the vast wealth of publicly available and easily computable molecular structure data to augment training of a neural network for binding affinity prediction from historical assay data. To do

so, we train a Transformer neural network, an architecture first introduced in the context of natural language translation,[27] to translate between two distinct, text-based molecular representations in a well-studied subset of chemical space. An intermediate set of features computed by this trained model is considered as an embedding which contains abstract features describing general molecular structure. Molecules represented by this abstract embedding are then used to train a binding affinity prediction model directly on a limited set of assay results which quantify binding to a single target. The organization of structural and functional properties in embedding feature space enables simple classifiers to simulate screening assays in limited data scenarios.

Learning abstract representations of chemical information has recently been shown to improve performance in predicting molecular function.[5,28] Another recent study derived word embeddings and repurposed them through transfer learning for multiple NLP tasks, outperforming classification tasks without such embeddings.[29] Here, we utilize a Transformer network to create such embeddings for functional assays that may otherwise be poor candidates for virtual screening. Since the molecular representations from the Transformer network are learned by text-translation to encode the functional properties indicated by structure, they can be applied to any screening assay model, regardless of bioactivity. This approach shows pretraining embeddings for generic chemical representations can improve supervised classification. Our translation-based pretraining extends that insight to the task of predicting binding assay results.

We evaluate our novel molecular embedding learned by our Transformer on three single-target prediction tasks and observe improvement upon baselines for direct prediction of binding assay results. Since neural network training is data-driven, embedding features are also suitable for fine-tuning to consider target-specific information. Furthermore, The operations in the transformer model used to compute molecular embeddings are easily parallelizable on modern computing infrastructure (GPUs), enabling rapid screening of millions of molecules to assist wet-lab screening assays and other drug discovery pipelines.

Methods

To accurately predict binding assay results for a single target with few active compounds, we first perform an auxiliary text translation task based on state-of-the-art NLP techniques and structural text representations of millions of molecules. We collect SMILES strings and IUPAC chemical names for a large set of molecules on PubChem. SMILES and IUPAC representations are selected because they both describe similar aspects of molecular structure following consistent rules in a machine-readable format. While the atoms, bonds, and substructures described in the two representations are similar, the SMILES grammar and IUPAC nomenclature have distinct text representations. By learning to translate between the two, the common information they contain must be organized efficiently in an intermediate set of features. We then repurpose these features of the learned embedding for direct prediction of assay results, treated as a binary classification task between binding and non-binding regions of chemical space.

An overview of this process is shown in Figure 1. In Step 1, a high-level depiction of the network architecture is illustrated which demonstrates how the network layers generate molecular embeddings when performing SMILES-IUPAC translation. In Step 2, embeddings generated from the trained network are provided as input to a target-specific binding classification network.

Transformer Neural Network

The Transformer[27] is a deep neural network suited for NLP tasks. It relies on large weight matrices to store patterns and learn short and long-term dependencies in training sequences. The Transformer network architecture is flexible and can be used for both classification and text generation tasks. In our implementation,

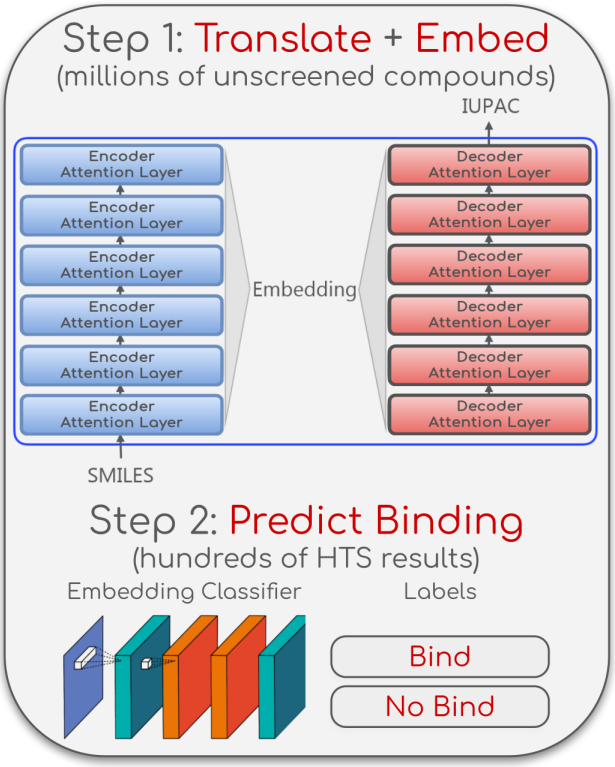

Figure 1: Diagram of the two-step procedure followed to predict binding affinity using the learned embedding of a Transformer network.

the Transformer is used to generate output as an IUPAC chemical name which corresponds to a molecule described by a SMILES string provided as input.

Before being processed by the main layers of the Transformer, SMILES strings are converted to an initial, random embedding. Each character in the SMILES alphabet is replaced with a random vector, where the same vector is used for multiple occurrences of the same character. The values in this vector are the first network weights of the Transformer, and they are tuned during training based on the frequency, co-occurence, and sequential dependencies of each SMILES character. Periodic functions at different frequencies are added to the signal of each vector so that the frequency of the added signal encodes a character's location in the SMILES sequence. This allows the character-specific layers of the Transformer to determine the order of one character relative to others.

