Mol2vec: Unsupervised Machine Learning Approach with Chemical Intuition

Sabrina Jaeger,[1] *Simone Fulle,*[1*] *Samo Turk*[1*]

[1] BioMed X Innovation Center, Im Neuenheimer Feld 515, 69120 Heidelberg, Germany.

[*] Corresponding authors: fulle@bio.mx, turk@bio.mx

ABSTRACT: Inspired by natural language processing techniques we here introduce Mol2vec which is an unsupervised machine learning approach to learn vector representations of molecular substructures. Similarly, to the Word2vec models where vectors of closely related words are in close proximity in the vector space, Mol2vec learns vector representations of molecular substructures that are pointing in similar directions for chemically related substructures. Compounds can finally be encoded as vectors by summing up vectors of the individual substructures and, for instance, feed into supervised machine learning approaches to predict compound properties. The underlying substructure vector embeddings are obtained by training an unsupervised machine learning approach on a so-called corpus of compounds that consists of all available chemical matter. The resulting Mol2vec model is pre-trained once, yields dense vector representations and overcomes drawbacks of common compound feature representations such as sparseness and bit collisions. The prediction capabilities are demonstrated on several compound property and bioactivity data sets and compared with results obtained for Morgan fingerprints as reference compound representation. Mol2vec can be easily combined with ProtVec, which employs the same Word2vec concept on protein sequences, resulting in a proteochemometric approach that is alignment independent and can be thus also easily used for proteins with low sequence similarities.

KEYWORDS: Machine learning, artificial neural networks, high dimensional embeddings, feature engineering

Introduction

As numeric representation of molecules is an essential part of cheminformatics, a variety of descriptors and molecular fingerprints (FP) exists which are either fed into machine learning (ML) models or form the basis for similarity searching and clustering approaches. Most commonly used representations include Morgan FPs (also known as extended-connectivity fingerprints (ECFP))[1] as they often outperform other types of FPs in similarity search and virtual screening tasks[2,3] and are also successfully used for molecular activity predictions.[4–7] To generate a Morgan FP, all substructures around all heavy atoms of a molecule within a defined radius are generated and assigned to a unique identifier (called Morgen identifier below). These identifiers are then usually hashed to a vector with fixed length. However, the vectors obtained are very high-dimensional and sparse, and on top of that might also contain bit collisions introduced by the hashing step.

The recent rise in popularity of artificial neural networks brought several breakthroughs in ML and ideas from various fields of data science are also spilling over to cheminformatics. Convolutional neural networks, originally developed for image recognition, were successfully applied on molecular graphs[8,9] and on 2D depictions of molecules.[10] In parallel, natural language processing (NLP) techniques were adopted to learn from classical features, like molecular FPs,[11] SMILES strings,[12] and graph representations of compounds.[8] Most worth noting, the NLP method "term frequency-inverse document frequency" (tf-idf) was applied on Morgan fingerprints for compound-protein prediction[11] and the "Latent Dirichlet Allocation" method for chemical topic modeling.[13] Another popular NLP approach is Word2vec[14] which learns high dimensional embeddings of words where vectors of similar words end up near in vector space. This concept was already adopted to protein sequences (ProtVec) for the classification of protein families and disordered proteins[15] but was not applied to molecules so far.

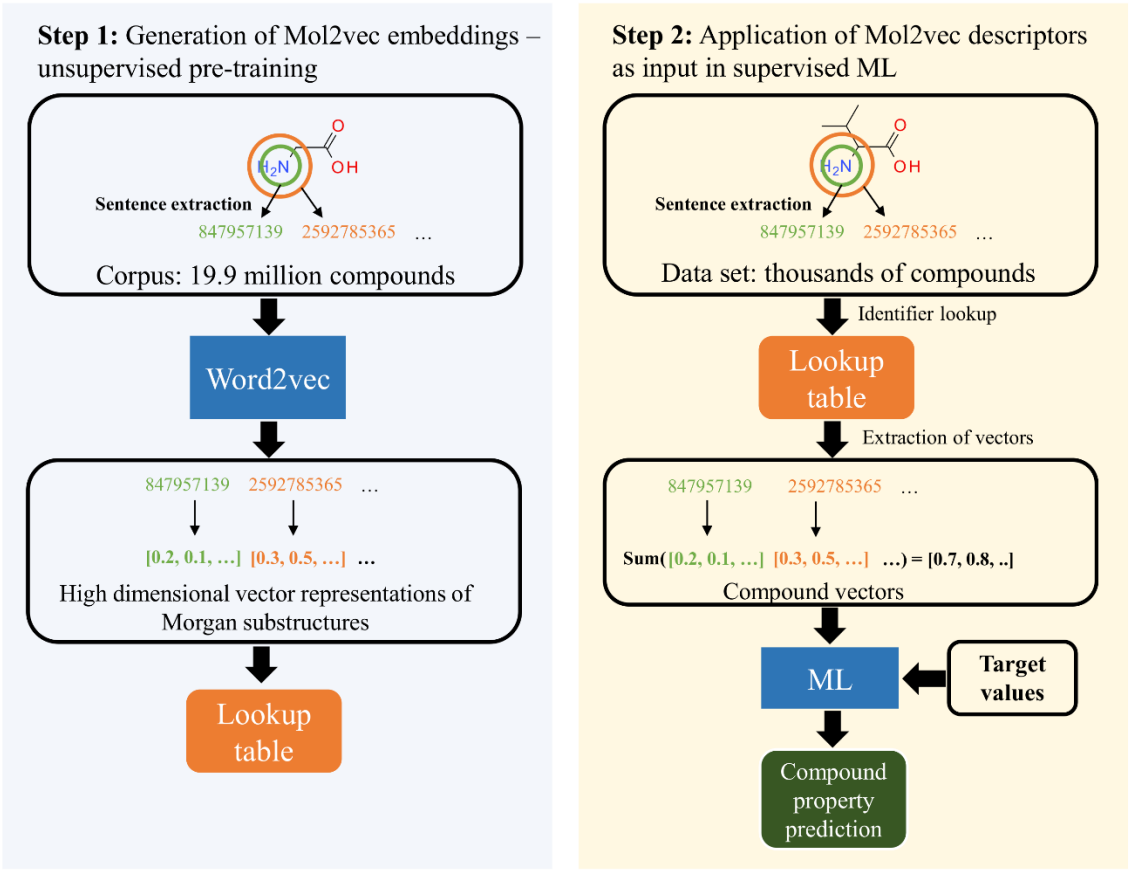

Figure 1. Overview of the generation and usage steps of Mol2vec. Step 1: Generation of Mol2vec embeddings (i.e. vector representations of substructures) via an unsupervised pre-training step. Step 2: The application of Mol2vec vectors requires that substructure vectors are retrieved and summed up to obtain compound vectors, which can finally be used to train a supervised prediction model.

