
Exploring molecular embeddings for cancer drug screening

Neil Thomas

1 Introduction

There is great demand for new compounds that could be used as drugs to treat disease. Drugs may be precisely targeted to subpopulations, or that can circumvent disease resistance. In addition, drugs may have many negative side effects: can similar therapeutic effects be found in different compounds with fewer side effects?

Testing massive libraries with hundreds of thousands of compounds for their biological effects is time consuming and expensive. To reduce this cost, recent research computationally explores the space of compounds using generative models (e.g. variational autoencoders). The parameter space of these generative models can be optimized over to maximize or specify the output of a predictor function (e.g. drug effect) as in Brookes and Listgarten [2018], Gómez-Bombarelli et al. [2018]. Chemistry suffers from a problem similar to biology, natural language, and natural images: there are massive unlabeled datasets, and very few corresponding labeled datasets. For this reason, unsupervised and semi-supervised modeling approaches hold promise for helping models predict function from chemical structure.

Another promising avenue for meaningfully summarizing structural similarity follows research from natural language processing, and the learning of word and document embeddings as in Goldberg et al. [2014]. By drawing the analogy of graphical substructures (rooted subgraphs) within a complete graph as “words” within a “document”, similar intuition can be applied to learning δ -dimensional continuous embeddings for arbitrary graph structures, deemed *graph2vec* by Narayanan et al.. Graphs with similar substructures are similar graphs, and are close (in a Euclidean distance sense) in the embedding space. Since molecules can be represented as graphs, *graph2vec* embeddings can take advantage of large, unlabeled molecular datasets to learn rich embeddings that capture structural similarity between molecules. This also circumvents one of the problems with learning molecular embeddings via complicated strings written in Simplified Molecular-Input Line-Entry System (SMILES) as in Gómez-Bombarelli et al. [2018]. *graph2vec* does not need to learn the complex chemical syntax of SMILES. “Word” embeddings for n-grams of amino acids and nucleotides have already shown promise for being able to meaningfully ,capture biophysical characteristics of proteins and DNA sequences such as in Hamid and Friedberg [2018], Asgari and Mofrad [2015].

In this paper we examine the use of *graph2vec* as a means for refining compound screens.

2 Methods

2.1 Word Embeddings or *word2vec*

Finding vector-space embeddings for words from large text corpora is an essential pre-processing step for any downstream Natural Language Processing (NLP) task (text prediction, sentiment analysis, etc). The fundamental algorithm for determining word embeddings developed in Mikolov et al. [2013] is known as *word2vec*. The objective of the *word2vec* model (parameterized by θ) is: given a word w_t , to maximize the log-likelihood of the context $c = [w_{t-c}, \dots, w_{t+c}]$, i.e.

$$\arg \max_{\theta} \sum_{t=1}^T \log Pr(w_{t-c}, \dots, w_{t+c} | w_t; \theta)$$

The underlying assumption of this model is the *Distributional Hypothesis*, which states that *words that occur in the same contexts tend to have similar meanings*. Assuming the context words are independent of one another (the “bag of words” model) given the target word, the log probability can be rewritten

$$\arg \max_{\theta} \sum_{t=1}^T \sum_{\substack{c \leq j \leq c \\ j \neq 0}} \log Pr(w_{t+j} | w_t; \theta)$$

For learning word embeddings, $Pr(w_{t+j} | w_t)$ can be written as a softmax over the dot-product of δ -dimensional word embeddings, i.e. their vector representations $v_w \in \mathcal{R}^\delta$

$$Pr(w_{t+j} | w_t) = \frac{\exp(v_{w_t} \cdot v_{w_{t+j}})}{\sum_{w \in \mathcal{V}} \exp(v_{w_t} \cdot v_w)}$$

Where \mathcal{V} is the vocabulary size.

Training this model can be inefficient, as it involves computing a sum over a very large vocabulary \mathcal{V} at every step. Instead, word embedding models are trained using a procedure known as negative sampling.

2.2 Negative Sampling

The goal of negative sampling is to maximize the probability that observed (word, context) pairs are classified as coming from the data $Pr(D = 1 | w, c; \theta)$, and similarly to minimize the probability the (word, context) pairs sampled from a negative distribution \mathcal{D}' are classified as coming from the data.

$$\arg \max_{\theta} \sum_{(w,c) \in \mathcal{D}} \log Pr(D = 1 | c, w; \theta) + \sum_{(w,c) \in \mathcal{D}'} \log Pr(D = 0 | c, w; \theta)$$

In practice these are usually sampled at random from the vocabulary. The probabilities can be rewritten in terms of the vector embeddings

$$\arg \max_{\theta} \sum_{(w,c) \in \mathcal{D}} \log \frac{1}{1 + \exp(-v_c \cdot v_w)} + \sum_{(w,c) \in \mathcal{D}'} \log \frac{1}{1 + \exp(v_c \cdot v_w)}$$

In practice more than one negative sample corresponds to each positive sample. This objective encourages similar words to be mapped to similar vectors, while dissimilar words (i.e. words appearing in dissimilar contexts) to be mapped to dissimilar vectors.

Note that the parameters of the model θ are precisely the word embeddings v_w . The output classification probability can be represented as a neural network that takes as input two one-hot encoded words w and w' sampled from the context, maps to the vector representation of those words via a

“representation” matrix $w^\top R = v_w$ and $w'^\top R = v_{w'}$, and then takes the inverse-logit of their dot product to get the classification probabilities. A negative cross-entropy loss function can then be used to back-propagate gradients through the network to train.