Once an initial embedding of character vectors are generated, the signal in each vector is modulated by the layers in the Transformer's encoder stack. An encoder layer consists of a self-attention operation which modifies each character vector based on its relation to other characters in the sequence, followed by a simple matrix multiplication and nonlinearity which is applied on each character vector individually. The output of each layer is a set of character vectors with the same size as the input. The output of the final encoder layer is treated as a molecular embedding, where each character vector has been modified to contain abstract features useful for describing the structure of a molecule. The features in this embedding are used by an equivalent set of decoder layers for IUPAC name generation. Character vectors are processed by the decoder stack one at a time, resulting in a new character in the IUPAC alphabet being predicted. Decoder layers share a similar structure to encoder layers, except for a slightly modified form of self-attention which looks at previously predicted IUPAC characters to inform prediction of the next character. During training, previous predictions are ignored in favor of characters from the correct IUPAC name for a molecule.

Molecular Self-Attention

The core mechanism of the Transformer is 'self-attention'. In this operation, input vectors representing each character in a SMILES string are output as a linear combination of vectors for all characters in the string. The output of the attention layer is a vector for each character with the same length as the input. However, vectors for each character are weighted by an output-specific attention score which represents how relevant every character in the string is to a particular output. Attention scores are computed by weight matrices, which accept character vectors as input. The meaning of the attention scores depends on the task on which the model is being trained, and multiple sets of model weights are used to produce multiple sets of scores which may attend to different relevant feature in the input. In the case of our translation task, attention scores may indicate the

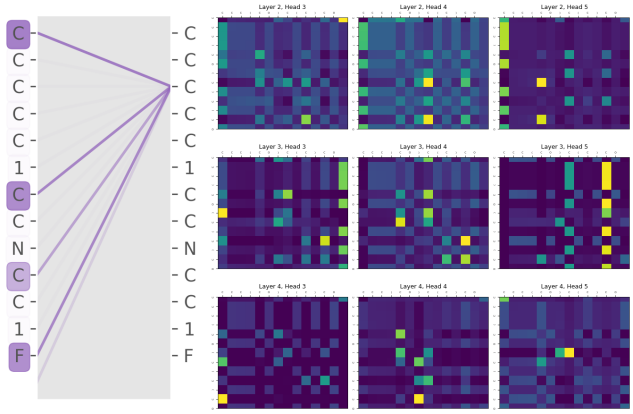

Figure 2: Left: A selection of attention weights for SMILES string CCCCC1CCNCC1F are visualized, showing how the character vectors of each input on the left are weighted to produce the output vector for the 3rd carbon atom on the right. Opacity indicates larger attention values. Right: Some of the self-attention weights from the trained Transformer are visualized for a molecule. Separate weights in a single layer attend to different features which describe the same structure.

importance of a certain substructure for generating part of a molecule's IUPAC name. Example visualizations from the trained Transformer are shown in Figure 2. The matrices on the right of the figure demonstrate the capacity of the Transformer network to learn a descriptive, varied set of abstract molecular features useful for describing structure.

Training Procedure

To train the Transformer network for translation, pairs of SMILES strings and IUPAC names are sourced directly from the PubChem compound database for 83,000,000 molecules. SMILES strings are used as-is, and no canonicalization is performed . Similarly, IUPAC names for each molecule are collected from PubChem with no modification. Deep neural networks trained on large, labeled datasets have been shown to be robust to unreliable annotation.[30] Noise during neural network training can increase generalization due to the intricacies of network optimization,[31] making the unmodified molecular text representations robust to under-fitting.

A Transformer network is created with 512-dimension character embedding vectors. A maximum SMILES string length during training of 256 characters is imposed, although this limit can be exceeded during screening inference. Thus, each molecular embedding contains $256 * 512$ dimensions. Training is performed in batches of 96 molecular string pairs. The Adam optimization algorithm[32] is used to to update the weights of the network. The learning rate during optimization begins at 0.001 and decreases two orders of magnitude, following half a period of a cosine function, over the course of a single pass, or epoch, over the 83,000,000 molecule training set. Training continues for three epochs.

Experiments

The utility of the Transformer embedding was investigated by training and evaluating binding prediction models on molecular embeddings for three binary classification tasks. Equivalent prediction models are also trained on two representation baselines to quantify the Transformer embeddings' usefulness for binding affinity prediction and explain the relation between learned features and molecular properties. Finally, an unsupervised evaluation of the learned embedding is performed by visualizing how changes in molecular structure compare correspond to embedding changes.

Assay Datasets

Three datasets for supervised prediction of binding assay results were compiled. In each, the results of an assay (or compilation of assays) for binding affinity to a target were sampled into a small, labeled dataset. Binary classification of molecular embeddings was performed by binning continuous, assay-specific binding affinity values into binding and non-binding categories according to an activity threshold. In each assay, fewer binding than non-binding molecules were identified. To account for this, the non-binding sets were randomly undersam-

Table 1: Screening Assay Datasets

	HIV-1 Protease	AID 652067	AID 1053197
Data Source	BindingDB[33]	PubChem[34]	Pubchem[35]
Target	HIV-1 Protease	DAF-12	Sialic Acid Acetylesterase
Measured	Ki (nM)	% Activation at 6.8uM	% Inhibition at 9.66uM
Activity Cutoff	<100nM	>5.55%	>18%
Tested	7,462	370,276	370,256
Active	2,159	5,354	2,555
Balanced Dataset Size	4,318	10,708	5,110

pled to match the count of binding molecules for the purpose of training and evaluating a balanced binding classifier. Thus, the dataset for each experiment was equally balanced between binding and non-binding compounds. Additional details regarding the assay procedures and results used in each of the three experiments are located in Table 1.