Here, we introduce Mol2vec, which is an NLP inspired technique that considers compound substructures derived from the Morgan algorithm as "*words*" and compounds as "*sentences*". By applying the Word2vec algorithm on a corpus of compounds, high-dimensional embeddings of substructures are obtained, where the vectors for chemically related substructures occupy the same part of vector space. Mol2vec is an unsupervised method which is initially trained on unlabeled data to obtain feature vectors of substructures which can be summed up to obtain compound vectors. Please note that while the generation of a Mol2vec

model is an unsupervised pre-training step, subsequent machine learning models for property predictions are supervised throughout the manuscript (Figure 1). Questions addressed below are how Mol2vec performs on different compound data sets, on regression and classification problems, and combined with the ProtVec representation for proteins in proteochemometric (PCM) approaches on proteins with different sequence similarities ranges.

Materials and Methods

Mol2vec and ProtVec are unsupervised pre-training methods that can be used to obtain high dimensional embeddings of molecular substructures or n-grams of protein sequences (i.e. they provide featurization of compounds and proteins). These vectors can then be further used in supervised ML tasks. In this section, we first describe the data sets used for the pre-training of the Mol2vec and ProtVec models and the pre-training itself, followed by the data sets used for the evaluation of Mol2vec in supervised tasks and the employed machine learning methods for property predictions.

Pre-training compound data set. The corpus of compounds was composed using the ZINC v15[16] and ChEMBL v23[17,18] databases as source of compounds. The two databases were merged, duplicates removed, only compounds kept that could be processed by RDKit, and filtered using the following cutoffs and criteria: molecular weight between 12 and 600, heavy atom count between 3 and 50, clogP between -5 and 7, and only H, B, C, N, O, F, P, S, Cl, Br atoms allowed. Additionally, all counter ions and solvents were removed and canonical SMILES generated by RDKit.[19] This procedure yielded 19.9 million compounds.

Compound encoding and Mol2vec model. In an NLP analogous fashion, molecules were considered as sentences and substructures as words. To obtain words for each molecule, the Morgan algorithm [1] was used to generate all atom identifiers at radii 0 and 1, resulting into 119 and 19831 identifiers, respectively. Identifiers of each atom (radius 0 followed by radius 1

each) were then ordered into a sentence with the same atom order as present in the canonical SMILES representation (Figure 2).

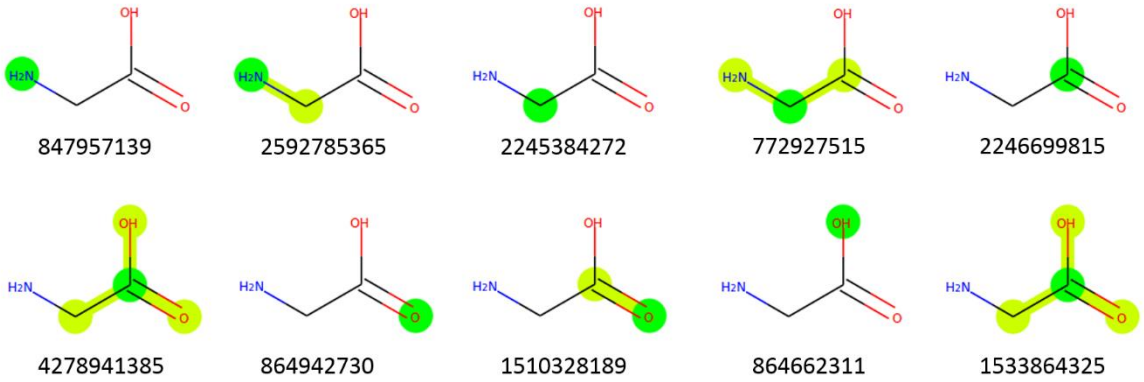

Figure 2. Depiction of identifiers obtained with the Morgan algorithm on the structure of glycine forming a Mol2vec sentence. Identifiers are ordered in the same order as the atoms in the canonical SMILES. If an atom has more than one identifier, the first identifier for that atom is the one for radius 0, followed by radius 1, etc.

The Mol2vec model was trained utilizing all 19.9 million compounds in the corpus and using the gensim[20] implementation of Word2vec, which is a shallow, two-layer neural network. Although Word2vec is an unsupervised method it still internally defines an auxiliary prediction task. Depending on the auxiliary task Word2vec can be trained using one of the following two approaches: 1) continuous bag-of-words (CBOW) if the task is to predict a word from the context words, and 2) Skip-gram if the context is predicted based on a word. In CBOW the order of words in the context is not important due to the bag-of-words assumption, while in skip-gram adjacent words are assigned with higher weights. Furthermore, the two parameters "window size" and "dimensional embeddings" were explored to find the best settings for Mol2vec. The window size is controlling the size of the context and was set to the in NLP commonly used sizes of 5 and 10 in the case of CBOW and Skip-gram, respectively.

Furthermore, to account for the fact that each atom is represented twice via Morgan identifiers (i.e. at radius 0 and 1), the effect of double window sizes (i.e. 10 for CBOW and 20 for Skip-gram) was also evaluated. Finally, 100 and 300-dimensional embeddings were generated for all combinations.