2.3 Document Embeddings or *doc2vec*

An extension to *word2vec* for learning embeddings of arbitrary length word sequences (sentences, paragraphs, documents) was proposed by Le and Mikolov [2014]. In their distributed bag of words model, for a document d_i with length l_i composed of words $\{w_1, \dots, w_{l_i}\}$ the model maximizes

$$\arg \max_{\theta} \sum_{j=1}^{l_i} \log Pr(w_j | d_i; \theta)$$

Where again the probability is defined by a softmax as before. Both the documents d and words w are embedded in a δ -dimensional space, i.e. $v_d, v_w \in \mathcal{R}^\delta$. At each iteration of training, a text window is sampled from the document D , and a random word is sampled from that window. A classification task is then formed using that word.

2.4 Graph Embeddings or *graph2vec*

The problem statement here follows precisely the statement in the *graph2vec* paper by Narayanan et al. (Section 2). We give the intuition here. For a given graph $G = (N, E, \lambda)$, a rooted subgraph of degree k around node $n \in N$ includes all the nodes and edges that are reachable in k hops from n , see Fig 4. These rooted subgraphs sg make up a graph G in the same way that words make up a sentence. Our negative sampling objective then looks like:

$$\arg \max_{\theta} \sum_{(sg, G) \in \mathcal{D}} \log \frac{1}{1 + \exp(-v_{sg} \cdot v_G)} + \sum_{(sg, G) \in \mathcal{D}'} \log \frac{1}{1 + \exp(v_{sg} \cdot v_G)}$$

These subgraphs capture nonlinear structure in graphical data that linear paths, for example, do not, according to Shervashidze et al. [2011]. Subgraphs depend on the labels of the nodes and edges. In the results outlined in this paper, node labels were ignored (in our case, elements of the molecule), and instead nodes were by default labeled by their connectivity.¹

¹This was due to an unintentional technical mishap, but ended up providing a very interesting baseline regardless.

3 Results

3.1 Chemical Compound Dataset

284,176 chemical compounds were downloaded from the National Cancer Institute (NCI) Developmental Therapeutics Program (DTP).² These compounds correspond to the NCI Open Set: a large set of compounds provided for research purposes, with little experimental data associated with them. Predicted activity for the compounds in the Open Set was extracted from the Cactus NCI Database Browser.³ These predicted activities (e.g. water-octanol partition coefficient, mutagenicity) are computed using Prediction of Activity Spectra for Substances (PASS) Lagunin et al. [2000], a method trained on around 30,000 biologically active compounds which uses shared “Multilevel Neighborhood of Atoms” (akin to subgraphs) to infer properties of compounds not in the training set.

In addition, the Diversity Set, a strict subset of the Open Set totalling 1593 compounds, was downloaded from the NCI.⁴ The Diversity Set corresponds to compounds which were selected to represent the structural diversity present in the Open Set. They were selected via a greedy algorithm which uses defined “pharmacophoric centers” (structural motifs deemed biologically active). Iteratively, new compounds were added to the set if they had more than 5 new pharmacophores compared to the existing set. Though no journal references or code were found documenting the procedure, it is described on the NCI website.⁵

The Open Set was filtered to compounds that only contain elements present in the Diversity Set. Precisely, these elements were Arsenic, Bromine, Carbon, Chlorine, Fluorine, Iodine, Nitrogen, Oxygen, Phosphorous, and Sulpher. Hydrogens are implicitly in every structure, but they are not explicitly considered in the construction of subgraphs. Compounds were also filtered from the Open Set if they were not a single connected component (e.g. solute and solvent pair, or a salt). This filtered the total unlabeled compound set used for downstream analysis to 232,941 compounds.

3.2 Learning the Molecular Embedding

The set of compounds were converted into graphs and then used as training data. *graph2vec* uses subgraphs of depth k and smaller as its “vocabulary” for prediction. Maximum depth vocabularies of subtree depth 3 and depth 4 were used. The size of the subtree vocabulary at depths 1, 2, 3, and 4 were 7, 126, 7877, and 113944, respectively. 128 and 256 dimensional embeddings were trained, each for 10 and 100 epochs of training with an exponentially decaying learning rate initialized at 0.3. The models were trained on a Microsoft Azure NC6 GPU node (Tesla K80 GPU, 6 CPUs) and took between 6 hours (for degree 3 subgraphs) and 10 hours (for degree 4 subgraphs) for 100 epochs of training.

3.3 Visualizing the Embedding

After the embeddings were trained, they were visualized using UMAP, a “topological” dimensionality reduction method that is a much faster alternative to t-SNE McInnes et al. [2018]. Each embedding was visualized in 2 dimensions via UMAP, and colored by PASS predicted properties. To create meaningful colorations for these heavy tailed distributions (see Fig 6), any compound in the 99th percentile and above was mapped to the same color, and the same for the 1st percentile and below. The visualized properties include: “logP”, a predicted water-octanol partition coefficient, which is a proxy for hydrophobicity, an important component of whether compounds will pass through the cell membrane upon delivery, “Weight”, the molecular weight, computed as the sum of the constituent atomic weights, and the predicted (via PASS) probabilities of activity in four classes of drug: Anti-inflammatory, Carcinogenic, Immunostimulant, and Immunosuppressant. Where predicted property data was available for only a subset of the compounds, only that subset is plotted.