HIV-1 Protease

Molecules identified by various sources to bind and inhibit HIV-1 Protease were compiled from the BindingDB[33] filtered based on binding affinity. To acquire non-binding examples, random molecules were sampled from the PubChem Compound database. Any molecules which appeared in the HIV-binding set were excluded. Any discrepancies in the experimental procedures of the multiple screening assays in this compiled dataset were ignored and all Ki measurements were treated equivalently.

AID 652067

Molecules tested in a high-throughput screening assay for activation of the DAF-12 nuclear receptor in H.*contortus*[34] were compiled. The binding affinity threshold was set by the assay authors as three standard deviations above the average percent activation measured at the tested concentration. Molecules beyond this threshold were labeled binding to DAF-12. A random sample of molecules within three standard deviations of the average were used as the non-binding set.

AID 1053197

Results from a high-throughput screening assay for binding and inhibition of Sialic Acid Acetylesterase (SIAE) were sampled. Molecules which caused inhibition greater than three standard deviations above the average of screen compounds were labeled as binding, while molecules within three standard deviations were randomly sampled as non-binding.

Affinity Prediction and Baselines

A simple, fully-connected neural network with four layers was created to classify each assay dataset. The number of input neurons in the network is determined by the size of the molecular representation used for classification. In the case of the Transformer embeddings, $256 * 512$ neurons are used. 500 neurons are computed in the first two hidden layers. Dropout[36] is used to mask a random 20% of network activations after the second layer. Following dropout, a hidden layer with 100 neurons is computed. Finally, two output neurons are used for binary classification. The Rectified Linear Unit (ReLU) nonlinear activation function[37] is applied after each hidden layer except the final output, where the softmax function is applied to convert the two raw outputs to non-binding and binding class probabilities, respectively. Embedding inputs are not normalized before classification.

In order to quantify and explain the Transformer embedding's predictiveness for binding affinity tasks, identical networks for two baseline representations are trained and evaluated on all three datasets.

Table 2: Numerical Features Computed for Each Molecule

RDKit Feature Name	Description
ExactMolWt	molecular weight
FractionCSP3	fraction of carbons with SP3 orbitals
HeavyAtomCount	count of heavy atoms
LabuteASA	accessible surface area
MaxAbsPartialCharge	largest partial charge
MinAbsPartialCharge	smallest partial charge
MolLogP	Log P (octanol-water partition coefficient)
NumAliphaticCarbocycles	count of carbocycles containing a non-aromatic bond
NumAliphaticHeterocycles	count of hetereocycles containing a non-aromatic bond
NumAromaticCarbocycles	count of carbocycles containing aromatic bonds
NumAromaticHeterocycles	count of heterocycles containing aromatic bonds
NumHAcceptors	count of nitrogen and oxygen atoms
NumHDonors	count of NH and OH bonds
NumHeteroatoms	count of atoms excluding carbon and hydrogen
NumRotatableBonds	count of bonds physically likely to rotate
NumSaturatedCarbocycles	count of carbocycles with only single bonds
NumSaturatedHeterocycles	count of heterocycles with only single bonds
NumValenceElectrons	count of valence electrons in the entire molecule
RingCount	count of all rings
TPSA	Surface area of polar atoms (oxygen and nitrogen)

Random Embedding Baseline

First, we compare trained versus untrained Transformer embeddings to quantify the relevance of features learned through our SMILES-IUPAC translation task for binding affinity prediction. In this case, the SMILES string for each molecule is converted to a random embedding of 512-dimensional vectors for each character. The random embeddings have the same initialization scheme as the Transformer network. Thus, this baseline is equivalent to evaluating embeddings produced by an untrained Transformer configured for our translation task. This is also equivalent to classifying SMILES strings directly with a neural network, as vectors are consistent for repeated characters across the dataset.

Properties Baseline

Second, we compare trained embedding performance versus a set of hand-selected properties describing molecular structure to investigate how the embedding may represent different quantifiable structural metrics. Properties were selected based on their potential relevance to the structural information in SMILES strings, as well as for binding affinity prediction. Values for 20 total properties were computed using RDKit[38] implementations. The properties, their RDKit identifiers, and brief descriptions are detailed in Table 2. Numeric property values were normalized between 0 and 1 according to the minimum and maximum values of all screened molecules, on a per-dataset basis. The networks used to classify binding affinity are identical to the Transformer and random embedding networks, except only 20 input neurons are needed in this case.