All rare words that occur less than three times in the corpus were replaced with a string "UNSEEN", because 1) Word2vec is not able to get meaningful embeddings for rare words and 2) this enables the model to gracefully handle unseen (or unknown) words that might appear when performing predictions on new data. The distribution of "UNSEEN" in the corpus is random and hence a vector close to zero is usually learned. If an unknown identifier occurs during featurization of the new data, the "UNSEEN" vector is used to embed it. The vector for a molecule is finally obtained by summing up all vectors of the Morgan substructures of this molecule.

ProtVec model. The protein corpus of 554,241 sequences was collected from UniProt.[21] The protein sequences were afterwards featurized using the ProtVec[15] approach. All possible words were generated by representing each sequence in the corpus as 3 sequence variants (i.e. sentences) that are each shifted by one amino acid, followed by the generation of all possible 3-grams (words) (Figure 3). This yielded in 1,662,723 sentences for the protein corpus.

ProtVec model was trained with the gensim Word2vec implementation, using a Skip-gram architecture with a window size of 25 and output vector size of 300. To handle potentially new 3-grams the model was trained on "UNSEEN" words in a similar way as the Mol2vec models. The final model resulted in high-dimensional embeddings of 9,154 unique 3-grams.

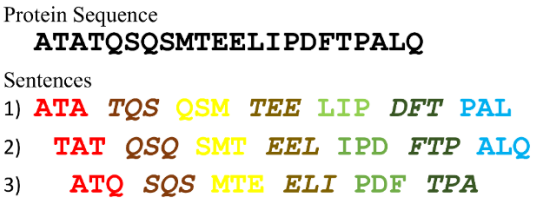

Figure 3. Protein sequence processing. Each sequence is represented as *n* sequences (i.e. sentences) with shifted reading frame and split in *n*-grams (i.e. words), with $n = 3$.

PCM vectors. For the PCM approach Mol2vec was combined with ProtVec by concatenating both vectors (called PCM2vec below). Baseline PCM vectors were concatenated Morgan FPs (2048 bits) and z-scales [22] which are sequence-based physicochemical protein descriptors. Since the use of z-scales relies on a sequence alignment they were only used for the kinase data set. Following the study described in ref. [7], kinase sequences were aligned and z-scales (Z3) calculated only for the 85 binding site residues defined in KLIFS.[23] The length of the target descriptor was adjusted to 2048 using a WTA-hash function to match the dimensionality of the Morgan FP. [24]

Benchmarking data sets. The performance of Mol2vec in subsequent ML models were evaluated using the ESOL, Ames, and Tox21 data sets as well as one curated kinase data set:

- *ESOL solubility data set*[25] was chosen to evaluate the performance of Mol2vec in a regression task to predict aqueous solubility of 1144 compounds.
- *Ames mutagenicity data set*[26] contains 6511 compounds that were determined to be either mutagenic (3481) or non-mutagenic (2990), and thus represent a balanced data set for classification.
- *Tox21 data set*[27] consists out of 12 targets which were associated with human toxicity and contains a total of 8192 compounds. Tox21 was retrieved as a part of the DeepChem package[28] to enable a comparison with established methods.
- *Kinase data set.* A kinase data set was compiled using ChEMBL v23 and evaluated with respect to classification tasks.[18] Bioactivities for 284 kinases (list see Supporting

Information) were extracted and filtered to keep only IC_{50}, *Kd* and *Ki* values from binding assays and with a target confidence of at least 8. Bioactivities were converted to pIC_{50} and an activity threshold of 6.3 was employed.

Validation of models based on Mol2Vec vectors. All machine learning models using on compound data were trained using 20x 5-fold cross validation and compared using the Wilcoxon signed rank test. Employed performance metrics are for the regression tasks: coefficient of determination (R^2_{ext}), mean absolute error (MAE), and mean squared error (MSE), and for classification tasks: area under the curve (AUC) of receiver operating characteristic curve (ROC), sensitivity (i.e. true positive rate) and specificity (i.e. true negative rate). Compounds in all data sets were processed using RDKit to remove compounds with less than 3 heavy atoms, to remove all salts (i.e. counter ions) and solvents, and to generate canonical SMILES. Compounds were encoded as vectors (featurization) by summing up vectors of Morgan substructures retrieved from the pre-trained Mol2vec model. Morgan FPs with radius 2 and hashed to 4096 (2048 for PCM experiments due to memory constraints) bits were used as baseline features to train fingerprint based ML models.

A PCM approach was evaluated for the two data sets with several targets (i.e. Tox21 and kinase bioactivities) to assess the influence of adding protein information by concatenating compound descriptors (Morgan FP or Mol2vec) with protein descriptors (Z-scales or ProtVec). ProtVec descriptors for the proteins in the Tox21 (List S1) and kinase (List S2) data sets were calculated based on the entire protein and catalytic domain, respectively. The performance of the PCM models was evaluated by a rigorous 4-level (CV1-CV4) validation scheme.[7] Briefly, CV1 tests the model performance on new compound-target pairs, CV2 on new targets, CV3 on new compounds and CV4 on by the model new compounds and targets.

Machine learning methods. Three different machine learning methods (i.e. Random forest (RF), Gradient Boosting Machine (GBM) and Deep Neural Network (DNN)) were evaluated using Mol2vec embeddings as compound features. RF implementation in scikit-learn[29] was used with 500 estimators, square root of number of features as maximum number of features, and balanced class weight. The XGBoost implementation[30] of GBM was used with 2000 estimators and setting maximum depth of trees to 3 and learning rate to 0.1. For the GBM classifier the weight of the positive samples was adjusted to reflect the ratio of actives/inactives in the respective data set. Several feed-forward DNNs were built using Keras[31] with the TensorFlow[32] backend. After an initial benchmarking, variations of two different DNNs architectures were used based on the input data and prediction task. 1) DNNs trained with Morgan FPs (radius 2) had one hidden layer with 512 neurons each and an output layer with one neuron. All layers had normal initialization and employed rectified linear unit (ReLU)[33] activation function except for the output neuron in the case of classification tasks which employed a sigmoid activation function. Adam optimizer[34] was used to minimize Poisson loss function for classification and to minimize mean squared error (MSE) for regression. 2) DNNs trained with Mol2vec embeddings had 4 hidden layers with 2000 neurons each and one output neuron. All layers had normal initialization and employed ReLU activation function except for the output neuron in the case of classification tasks which again employed a sigmoid activation function. Adamax optimizer[34] was used to minimize binary cross entropy loss function for classification and to minimize MSE for regression. All DNNs used a dropout value of 0.1 to avoid overfitting.[35] In the case of the DNN classifiers, the weight of the actives was adjusted to reflect the imbalance in the data.