These colorings, as seen in Figures 9, 10, 11, 12, reveal that these properties of chemical interest are not randomly distributed throughout the space, but rather show qualitatively meaningful clustering

²<https://wiki.nci.nih.gov/display/NCIDTPdata/Chemical+Data> June 2016 Release

³<https://cactus.nci.nih.gov/ncidb2.2/>

⁴<https://wiki.nci.nih.gov/display/NCIDTPdata/Compound+Sets> Diversity Set V

⁵https://dtp.cancer.gov/organization/dscb/obtaining/available_plates.htm

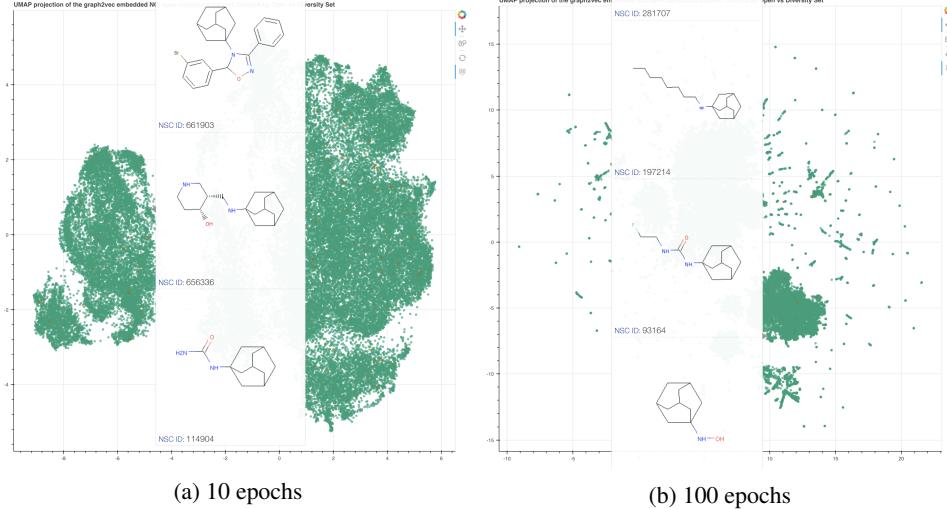


Figure 1: Adamantane clearly picked out as a signature as early as 10 epochs. UMAP 2D visualization of 128 dimensional embedding of degree 4 subgraphs. The small grey triangle at the border of the tooltip (hard to see) indicates the position of the cluster.

within the embedding space. This shows that structural similarity (as defined by our *graph2vec* embedding) can act as a good proxy for chemical activity. More intriguing, are the appearance of multiple clusters of the same color, such as the yellow in Fig 12 (a), which imply that such an embedding could be used to find *dissimilar* structures with similar activity, potentially mediating different biological pathways.

All of the cancer drug embeddings are available to the curious reader for interactive exploration at <https://people.eecs.berkeley.edu/~nthomas/embeddings.html>

3.3.1 Clustering within the Embedding Space

The interactive visualizations were used for qualitatively evaluating clustering within the embedding space. Some very strong connectivity motifs popped out early in training, as in Figure 1, where compounds with four fused 6-membered rings (Adamantane) were all mapped to similar positions. This motif has a very distinct 3-dimensional structure, shown in Figure 2, which bodes well for this type of embedding’s ability to predict biological interaction (e.g. with proteins), which is mediated by spatial extent and stereochemistry. By 100 epochs the embedding has learned a lot of very tight clusters, as seen in Figure 8b. Tight clusters in general correspond to identical graphs (ignoring the elemental node labels), some of which are enumerated in Figure 3.

In contrast to the tight clusters, the more diffuse clouds are difficult to interpret, and may correspond very loosely to the number of rings in the structure. The connectivity-based embedding may simply not hold enough information to reasonably disentangle the vast majority of compounds which inhabit the cloud.

⁶<https://dtp.cancer.gov/dtpstandard/servlet/Chem3D?testshortname=3D+chemical+structure&searchtype=NSC&searchlist=93164>

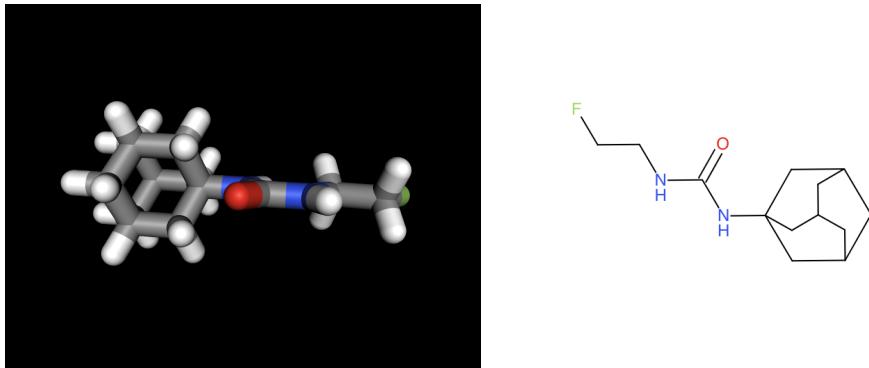


Figure 2: 3D structure (left) and 2D structure (right) of a molecule (NSC ID: 93164) with the fused tetrahedron (Adamantane). A nice example of molecular graph connectivity that is indicative of unique 3D structure. 3D structure taken from DTP website.⁶