Evaluation

Binding classification networks trained on the three datasets for the Transformer embedding and two baseline representations were evaluated by computing balanced, binary classification accuracy between binding and non-binding categories, where chance is 50%. ROC curves and AUC were also computed by comparing

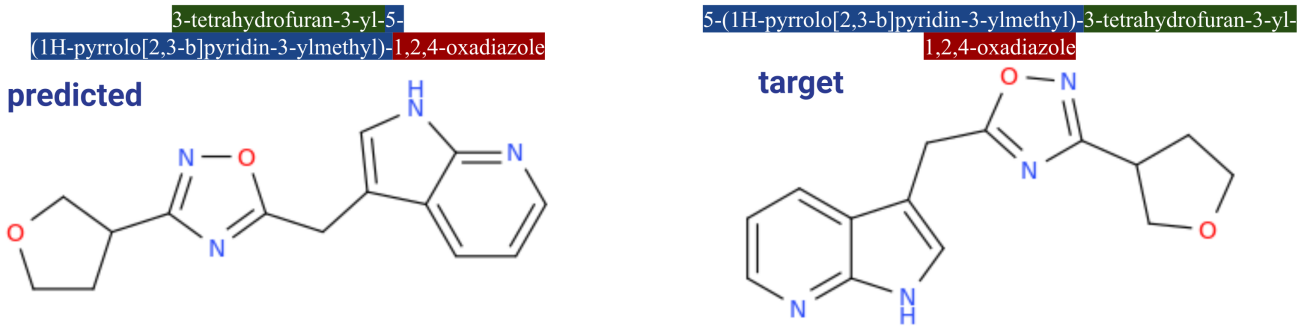

Figure 3: Left: The depicted molecule was given to the trained Transformer, and three segments of the predicted IUPAC name are highlighted in green, blue, and red. Right: The IUPAC name in the training data for the corresponding molecule is shown, where matching segments from the predicted name are highlighted with matching colors. The structure described by the predicted and target strings is identical.

binding confidence predictions to ground-truth binned assay results. Each dataset was split into 10 folds, and networks were trained 10 times to perform 10-fold cross validation. The average accuracy and AUC across the 10 runs is reported, along with the standard deviation of scores. A single standard deviation confidence interval is added for each ROC curve.

Reaction Experiments

To analyze how the learned molecular embeddings encode binding properties, we modified molecular sequences and observed changes in binding confidence to HIV-1 Protease from a binary classifier. Changes in confidence could then be compared to changes in molecular embeddings produced by the Transformer. Molecules were modified by three methods: neutralizations, deletions, and functional group swapping. The same model used in all three reaction experiments was a simple CNN composed of an input layer, two hidden convolutional layers with ReLu, and a fully connected output layer originally trained on the HIV dataset for target binding classification using a random embedding of SMILES strings. Values from the two output neurons of the CNN were modified with the softmax function to compute probabilities of non-binding and binding to HIV. Probabilities were obtained for both the original molecules and modified molecules in each reaction experiment. The change in binding confidence was then calculated by subtracting the modified and original probabilities. Chances in confidence from the classifier were then visualized against changes in dimensionality-reduced embedding space to observe how the Transformer's embedding encodes features for molecules of similar structure, and whether features useful for binding affinity prediction can be seen in unsupervised visualization of the embedding.

Neutralizations

Neutralization-capable reactions were identified as molecules containing molecular ion fragments and reacted to form neutralized products as seen in Table 3 .

Table 3: Neutralization Reaction Experiment

Reactant	Product
[n+;H]	n
[N+;!H0]	N
[$([O-]);!$([O-][#7])]	O
[S-;X1]	S
[$([N-;X2]S(=O)=O)]	N
[$([N-;X2][C,N]=C)]	N
[n-]	[nH]
[$([S-]=O)]	S
[$([N-]C=O)]	N

The molecules were "reacted" with RDKit protocol to neutralize the ion charges resulting

in a new smiles formation.

Deletions

Deletions were performed by randomly selecting one of the identified functional groups in the molecule and replacing it with the chemically correct number of hydrogens using RDkit chemical API protocol. Some of the modified molecules from this experiment resulted in separate compounds, likely do to chemical bond constraints. Thus, the resultant smiles were discarded from the visualization process. Function groups identified are provided in Supplemental Information One (SI:1).

Functional Group Swapping

Functional group swapping was performed similarly to the deletion modifications. A randomly selected functional group was replaced with another random functional group. If the substitution resulted in a chemically incorrect molecule, it was discarded from the set of evaluated molecules and excluded from visualizations. Function groups identified are provided in Supplemental Information One (SI:1).

Results

IUPAC Name Prediction

After the transformer network's stack of decoder layers, the output of the final linear transformation used to predict each character in the IUPAC name of a molecule is assessed. Due to the triviality of our translation task, we do not perform exstensive NLP evaluation of IUPAC name predictions made by the trained Transformer network. Instead, qualitative results are assessed. In general, predictions conform to the IUPAC nomenclature without error. However, for a significant minority of molecules, the IUPAC string predicted by the Transformer bears little character similarity to the matching string label in the dataset compiled from PubChem. While, the predicted and target strings are empirically different, they are simply permutations of one another which still accurately describe the same molecular structure. This phenomenon is exemplified in Figure 3