Results and discussions

Mol2vec is an unsupervised pre-training method to generate an information rich representation of molecular substructures. Since it is an unsupervised method, it does not require labeled data as input and can leverage from larger amounts like the here employed 19.9 million compounds. The obtained embeddings from the pre-training can be used for instance to explore the relationships between different substructures, while derived compound vectors can be used for assessing compound similarity or as features for supervised ML predictions.

Mol2vec training and hyperparameter evaluation. The evaluation of different parameters for Mol2vec revealed that the best settings are overlapping with those recommended for Word2vec in NLP on text data and comprise the Skip-gram model with a window size of 10 and 300-dimensional embeddings of Morgan substructures. The quality of the individual embeddings was assessed by using them as features in supervised ML tasks on the Tox21 data set (Supporting Information Table S1). As it was observed for NLP applications,[14] also in our case Skip-gram yielded higher performing embeddings compared to CBOW, possibly because it captures spatial relationships better due to the weighting of words in the context. Higher dimensionality of embeddings also had beneficial effect on the performance while varying the window size had almost no effect. The final Mol2vec model was trained on a corpus of 19.9 million molecules.

Chemical intuition of Mol2vec descriptors. A key assumption of the Mol2vec approach is that related functional groups and molecules are close in the generated vector space. This was visually investigated as well as quantified by extracting the 25 most common substructures from the compound corpus as well as featurizing standard amino acids via Mol2vec descriptors. Encouragingly, the Mol2vec vectors of the 25 most common substructures cluster in expected relationships (Figure 4). Aromatic carbon types are correctly identified to be chemically related (red identifiers) as well as aliphatic carbons in ring systems (purple), non-ring aliphatic carbons

(green), and carbonyl carbon and oxygen (turquoise). Further interesting relationships could be explored by looking at more substructures concurrently (not shown).

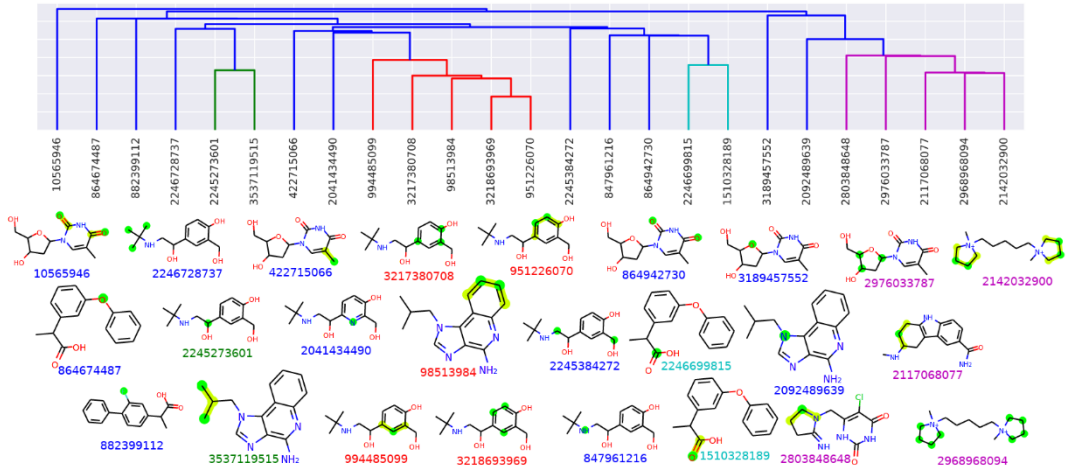

Figure 4. Dendrogram showing relationships between vectors representing the 25 most common substructures in the compound corpus. Substructures are depicted (central atoms in green and surrounding atoms in light green) on a representative compound from a pool of FDA approved drugs.

Similarly, also 2D projections of vector representations obtained for the 20 proteinogenic amino acids agree with chemical intuition and capture the similarity between related amino acids (Figure 5; Table S2). For instance, Pro is an obvious outlier while other amino acids are nicely grouped based on their functional groups and properties. Also interesting is that the transition distance between Glu and Gln is similar to the distance between Asp and Asn, which is line with the underlying change of the carboxylic acid group to an amide.

Next, the prediction capabilities of Mol2vec vectors are demonstrated on several compound property and activity data sets and compared with results obtained for the Morgan fingerprints as reference compound representation.