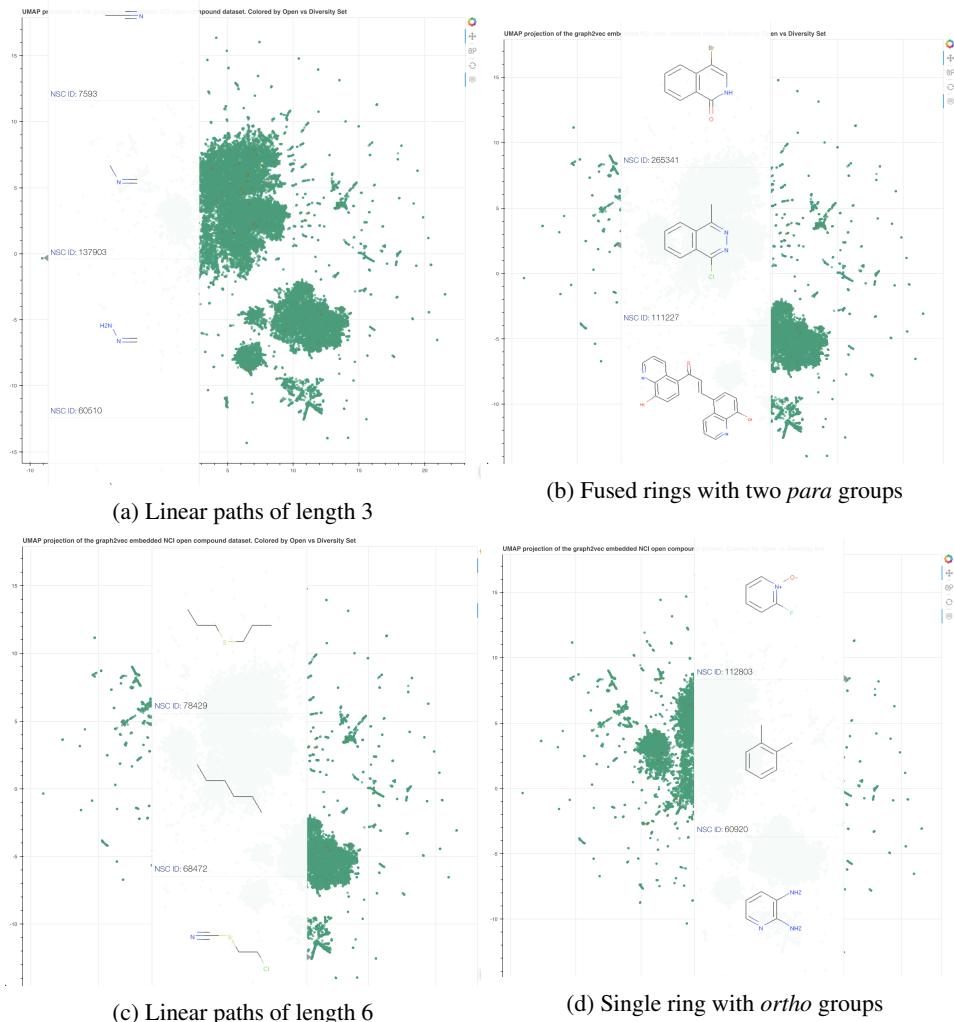


Figure 3: Clusters found in the 128 dimensional embedding of degree 4 subgraphs trained for 100 epochs. The small grey triangle at the border of the tooltip (hard to see) indicates the position of the cluster. Of particular note are the fused rings with *para* groups in (b) where the bottom example shows that a molecule with that motif appearing twice (with a small bridge in between) is mapped to similar vectors as molecules with the motif appearing once

4 Discussion

The *graph2vec* embedding, qualitatively, shows promise for learning structural motifs from large unlabeled molecular datasets. In addition, there seems to be some nontrivial correlation between properties of chemical interest and their positioning within the embeddings.

Possible applications of this kind of embedding include refining molecular searches with queries of the type “find a compound with structural similarity to compound X, while maximizing predicted probability of activity Y.” In addition, molecular embeddings serve as a superior alternative to molecular kernels (some of which are based on the Weisfeiler Lehman kernel), as one can train arbitrary machine learning models on top of them for any type of supervised problem (as opposed to only kernel-based methods like SVMs). Future work should involve using these embeddings for prediction tasks on labeled datasets, which will be easier in the future as hopefully more datasets become open.

In particular, the work described in this paper suffers from a major flaw, which is that it does not take into account node labels within the graph. A path between C-C and N-C, say, are treated as identical, as long as the connectivity of each node is the same. Including element labels in the *graph2vec* subgraph vocabulary construction should be a first priority in future work, and should improve any prediction task based on the embedding. In addition, the application of *graph2vec* described here does not take into account edge labels (e.g. bond type), so single, double, and triple bonds are not explicitly treated differently. The implicit change in connectivity of neighboring molecules would give them different signatures, but this is not always enough, for example in the graphs C-C and C=C. An explicit consideration of distinct bond types could improve molecular embeddings, as different bond types have different 3 dimensional geometry. Another avenue would be to include bond information implicitly by explicitly including hydrogens in the construction of the molecular graph, guaranteeing that there would be downstream connectivity changes for a bond type change.

Much of the future of cheminformatics depends on improved predictors for complex behavior, especially for drug screening, where patient response can depend on a myriad of biological pathways. Leveraging rich labeled datasets of compound response, like those produced by high-throughput light microscopy drug screens in Kang et al. [2016] to improve compound embeddings is another line of future work.

5 Conclusion

The results outlined in this work show that even naive notions of connectivity already show some important trends in chemical function. In the future, we hope that the molecular embedding explored in this work procedures could be used by the National Cancer Institute to recommend optimal Diversity Sets for larger compound sets, or to inform molecular queries for researchers looking for similar structures to ones that they find to have therapeutic function.