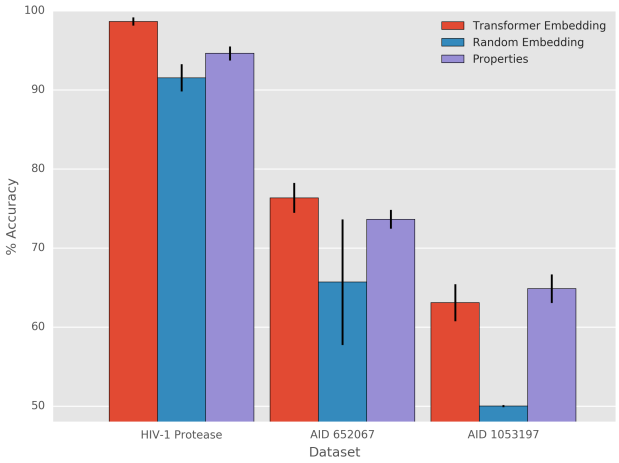

Figure 4: Percent binary classification accuracy for each dataset is shown as a bar plot. The Transformer embedding and baseline approaches are color-coded, and single standard deviation confidence intervals are represented with a vertical black bar.

Binding Prediction on Assay Datasets

ROC plots including AUC scores for the Transformer embedding and baseline classifiers on all three datasets are displayed in Figure 5. Percent accuracy from the same experiment results is displayed in Figure 4. Based on these results, the binary classification networks trained on the Transformer embedding outperform both selected baseline representations on two out of three binding prediction datasets. In the third, the Transformer embedding classifier's mean accuracy and AUC is within a standard devation confidence interval of the classifier trained on computed properties, and significantly outperforms the random embedding classifier. In general, there is large variation in the predictive ability of all classifiers depending on the dataset. It should be noted that the accuracy metric displayed here is based on a balanced ratio of binding to non-binding molecules in order to perform a fair comparison. This deviates from best practices in real-world machine learning applications, where class ratio should

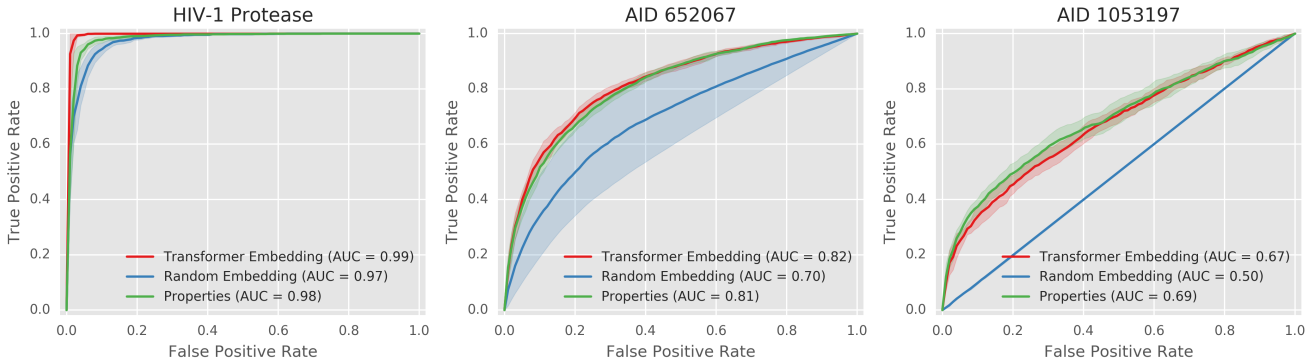

Figure 5: ROC curves for the three modified molecular datasets, with AUC displayed in the figure legends. ROC lines for the Transformer embedding and baseline approaches are color-coded and confidence intervals representing 1 standard deviation are transparently shaded in the matching color.

be considered to avoid unnecessary false positives.

Interpretation

Transformer Learns Meaningful Representation rather than Memorization

The trained Transformer network has learned the IUPAC nomenclature needed to describe the molecular structure, but exhibits poor performance if judged on its ability to output the exact character sequence from the training data. In many cases, the predicted IUPAC name for a molecule is a permutation of the IUPAC name uploaded to PubChem which describes an identical set of atoms, bonds, and groups. This phenomenon is likely due to the size of the SMILES-IUPAC pair dataset collected for training. These findings indicate the weights of the transformer network are not simply used for memorization of string pairs. Rather, the trained network has developed an internal representation of molecular structure which is distinct from the SMILES and IUPAC naming conventions.

Changes in Embedding Space Correspond with Binding Affinity

Visualizations of the reaction experiments performed are shown in Figure 6. This visualization gives insight into how properties related to binding are encoded by the transformer and suggests that the learned embedding contains information predictive of binding in addition to obvious structural properties. On the left plot for each reaction experiment the difference between original and reacted molecules in the first two principal components of embedding space indicates that structurally similar molecules are in fact represented with similar abstract features learned by the Transformer. The decrease in SMILES string length after reactions, and thus molecular weight, likely explains the tighter grouping of reaction products compared to original molecules.