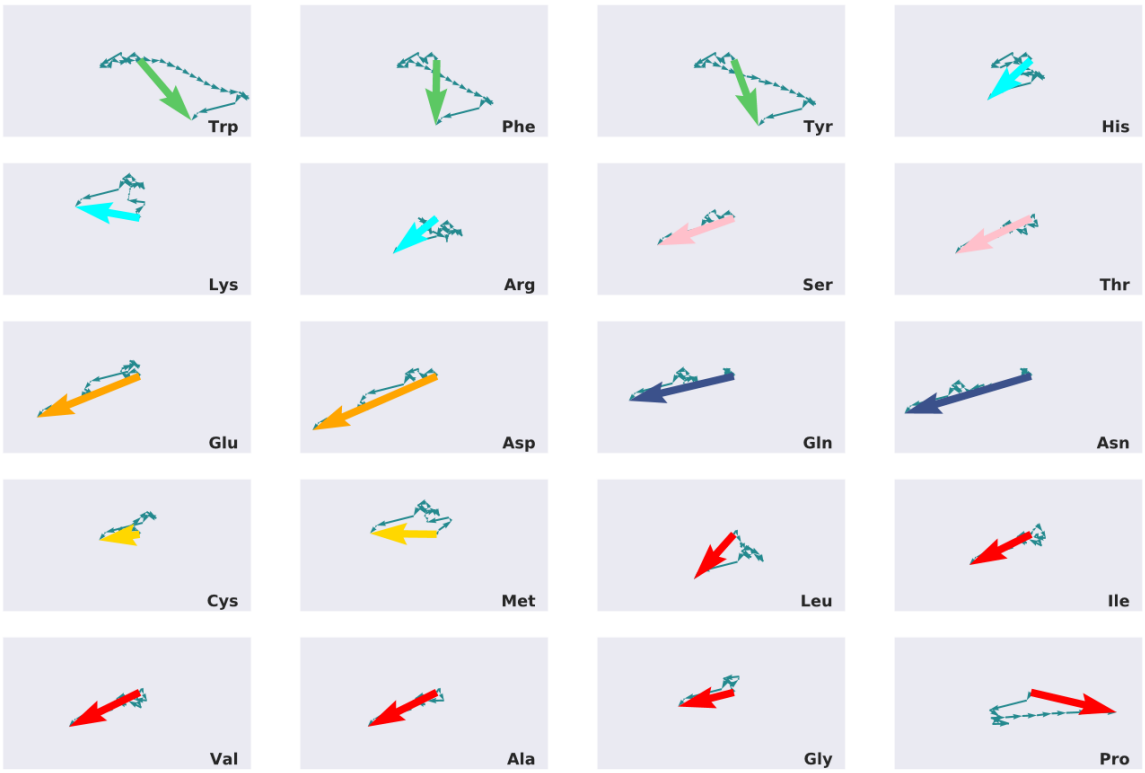

Figure 5. 2D projection (t-SNE) of Mol2vec vectors of amino acids (bold arrows). Vectors were obtained by summing up the vectors of the Morgan substructures (small arrows) present in the respective molecule (amino acids in the present example). The directions of the vectors provide a visual representation of similarities. A quantitative assessment can be obtained via cosine angle calculations (Supporting Information Table S2).

Mol2vec compound vectors as features for supervised ML tasks. Employing Mol2vec vectors as input for different ML tasks (Figure 1), such as classification and regression, and on a variety of data sets indicated overall that Mol2vec vectors yield state-of-the-art performance. The combination with GBM seems to be very suitable for regression task (i.e. Ames data set) while the combination with RF can be recommended for classification tasks including proteochemometric (PCM) approaches.

Although several DNN architectures were evaluated, they were still outperformed by the tree based methods GBM and RF. However, we would like to note that further fine-tuning might

improve the prediction power of the Mol2vec-DNN combination. See Tables S3-S5 for detailed performance numbers for all employed methods. In the following, the best predictions obtained for the Mol2vec vectors are described more in detail and compared with results obtained for the Morgan FP as baseline descriptor as well as results described in the literature for the employed data sets.

ESOL solubility data set was selected to test Mol2vec in a regression task (Table 1). Mol2vec yields better predictions (R^2_{ext} = 0.86) than the originally reported multiple linear regression (MLR) model[25] as well as a molecular graph convolution method,[36] and importantly outperforms our Morgan FP based baseline model (R^2_{ext} = 0.66). However, the best reported results on the ESOL data set were so far obtained by two molecular graph convolution[9,37] and one recurrent network based[38] methods ($R^2_{ext} \approx 0.93$).

Table 1. Performance of Mol2vec and other models on regression predictions of the ESOL data set.

ML Features	ML Method	R^2_{ext}	MSE	MAE	Ref.
Descriptors	MLR	0.81 ± 0.01	0.82	0.69	[25]
Molecular Graph	CNN	0.82	-	-	[36]
Molecular Graph	CNN	-	-	0.52 ± 0.07	[37]
Molecular Graph	CNN	0.93	0.31 ± 0.03	0.40 ± 0.00	[9]
Molecular Graph	RNN	0.92 ± 0.01	0.35	0.43	[38]
Morgan FP	GBM	0.66 ± 0.00	1.43 ± 0.00	0.88 ± 0.00	
Mol2vec	GBM	0.86 ± 0.00	0.62 ± 0.00	0.60 ± 0.00	

Ames benchmarking data set is a classic toxicological data set used to benchmark various classification ML methods. Here, Mol2vec and Morgan FP result in equally good predictions and are in line with AUC results reported for a SVM model and a Naïve Bayes Classifier (NBC)[39] on this data set[26] but the former two achieved higher sensitivity values (Table 2).

Table 2. Performance of Mol2vec and other methods on classification prediction of the Ames data set.

ML Features	ML Method	AUC	Sensitivity	Specificity	Ref.
Descriptors	SVM	0.86 ± 0.01	-	-	[26]
Descriptors + Morgan FP	NBC	0.84 ± 0.01	0.74 ± 0.02	0.81 ± 0.01	[39]
Morgan FP	RF	0.88 ± 0.00	0.82 ± 0.00	0.80 ± 0.01	
Mol2vec	RF	0.87 ± 0.00	0.80 ± 0.01	0.80 ± 0.01	

The Tox21 data set represents a challenging classification data set covering 12 targets and over 8000 compounds with unbalanced classes. Here, Mol2vec and Morgan FP result in equally good predictions (i.e. both obtained average AUC values of 0.83) and this time outperform existing literature results (Table 3, Table S6).

Table 3. Performance of Mol2vec and other methods on classification predictions of the Tox21 data set.