References

- David H. Brookes and Jennifer Listgarten. Design by adaptive sampling. oct 2018. URL <http://arxiv.org/abs/1810.03714>.
- Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Central Science*, 4(2):268–276, feb 2018. ISSN 23747951. doi: 10.1021/acscentsci.7b00572. URL <http://pubs.acs.org/doi/10.1021/acscentsci.7b00572>.
- Yoav Goldberg, Omer Levy, Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. word2vec Explained: Deriving Mikolov et al.’s Negative-Sampling Word-Embedding Method The skip-gram model. 2014. URL <https://arxiv.org/pdf/1402.3722.pdf>.
- Annamalai Narayanan, Mahinthan Chandramohan, Rajasekar Venkatesan, Lihui Chen, Yang Liu, and Shantanu Jaiswal. graph2vec: Learning Distributed Representations of Graphs. doi: 10.1145/1235. URL <https://arxiv.org/pdf/1707.05005.pdf>.
- Md Nafiz Hamid and Iddo Friedberg. Identifying Antimicrobial Peptides using Word Embedding with Deep Recurrent Neural Networks. *bioRxiv*, page 255505, jan 2018. doi: 10.1101/255505. URL <https://www.biorxiv.org/content/early/2018/01/29/255505>.
- Ehsaneddin Asgari and Mohammad R.K. Mofrad. Continuous distributed representation of biological sequences for deep proteomics and genomics. *PLoS ONE*, 10(11):e0141287, nov 2015. ISSN 19326203. doi: 10.1371/journal.pone.0141287. URL <http://dx.plos.org/10.1371/journal.pone.0141287>.
- Tomas Mikolov, Greg Corrado, Kai Chen, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, pages 1–12, 2013. ISSN 15324435. doi: 10.1162/153244303322533223. URL <http://arxiv.org/pdf/1301.3781v3.pdf>.
- Quoc V. Le and Tomas Mikolov. Distributed Representations of Sentences and Documents. may 2014. URL <http://arxiv.org/abs/1405.4053>.
- Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn, and Karsten M Borgwardt. Weisfeiler-Lehman Graph Kernels. *Journal of Machine Learning Research*, 12:2539–2561, 2011. ISSN 1532-4435. doi: 10.1.1.232.1510. URL <http://www.jmlr.org/papers/volume12/shervashidze11a/shervashidze11a.pdf>.
- A. Lagunin, A. Stepanchikova, D. Filimonov, and V. Poroikov. PASS: prediction of activity spectra for biologically active substances. *Bioinformatics*, 16(8):747–748, aug 2000. ISSN 1367-4803. doi: 10.1093/bioinformatics/16.8.747. URL <https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/16.8.747>.
- Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. feb 2018. URL <http://arxiv.org/abs/1802.03426>.
- Jungseog Kang, Chien-Hsiang Hsu, Qi Wu, Shanshan Liu, Adam D Coster, Bruce A Posner, Steven J Altschuler, and Lani F Wu. Improving drug discovery with high-content phenotypic screens by systematic selection of reporter cell lines. *Nature Biotechnology*, 34(1):70–77, jan 2016. ISSN 1087-0156. doi: 10.1038/nbt.3419. URL <http://www.nature.com/articles/nbt.3419>.

6 Supplemental Figures

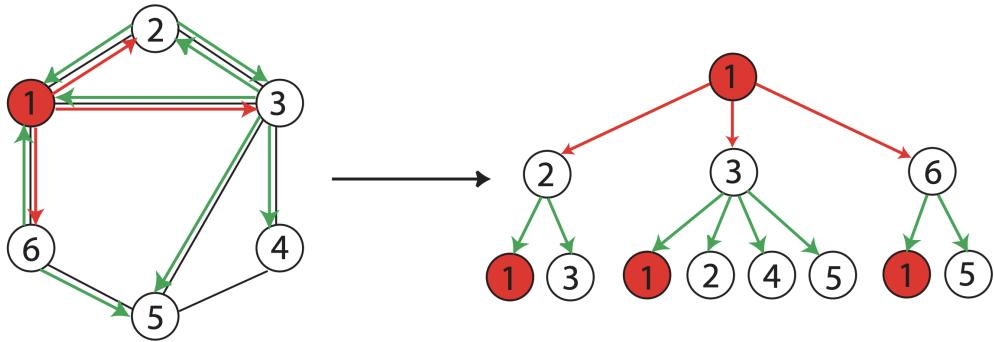


Figure 4: Schematic showing the nodes reachable within two hops of the root node 1. Figure taken from Shervashidze et al. [2011].

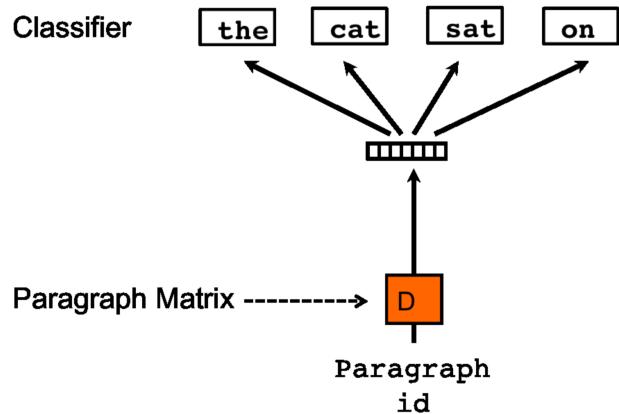


Figure 5: Schematic of *doc2vec* prediction taken from Le and Mikolov [2014]. The document vector is used to predict words in a small window.