When the modified molecular embeddings are compared to the original embeddings, changes in magnitude of binding confidence correspond with changes in distance and direction of the embedding space, as shown in the right images of Figure 6. This pattern is especially apparent for deletion and functional group swapping reactions, even though no supervised learning or information about molecular function has been used in the training of the embedding or the production of the figures. This finding suggests the presence of information useful for binding prediction in the latent embedding. Using these

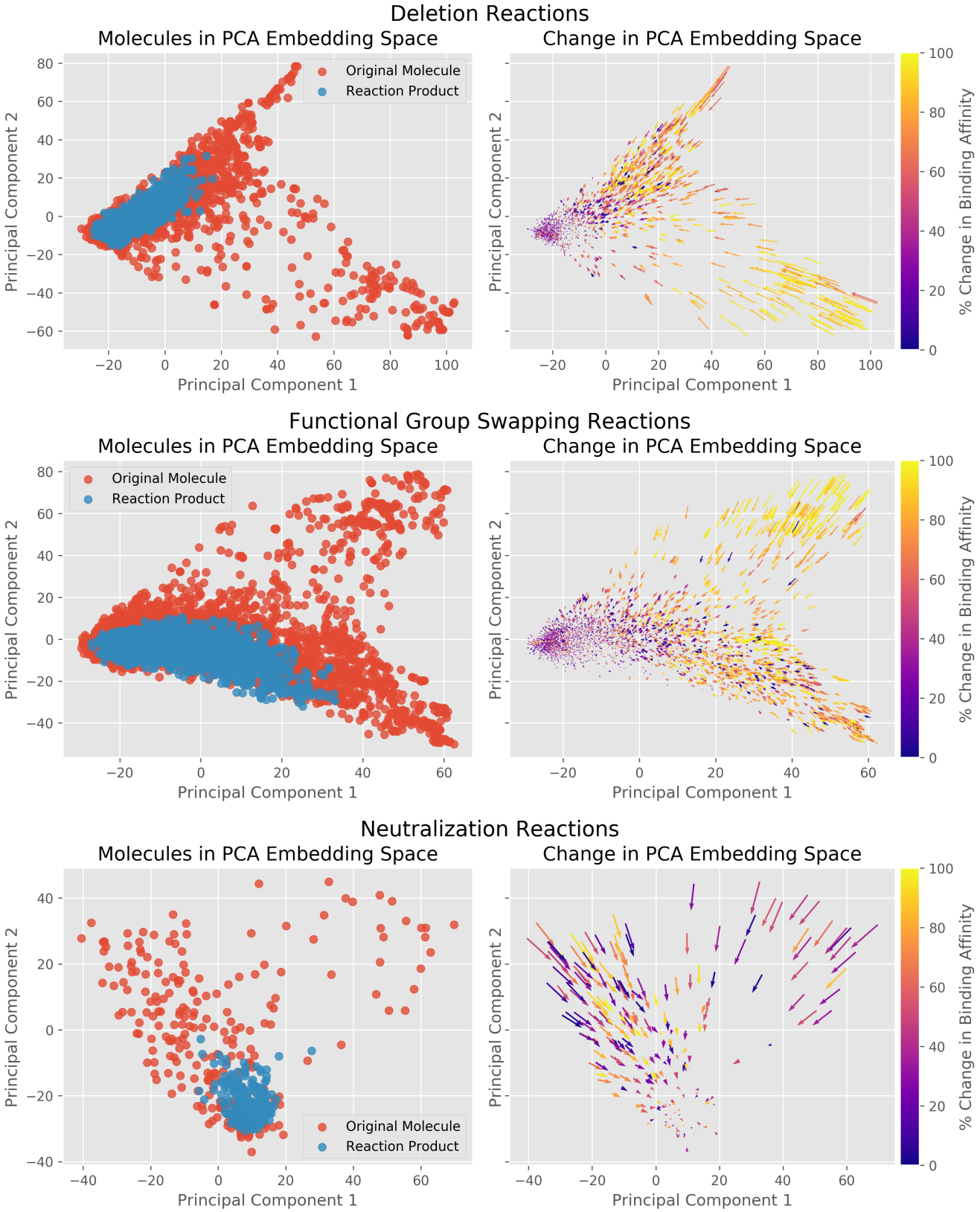

Figure 6: Change in the first two principal components embedding space after deletion reactions are used to modify a set of molecules. Binding affinity is predicted against HIV-1 Protease using a Random Embedding network. The percentile rank change in the network's binding confidence across the set of displayed molecules is color-coded, where brighter colors indicate a larger change in binding confidence after the reaction.

learned embeddings can allow for more robust virtual screening in conditions where assay data is limited, which is often the case for newly discovered targets.

Conclusion

Overall, we found that learning a mapping of chemical space via a Transformer network achieved increased accuracy of data-driven models on multiple binding affinity prediction tasks compared to models trained on hand-designed or untrained representations. In addition, we found that the abstract embedding of the trained network contains mappings of properties which are useful for predicting molecular function. While overall accuracy was somewhat limited and varied per-target, these results suggest a promising direction for further research into the application of deep learning to direct modeling of assay experiment results as a computational screening aid to existing drug discovery pipelines. Data-driven models trained on the Transformer embeddings can be applied as a quick, inexpensive computational screening method to assist the early drug discovery process for targets where a functional assay has been designed. Since the learned representation is data-driven, it also has the flexibility to be fine-tuned for simultaneous prediction of multiple properties relevant to the discovery of drug leads for a particular target. This new approach is not only an innovative computational method for conducting virtual screening, but increases the utility of previous HTS assay results for the identification of new drugs.