ML Features	ML Method	AUC	Sensitivity	Specificity	Ref.
Molecular graph	CNN	0.71 ± 0.13	-	-	[9]
Molecular descriptors and FPs	SVM	0.71 ± 0.13	-	-	[5]
Molecular descriptors and FPs	DNN	0.72 ± 0.13	-	-	[5]
Morgan FP	RF	0.83 ± 0.05	0.28 ± 0.14	0.99 ± 0.01	
Mol2vec	RF	0.83 ± 0.05	0.20 ± 0.15	1.00 ± 0.01	

Overall, Mol2vec descriptors shows competitive performance compared to the classic Morgan FP on the employed benchmarking data sets for classification (i.e. Ames and Tox21) and outperformed the Morgan FP on the regression predictions for the ESOL data set. Morgan

FP and Mol2vec features are both based on identifiers calculated by the Morgan algorithm. While these identifiers are hashed to a binary fingerprint in the case of the Morgan FP, they form a vector with continuous values in the case of the Mol2vec approach. Since the final Mol2vec vector is a sum of substructure vectors, it implicitly captures substructure counts as well as substructure importance via the vector amplitude. In addition, Mol2vec also has the advantage of lower dimensionality of final vectors which significantly speeds up the training and lowers memory requirements. Further tuning of the Mol2vec approach, for example by using Morgan identifiers with bigger radii might further improve the prediction performance.

Mol2vec compound and ProtVec protein vectors as features for PCM. Earlier studies using the PCM approach, where compound and protein descriptors are employed as concatenated features, indicate that the additional use of protein information can improve the prediction quality.[40] To test the benefit of Mol2vec for PCM applications, Mol2vec vectors were coupled with ProtVec vectors. Such PCM2vec models (Table 4-5) were compared with results obtained for Morgan FP, Mol2vec, and in the case of the kinase data set also with a classical PCM approach (i.e. Morgan FP for compounds combined with z-scales for proteins). In each scenario one model was trained on the entire training data for several targets allowing to quantify the benefit of protein descriptors. Several ML methods were evaluated (Tables S7 and S8) of which RF yielded overall the best results and employed for the results below.

So far existing PCM models required sequence alignments and were thus build for conserved target classes such as kinases and GPCRs.[40] Thus, it is worth noting that ProtVec is alignment independent and can thus not only be directly applied on the kinase data set (Table 5), but also on the Tox21 data set (Table 4) which consists of unrelated proteins with low sequence similarity. On both data sets, PCM2vec outperforms the compound features in CV1 and CV3 which indicates that the added protein information improves the extrapolation to new compound-target pairs and new compounds (Table 4-5). In CV2, the here employed PCM2vec

approach performs slightly worse than Morgan FP and Mol2vec on the Tox21 data set and performs equally well on the kinase data set. For CV4, PCM2vec achieves compared to the compound fingerprints similar performance on the Tox21 data and better predictions on the kinase data.

Table 4. Summary of prediction performance of PCM models in comparison to compound features on Tox21.

Validation Level	ML Features	AUC	Sensitivity	Specificity
CV1	Morgan FP	0.79 ± 0.01	0.73 ± 0.01	0.74 ± 0.00
	Mol2vec	0.78 ± 0.01	0.73 ± 0.00	0.72 ± 0.02
	PCM2vec	**0.87 ± 0.01**	**0.80 ± 0.01**	**0.79 ± 0.01**
CV2	Morgan FP	0.73 ± 0.07	0.63 ± 0.08	0.71 ± 0.03
	Mol2vec	0.72 ± 0.07	0.65 ± 0.09	0.68 ± 0.04
	PCM2vec	0.70 ± 0.04	0.55 ± 0.02	0.69 ± 0.04
CV3	Morgan FP	0.78 ± 0.02	0.65 ± 0.03	0.77 ± 0.02
	Mol2vec	0.79 ± 0.01	0.70 ± 0.02	0.74 ± 0.01
	PCM2vec	**0.85 ± 0.01**	**0.75 ± 0.01**	**0.80 ± 0.01**
CV4	Morgan FP	0.76 ± 0.03	0.59 ± 0.06	0.77 ± 0.05
	Mol2vec	0.73 ± 0.06	0.62 ± 0.10	0.74 ± 0.05
	PCM2vec	0.75 ± 0.02	0.61 ± 0.12	0.73 ± 0.11

Validation levels - CV1: new compound-target pairs, CV2: new targets, CV3: new compounds, and CV4: new compounds and targets. Highlighted cases mark the validation levels where PCM2vec outperforms the ML models based on compound features only.

Since kinases share high sequence similarity, Morgan FP + z-scales was added as baseline PCM approach when evaluating the impact of the employed features on the prediction of kinase data set. The comparison of PCM2vec with a classical PCM model for kinases (i.e. Morgan + z) revealed that the latter yield equally good results in CV1 and CV2 (i.e. new compound-target pairs and new targets) and slightly better results in CV3 and CV4 (i.e. new compounds, and new compounds and targets). One reason for the better prediction of the

classical PCM models might be that it considers binding site residues only, while ProtVec was build based on the entire kinase domain. However, although there is performance difference between PCM2vec and Morgan FP + z-scales, practically PCM2vec yields in the present case more balanced models with roughly equal sensitivity and specificity (i.e. in CV2 and CV4). Furthermore, it can be also directly applied on distant protein classes such as the Tox21 data set, resulting in general into better predictions for new compound + target pairs as well compounds compared to using only compound based features.

Table 5. Summary of prediction performance of PCM models in comparison to compound features on kinase data set.