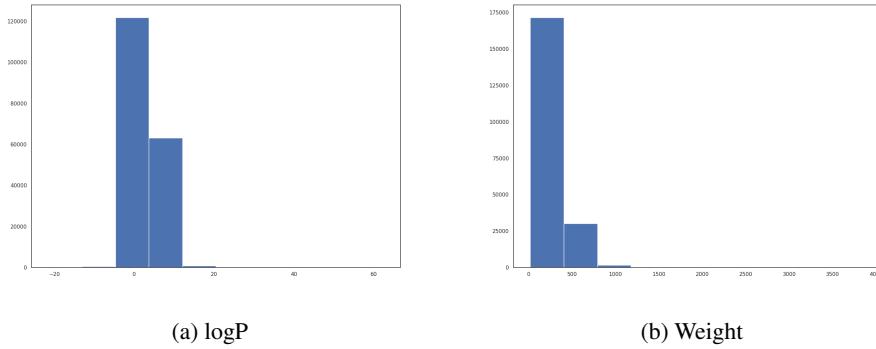


Figure 6: Histograms for predicted compound properties. These exhibit a heavy right skew.

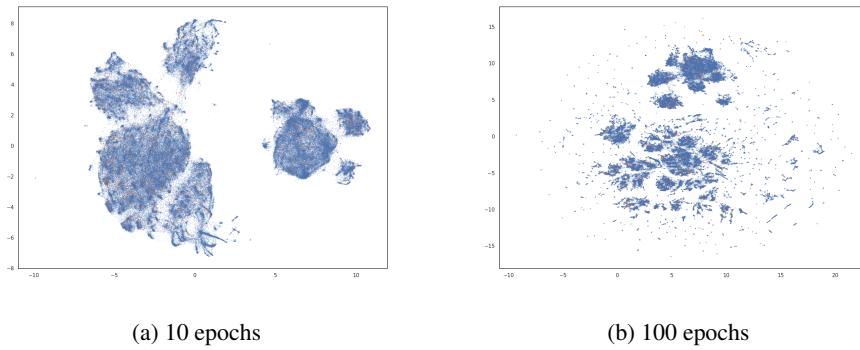


Figure 7: 256-dimensional embedding for degree 3 subtrees, visualized in 2 dimensions via UMAP. Diversity V Set compounds colored in orange, while other Open Set compounds colored in blue. The diversity compounds are spread well throughout the embedding space.

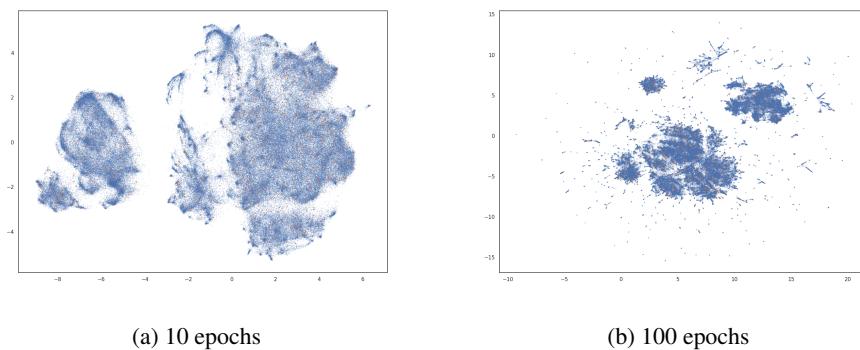


Figure 8: 128-dimensional embedding for degree 4 subtrees, visualized in 2 dimensions via UMAP. Diversity V Set compounds colored in orange, while other Open Set compounds colored in blue. The diversity compounds are spread well throughout the embedding space.