Acknowledgement The authors thank Dr. Karina Mayorga at the Departamento de Fisicoquímica Instituto de Química, UNAM México, D.F. for her insightful discussions on our methodology and support in sharing our research.

References

(1) Cherkasov, A.; Muratov, E. N.; Fourches, D.; Varnek, A.; Baskin, I. I.; Cronin, M.; Dearden, J.; Gramatica, P.; Martin, Y. C.; Todeschini, R., et al. QSAR modeling: where have you been? Where are you going to? *Journal of medicinal chemistry* **2014**, *57*, 4977–5010.

(2) Xu, Y.; Dai, Z.; Chen, F.; Gao, S.; Pei, J.; Lai, L. Deep learning for drug-induced liver injury. *Journal of chemical information and modeling* **2015**, *55*, 2085–2093.

(3) Cronin, M. T. Prediction of drug toxicity. *Il Farmaco* **2001**, *56*, 149–151.

(4) Åqvist, J.; Medina, C.; Samuelsson, J.-E. A new method for predicting binding affinity in computer-aided drug design. *Protein Engineering, Design and Selection* **1994**, *7*, 385–391.

(5) Goh, G. B.; Hodas, N. O.; Vishnu, A. Deep learning for computational chemistry. *Journal of computational chemistry* **2017**, *38*, 1291–1307.

(6) Kim, S. Getting the most out of PubChem for virtual screening. *Expert opinion on drug discovery* **2016**, *11*, 843–855.

(7) Mitchell, J. B. Machine learning methods in chemoinformatics. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2014**, *4*, 468–481.

(8) Mason, J. S. Computational screening: large-scale drug discovery. *Trends in Biotechnology* **1999**, *17*, 34–36.

(9) Pogany, P.; Arad, N.; Genway, S.; Pickett, S. D. De Novo Molecule Design by Translating from Reduced Graphs to SMILES. *Journal of chemical information and modeling* **2018**, *59*, 1136–1146.

(10) Lo, Y.-C.; Rensi, S. E.; Torng, W.; Altman, R. B. Machine learning in chemoinformatics and drug discovery. *Drug discovery today* **2018**, *23*, 1538–1546.

(11) LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *nature* **2015**, *521*, 436–444.

(12) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.; Thiessen, P. A.; Yu, B., et al. PubChem 2019 update: improved access to chemical data. *Nucleic acids research* **2018**, *47*, D1102–D1109.

(13) Jastrzkebski, S.; Lesniak, D.; Czarnecki, W. M. Learning to smile(s). *arXiv preprint arXiv:1602.06289* **2016**,

(14) Fooshee, D.; Mood, A.; Gutman, E.; Tavakoli, M.; Urban, G.; Liu, F.; Huynh, N.; Van Vranken, D.; Baldi, P. Deep learning for chemical reaction prediction. *Molecular Systems Design & Engineering* **2018**, *3*, 442–452.

(15) Wang, X.; Li, Z.; Jiang, M.; Wang, S.; Zhang, S.; Wei, Z. Molecule Property Prediction Based on Spatial Graph Embedding. *Journal of chemical information and modeling* **2019**, *59*, 3817–3828.

(16) Zheng, L.; Fan, J.; Mu, Y. OnionNet: a multiple-layer inter-molecular contact based convolutional neural network for protein-ligand binding affinity prediction. *arXiv preprint arXiv:1906.02418* **2019**,

(17) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *Journal of computer-aided molecular design* **2016**, *30*, 595–608.

(18) Li, R.; Wang, S.; Zhu, F.; Huang, J. Adaptive graph convolutional neural networks. Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

(19) Lusci, A.; Pollastri, G.; Baldi, P. Deep architectures and deep learning in chemoinformatics: the prediction of aqueous solubility for drug-like molecules. *Journal of chemical information and modeling* **2013**, *53*, 1563–1575.

(20) Gomes, J.; Ramsundar, B.; Feinberg, E. N.; Pande, V. S. Atomic convolutional networks for predicting protein-ligand binding affinity. *arXiv preprint arXiv:1703.10603* **2017**,

(21) Jiménez, J.; Skalic, M.; Martinez-Rosell, G.; De Fabritiis, G. K DEEP: protein–ligand absolute binding affinity prediction via 3D-convolutional neural networks. *Journal of chemical information and modeling* **2018**, *58*, 287–296.

(22) Rezaei, M.; Li, Y.; Li, X.; Li, C. Improving the Accuracy of Protein-Ligand Binding Affinity Prediction by Deep Learning Models: Benchmark and Model. **2019**,

(23) Li, Y.; Liu, Z.; Li, J.; Han, L.; Liu, J.; Zhao, Z.; Wang, R. Comparative assessment of scoring functions on an updated benchmark: 1. Compilation of the test set. *Journal of chemical information and modeling* **2014**, *54*, 1700–1716.

(24) Pereira, J. C.; Caffarena, E. R.; dos Santos, C. N. Boosting docking-based virtual screening with deep learning. *Journal of chemical information and modeling* **2016**, *56*, 2495–2506.

(25) Xu, Y.; Pei, J.; Lai, L. Deep learning based regression and multiclass models for acute oral toxicity prediction with automatic chemical feature extraction. *Journal of chemical information and modeling* **2017**, *57*, 2672–2685.