Validation level	ML Features	AUC	Sensitivity	Specificity
CV1	Morgan FP	0.91 ± 0.00	0.82 ± 0.00	0.85 ± 0.00
	Mol2vec	0.91 ± 0.00	0.83 ± 0.00	0.84 ± 0.00
	Morgan + z	**0.96 ± 0.00**	**0.89 ± 0.00**	**0.90 ± 0.00**
	PCM2vec	**0.95 ± 0.00**	**0.89 ± 0.00**	**0.87 ± 0.00**
CV2	Morgan FP	0.88 ± 0.00	0.76 ± 0.01	0.85 ± 0.01
	Mol2vec	0.89 ± 0.00	0.80 ± 0.01	0.83 ± 0.00
	Morgan + z	0.89 ± 0.00	0.37 ± 0.03	0.96 ± 0.00
	PCM2vec	0.89 ± 0.00	0.65 ± 0.01	0.90 ± 0.01
CV3	Morgan FP	0.82 ± 0.00	0.94 ± 0.00	0.41 ± 0.01
	Mol2vec	0.78 ± 0.01	0.97 ± 0.00	0.24 ± 0.01
	Morgan + z	**0.94 ± 0.00**	**0.86 ± 0.01**	**0.89 ± 0.00**
	PCM2vec	**0.91 ± 0.00**	**0.92 ± 0.01**	**0.63 ± 0.02**
CV4	Morgan FP	0.74 ± 0.02	0.87 ± 0.02	0.43 ± 0.02
	Mol2vec	0.73 ± 0.02	0.94 ± 0.01	0.26 ± 0.03
	Morgan + z	**0.84 ± 0.02**	**0.35 ± 0.04**	**0.96 ± 0.01**
	PCM2vec	**0.77 ± 0.02**	**0.68 ± 0.04**	**0.72 ± 0.02**

Highlighted cases mark the validation levels where PCM models (i.e. Morgan + z and PCM2vec) outperforms the ML models based on compound features only (i.e. Morgan FP and Mol2vec).

Conclusion

Inspired by natural language processing (NLP) techniques, Mol2vec represents a novel way of embedding compound substructures as information rich vectors. The substructures were extracted in the present study by employing the Morgan algorithm and, in the context of NLP, represent words while the complete molecules are sentences. To describe new compounds substructure vectors from a pre-trained Mol2vec model are retrieved and summed up.

The Mol2vec model itself is an unsupervised pre-training method that is trained on all available chemical structures, yielding high quality embeddings of molecules. Results on common substructures as well as amino acids nicely illustrate that the derived substructure vectors of chemically related substructures and compounds occupy similar vector space. This result is not unexpected since Word2vec vectors representing similar words also end up near in vector space. Substructure vectors can be simply summed up to obtain compound vectors which can be used to calculate compound similarity or as features in supervised ML tasks. A thorough evaluation of Mol2vec on different chemical data sets showed that it can achieve state-of-the-art performance and compared to the Morgan FP fingerprint seems to be especially suited for regression tasks. Additionally, Mol2vec combined with ProtVec (i.e. PCM2vec) performs well in proteochemometrics approaches, and can be directly applied for data sets with unrelated targets with low sequence similarities.

ASSOCIATED CONTENT

Supporting Information.

Table S1: Evaluation of different Word2vec hyperparameters; Table S2: Pairwise cosine distance of Mol2vec vectors of amino acids; Table S3: Evaluation of different features and machine learning methods on ESOL data set; Table S4: Evaluation of different features and machine learning methods on Ames data set; Table S5: Evaluation of different features and machine learning methods on Tox21 data set; Table S6: Results of machine learning predictions on individual Tox21 targets; Table S7: Evaluation of different machine learning methods in PCM approach on Tox21 data set; Table S8: Evaluation of different machine learning methods in PCM approach on kinase data set; List S1: Tox21 assays used for PCM models; List S2: Kinases in the kinase data set.

AUTHOR INFORMATION

Corresponding Authors

* Simone Fulle (fulle@bio.mx) and Samo Turk (turk@bio.mx)

ORCID

Simone Fulle: 0000-0002-7646-5889

Samo Turk: 0000-0003-2044-7670

Author Contributions

The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

ACKNOWLEDGMENT

We thank Benjamin Merget for fruitful discussions and Katra Kolšek for feedback on readability of the manuscript.

ABBREVIATIONS

AUC, Area Under Curve; CNN, Convolutional Neural Networks; DNN, Deep Neural Networks; FP, Fingerprint; GBM, Gradient Boosting Machines; MAE, Mean Absolute Error; MLR, multiple linear regression; MSE, Mean Squared Error; PCM, proteochemometrics; ReLU, Rectified Linear Unit. RF, Random Forest; RNN, Recurrent Neural Networks; ROC, Receiver Operating Characteristic; SVM, Support Vector Machines; XGB, Extreme Gradient Boosting.

REFERENCES

(1) Rogers, D.; Hahn, M. Extended-Connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50* (5), 742–754.

(2) Riniker, S.; Landrum, G. A. Open-Source Platform to Benchmark Fingerprints for Ligand-Based Virtual Screening. *J Cheminform* **2013**, *5*, 26.

(3) O'Boyle, N. M.; Sayle, R. A.; O'Boyle, N. M.; Sayle, R. A. Comparing Structural Fingerprints Using a Literature-Based Similarity Benchmark. *J Cheminform* **2016**, *8* (1), 36.

(4) Riniker, S.; Fechner, N.; Landrum, G. A. Heterogeneous Classifier Fusion for Ligand-Based Virtual Screening: Or, How Decision Making by Committee Can Be a Good Thing. *J. Chem. Inf. Model.* **2013**, *53* (11), 2829–2836.

(5) Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S.; Mayr, A.; Klambauer, G.; Hochreiter, S. DeepTox: Toxicity Prediction Using Deep Learning. *Front. Environ. Sci.* **2015**, *3* (80), 1–15.

(6) Merget, B.; Turk, S.; Eid, S.; Rippmann, F.; Fulle, S. Profiling Prediction of Kinase Inhibitors: Towards the Virtual Assay. *J. Med. Chem.* **2017**, *60* (1), 474–485.

(7) Sorgenfrei, F. A.; Fulle, S.; Merget, B. Kinome-Wide Profiling Prediction of Small Molecules. *ChemMedChem* **2017**, cmdc.201700180.

(8) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving beyond Fingerprints. *J. Comput. Aided. Mol. Des.* **2016**, *30* (8), 595–608.

(9) Coley, C. W.; Barzilay, R.; Green, W. H.; Jaakkola, T. S.; Jensen, K. F. Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction. *J. Chem. Inf. Model.* **2017**, acs.jcim.6b00601.

(10) Goh, G. B.; Siegel, C.; Vishnu, A.; Hodas, N. O.; Baker, N. Chemception: A Deep Neural Network with Minimal Chemistry Knowledge Matches the Performance of Expert-Developed QSAR/QSPR Models. *CoRR* **2017**, *abs/1706.0*.