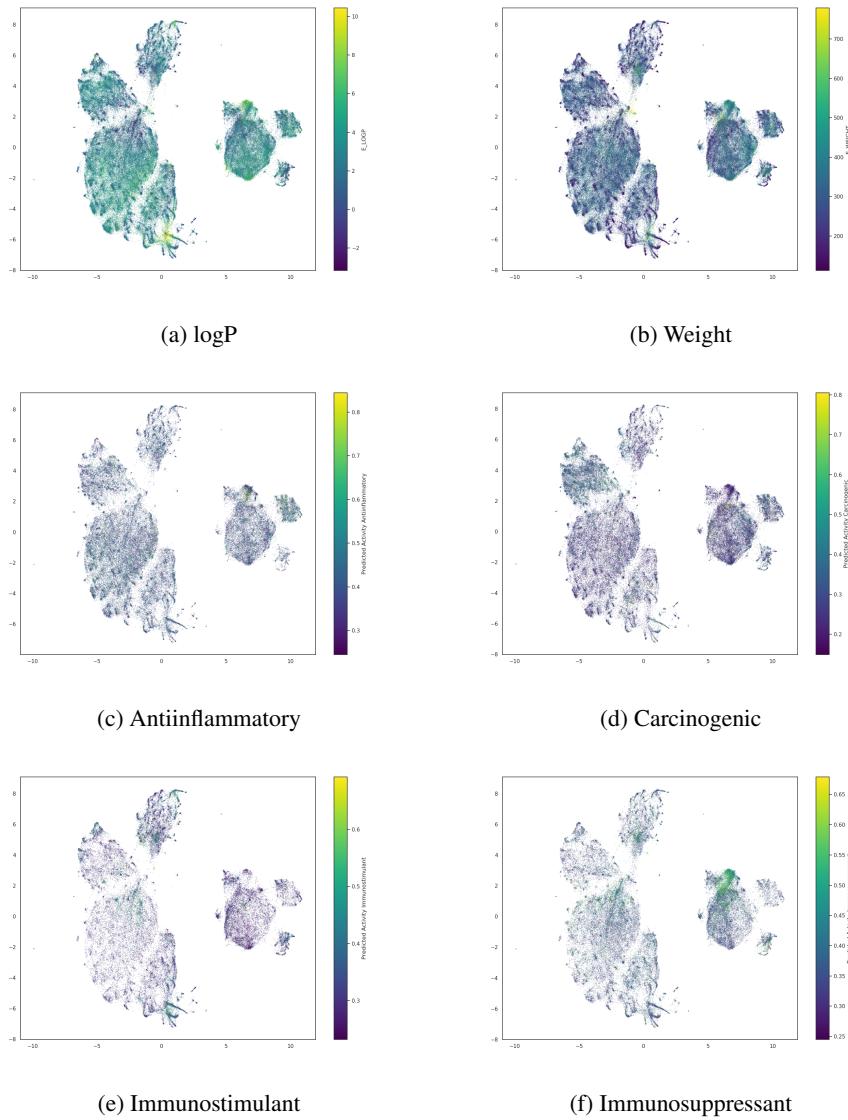


Figure 9: 256-dimensional embedding for degree 3 subtrees trained for 10 epochs

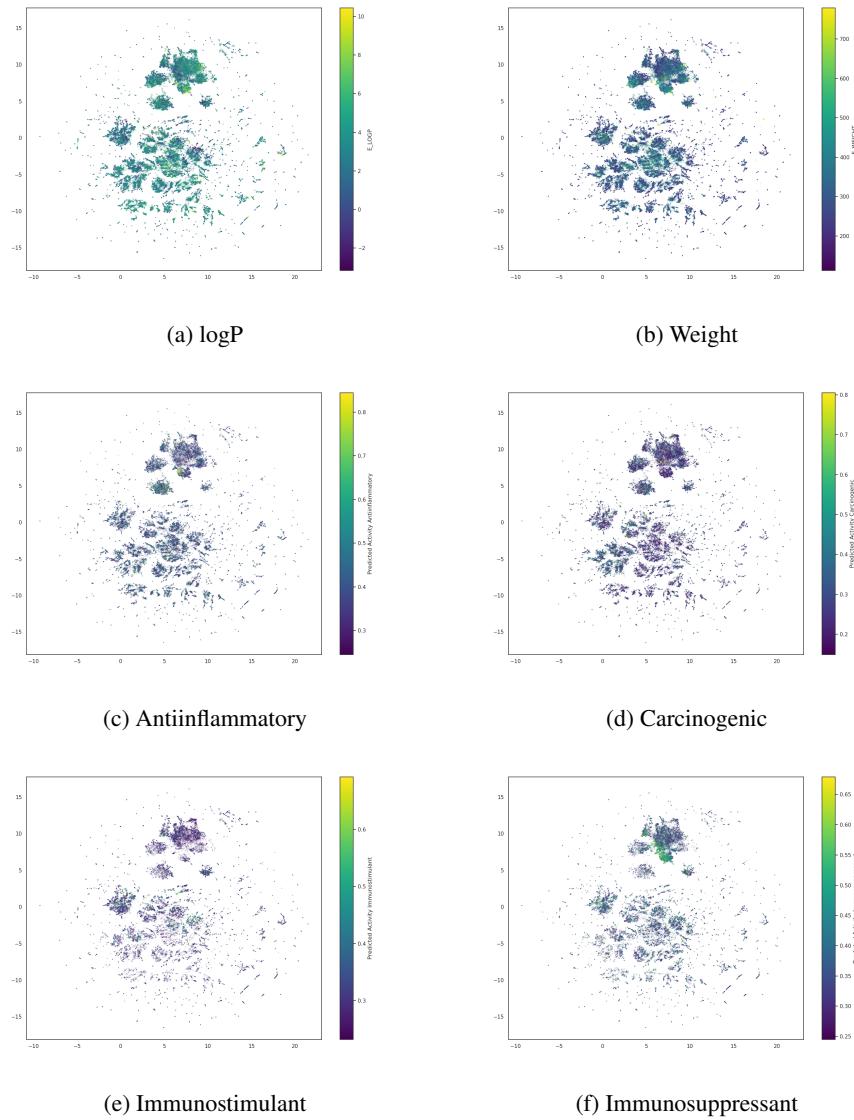


Figure 10: 256-dimensional embedding for degree 3 subtrees trained for 100 epochs.