(26) 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. *Journal of chemical information and modeling* **2019**, *59*, 3370–3388.

(27) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. Advances in neural information processing systems. 2017; pp 5998–6008.

(28) Rifaioglu, A. S.; Atas, H.; Martin, M. J.; Cetin-Atalay, R.; Atalay, V.; Doğan, T. Recent applications of deep learning and

machine intelligence on in silico drug discovery: methods, tools and databases. *Briefings in bioinformatics* **2018**, *20*, 1878–1912.

(29) Liu, X.; He, P.; Chen, W.; Gao, J. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504* **2019**,

(30) Rolnick, D.; Veit, A.; Belongie, S.; Shavit, N. Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694* **2017**,

(31) An, G. The effects of adding noise during backpropagation training on a generalization performance. *Neural computation* **1996**, *8*, 643–674.

(32) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. 2014; `http://arxiv.org/abs/1412.6980`, cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

(33) Gilson, M. K.; Liu, T.; Baitaluk, M.; Nicola, G.; Hwang, L.; Chong, J. BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic acids research* **2015**, *44*, D1045–D1053.

(34) National Center for Biotechnology Information. PubChem Database. Source=The Scripps Research Institute Molecular Screening Center, AID=652067. `https://pubchem.ncbi.nlm.nih.gov/bioassay/652067(accessedonDec.31,2019)`.

(35) National Center for Biotechnology Information. PubChem Database. Source=The Scripps Research Institute Molecular Screening Center, AID=1053197. `https://pubchem.ncbi.nlm.nih.gov/bioassay/1053197(accessedonDec.31,2019)`.

(36) Srivastava, e. a., Nitish Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* **2014**, *15*, 1929–1958.

(37) Nair, V.; Hinton, G. E. Rectified linear units improve restricted boltzmann machines. Proceedings of the 27th international conference on machine learning (ICML-10). 2010; pp 807–814.

(38) Landrum, G. RDKit: Open-source cheminformatics. `http://www.rdkit.org`.

Supporting Information Available

SI #1: Functional groups for deletion and swapping reactions.

['C=O', '[H+]', '[CX3]=[OX1]', '[CX3](=[OX1])C', '[OX1]=CN', '[CX3](=[OX1])O',
'[CX3](=[OX1])[F,Cl,Br,I]', '[CX3H1](=O)[#6]', '[CX3](=[OX1])[OX2][CX3](=[OX1])',
'[NX3][CX3](=[OX1])[#6]', '[NX3][CX3]=[NX3+]', '[NX3,NX4+][CX3](=[OX1])[OX2,OX1-]',
'[CX3](=O)[OX2H1]', '[NX3][CX2]#[NX1]', '[#6][CX3](=O)[#6]', '[OD2]([#6])[#6]',
'[NX3;H2,H1;!$(NC=O)]', '[NX3,NX4+][CX4H]([*])[CX3](=[OX1])[O,N]',
"[$(*-[NX2-]-[NX2+]#[NX1]),$(*-[NX2]=[NX2+]=[NX1-])]", "[NX2]=N", '[NX3][NX3]',
'[NX3][NX2]=[*]', '[CX3;$([C]([#6])[#6]),$([CH][#6])]=[NX2][#6]', '[NX3+]=[CX3]',
'[CX3](=[OX1])[NX3H][CX3](=[OX1])',
'[$([NX3](=[OX1])(=[OX1])O),$([NX3+]([OX1-])(=[OX1])O)]', '[NX1]#[CX2]',
'[$([NX3](=O)=O),$([NX3+](=O)[O-])][!#8]', '[NX2]=[OX1]',
'[$([#7+][OX1-]),$([#7v5]=[OX1]);!$([#7](~[O])~[O]);!$([#7]=[#7])]', '[OX2H]', '[$([OH]-*=[!#6])]',
'[OX2,OX1-][OX2,OX1-]',
'[$(P(=[OX1])([$([OX2H]),$([OX1-]),$([OX2]P)])([$([OX2H]),$([OX1-]),$([OX2]P)])[$([OX2H]),$([O
X1-]),$([OX2]P)]),$([P+]([OX1-])([$([OX2H]),$([OX1-]),$([OX2]P)])([$([OX2H]),$([OX1-]),$([OX2]
P)])[$([OX2H]),$([OX1-]),$([OX2]P)])]', '[#16X2H]', '[#16X2H0]',
'[$([#16X3](=[OX1])[OX2H0]),$([#16X3+]([OX1-])[OX2H0])]',
'[$([#16X4](=[OX1])=[OX1]),$([#16X4+2]([OX1-])[OX1-])]',
'[$([#16X3]=[OX1]),$([#16X3+][OX1-])]',
'[$([#16X4](=[OX1])(=[OX1])([OX2H,OX1H0-])[OX2][#6]),$([#16X4+2]([OX1-])([OX1-])([OX2H,O
X1H0-])[OX2][#6])]',
'[$([#16X4]([NX3])(=[OX1])(=[OX1])[OX2][#6]),$([#16X4+2]([NX3])([OX1-])([OX1-])[OX2][#6])]',
'[#16X2][OX2H,OX1H0-]', '[#16X2][OX2H0]', '[CX3](=[OX1])[F,Cl,Br,I]']|