(11) Wan, F.; Zeng, J. Deep Learning with Feature Embedding for Compound-Protein Interaction Prediction. *bioRxiv* **2016**, 0–20.

(12) Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-Novo Design through Deep Reinforcement Learning. *J. Cheminform.* **2017**, *9* (1), 48.

(13) Schneider, N.; Fechner, N.; Landrum, G. A.; Stiefl, N. Chemical Topic Modeling: Exploring Molecular Data Sets Using a Common Text-Mining Approach. *J. Chem. Inf. Model.* **2017**, acs.jcim.7b00249.

(14) Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. **2013**.

(15) Asgari, E.; Mofrad, M. R. Continuous Distributed Representation of Biological Sequences for Deep Proteomics and Genomics. *PLoS One* **2015**, *10* (11), e0141287.

(16) Irwin, J. J.; Sterling, T.; Mysinger, M. M.; Bolstad, E. S.; Coleman, R. G. ZINC: A Free

Tool to Discover Chemistry for Biology. *Journal of Chemical Information and Modeling*. 2012, pp 1757–1768.

(17) Gaulton, A.; Bellis, L. J.; Bento, A. P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani, B.; Overington, J. P. ChEMBL: A Large-Scale Bioactivity Database for Drug Discovery. *Nucleic Acids Res.* **2012**, *40* (D1), D1100-1107.

(18) Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P. The ChEMBL Bioactivity Database: An Update. *Nucleic Acids Res.* **2014**, *42* (D1), D1083-1090.

(19) Schneider, N.; Sayle, R. A.; Landrum, G. A. Get Your Atoms in Order—An Open-Source Implementation of a Novel and Robust Molecular Canonicalization Algorithm. *J. Chem. Inf. Model.* **2015**, *55* (10), 2111–2120.

(20) Řehuřek, R.; Sojka, P. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*; ELRA: Valletta, Malta, 2010; pp 45–50.

(21) Consortium, T. U. UniProt: The Universal Protein Knowledgebase. *Nucleic Acids Res.* **2017**, *45* (D1), D158–D169.

(22) Sandberg, M.; Eriksson, L.; Jonsson, J.; Sjöström, M.; Wold, S. New Chemical Descriptors Relevant for the Design of Biologically Active Peptides. A Multivariate Characterization of 87 Amino Acids. *J. Med. Chem.* **1998**, *41* (14), 2481–2491.

(23) Kooistra, A. J.; Kanev, G. K.; van Linden, O. P. J.; Leurs, R.; de Esch, I. J. P.; de Graaf, C. KLIFS: A Structural Kinase-Ligand Interaction Database. *Nucleic Acids Res.* **2016**, *44* (D1), D365–D371.

(24) Yagnik, J.; Strelow, D.; Ross, D. A.; Lin, R. S. The Power of Comparative Reasoning.

In *Proceedings of the IEEE International Conference on Computer Vision*; 2011; pp 2431–2438.

(25) Delaney, J. S. ESOL: Estimating Aqueous Solubility Directly from Molecular Structure. *J. Chem. Inf. Model.* **2004**, *44* (3), 1000–1005.

(26) Hansen, K.; Mika, S.; Schroeter, T.; Sutter, A.; ter Laak, A.; Steger-Hartmann, T.; Heinrich, N.; Muller, K. R. Benchmark Data Set for in Silico Prediction of Ames Mutagenicity. *J. Chem. Inf. Model.* **2009**, *49* (9), 2077–2081.

(27) Tox21 Data Challenge 2014 https://tripod.nih.gov/tox21/challenge/ (accessed Sep 27, 2017).

(28) Ramsundar, B.; Eastman, P.; Feinberg, E.; Gomes, J.; Leswing, K.; Pappu, A.; Wu, M.; Pande, V. DeepChem: Deep-Learning Models for Drug Discovery and Quantum Chemistry. 2017.

(29) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

(30) Chen, T.; Guestrin, C. XGBoost : Reliable Large-Scale Tree Boosting System. *arXiv* **2016**, 1–6.

(31) Chollet, F.; others. Keras. **2015**.

(32) Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2015.

(33) Nair, V.; Hinton, G. E. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*; Fürnkranz, J., Joachims, T., Eds.; Omnipress, 2010; pp 807–814.

(34) Kingma, D. P.; Lei Ba, J. Adam: A Method of Stochastic Optimization. *arXiv1412.6980*

[cs] **2015**, 1–15.

(35) Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15* (1), 1929–1958.

(36) Wu, Z.; Ramsundar, B. B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: A Benchmark for Molecular Machine Learning. *CoRR* **2017**, *abs/1703.0*.

(37) Duvenaud, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gómez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P.; Ryan P. Adams, D. M. D. D. J. A.-I. R. G.-B. T. H. A. A.-G. Convolutional Networks on Graphs for Learning Molecular Fingerprints. *arXiv.org* **2015**.

(38) Lusci, A.; Pollastri, G.; Baldi, P. Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-like Molecules. *J Chem Inf Model* **2013**, *53* (7), 1563–1575.

(39) Zhang, H.; Kang, Y.-L.; Zhu, Y.-Y.; Zhao, K.-X.; Liang, J.-Y.; Ding, L.; Zhang, T.-G.; Zhang, J. Novel Naïve Bayes Classification Models for Predicting the Chemical Ames Mutagenicity. *Toxicol. Vitr.* **2017**, *41*, 56–63.

(40) Cortés-Ciriano, I.; Ain, Q. U.; Subramanian, V.; Lenselink, E. B.; Méndez-Lucio, O.; IJzerman, A. P.; Wohlfahrt, G.; Prusis, P.; Malliavin, T. E.; van Westen, et al. Polypharmacology Modelling Using Proteochemometrics (PCM): Recent Methodological Developments, Applications to Target Families, and Future Prospects. *Med. Chem. Commun.* **2015**, *6* (1), 24–50.