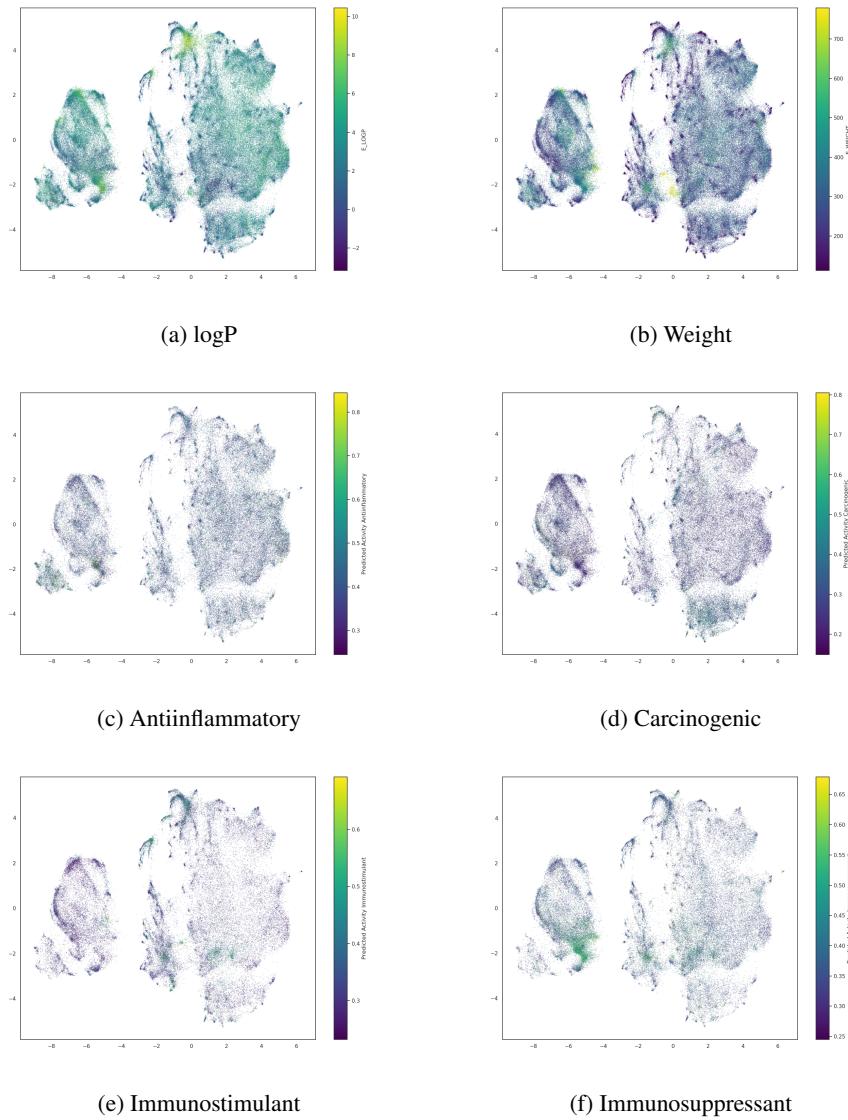


Figure 11: 128-dimensional embedding for degree 4 subtrees trained for 10 epochs.

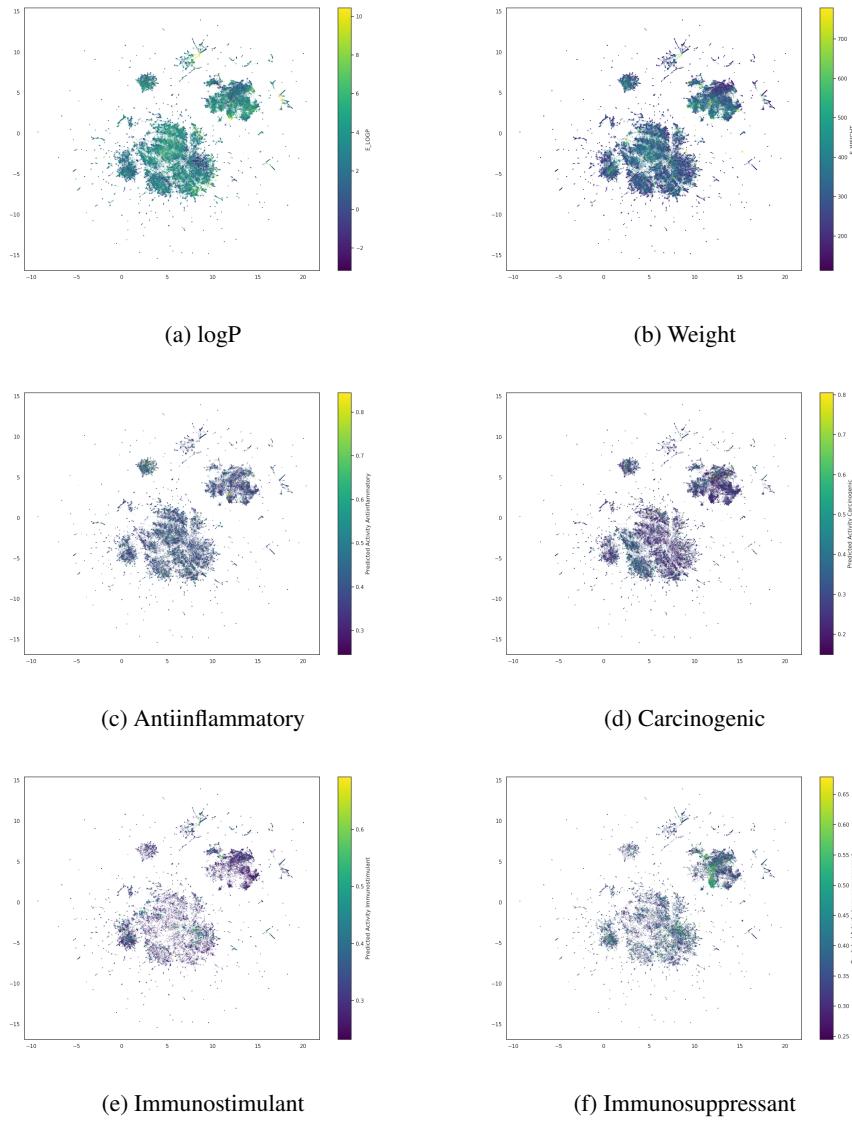


Figure 12: 128-dimensional embedding for degree 4 subtrees trained for 100 epochs.