The Leffingwell Odor Dataset

Predicting properties of molecules is an area of growing research in machine learning [1, 2], particularly as models for learning from graph-valued inputs improve in sophistication and robustness [3, 4]. A molecular property prediction problem that has received comparatively little attention during this surge in research activity is building Structure-Odor Relationships (SOR) models (as opposed to Quantitative Structure-Activity Relationships, a term from medicinal chemistry). This is a 70+ year-old problem straddling chemistry, physics, neuroscience, and machine learning [5].

To spur development on the SOR problem, we curated and cleaned a dataset of 3523 molecules associated with expert-labeled odor descriptors from the *Leffingwell PMP 2001* database [6]. We trained Graph Neural Networks (GNNs) [4, 7] to predict these odor descriptors using a molecule's graph structure alone, and compared their performance to alternative chemoinformatic models. We provide featurizations of all molecules in the dataset using bit-based and count-based fingerprints [8], Mordred molecular descriptors [9], and the embeddings from our trained GNN model [10]. This dataset is comprised of two files:

- **leffingwell_data.csv**: this contains molecular structures, and what they smell like, along with train, test, and cross-validation splits.
- **leffingwell_embeddings.npz**: this contains several featurizations of themolecules in the dataset.

1 Dataset

The Leffingwell dataset is stored as `leffingwell_data.csv`. We assembled this data from expert-labeled set of 3523 molecules from a single source, the *Leffingwell PMP 2001* database [6]. Molecules are labeled with one or more odor descriptors by olfactory experts (usually a practicing perfumer), creating a multi-label prediction problem. The Leffingwell data is originally free-form text descriptions (e.g. "Floral, somewhat animalic with musky undertones") which we canonicalize as a variable-sized list of descriptors (e.g. "floral, animalic, musky"). After filtering for odor descriptors with at least 20 representative molecules, 113 odor descriptors remained (Figure 1A), including an *odorless* descriptor. Some odor descriptors were extremely common, like *fruity* or *green*, while others were rare, like *radish* or *brothy*. This dataset is composed of materials for perfumery, and so is biased away from malodorous compounds. There is also skew in label counts resulting from different levels of specificity. For example, *fruity* will always be more common than *pineapple*. Most molecules are associated with four or five odor descriptors, up to a maximum of eight (Figure 1B).

The raw data CSV contains the following columns:

- **chemical_name**: Common name of the molecule, as originally listed in the Leffingwell database.
- **smiles**: Structure of the molecules, given as an isomeric canonical SMILES. This column incorporates multiple corrections on top of the original Leffingwell database.
- **database_id**: Database ID from the Leffingwell database.
- **cas**: CAS identifier for the molecule as listed in the original Leffingwell database.
- **odor_data**: Free-form text description as originally listed in the Leffingwell database. This column sometimes contains taste descriptors, which have been manually removed during processing.

- **odor_labels_filtered**: Post-processed odor labels. Processing steps include typo correction, synonym replacement, replacement of singleton labels with the minimally genericized label, removal of labels with < 20 occurrences, and finally, removal of rows with no labels.
- **labels_***: Train/test and CV splits for dataset. -1 indicates "Not part of split"; 0 indicates "Held-out split", and 1 indicates "Train split". The dataset uses a 80/20 train-test split, and the training split is further split into 16/16/16/16/16 cross-validation splits. All splits are constructed using scikit-multilearn's IterativeStratification with 2-order label combinations.

There is a strong co-occurrence structure among odor descriptors that reflects a common-sense intuition of which odor descriptors are similar and dissimilar (Figure 1C). For example, there is a *fruit* cluster that includes the *apple* and *berry*, descriptors, indicating that they often co-occur as descriptors in individual molecules. There is also a *clean* cluster with *mint*, *lemon*, *camphoreous* etc., and a *savory* cluster that includes *brothy*, *mushroom*, and *beefy*, among others. Previous approaches in SOR often train one model per odor descriptor, but in this dataset and benchmark, it is advantageous to use this correlation structure and predict on all 113 odor descriptor tasks at once.

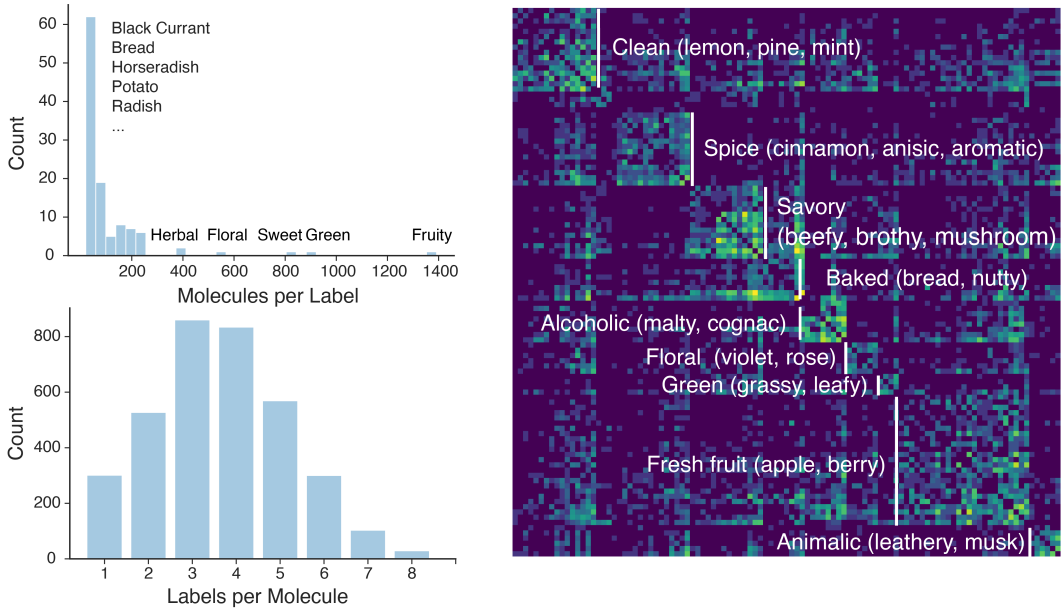

Figure 1: **Dataset overview.** **A.** Distribution of odor descriptor frequencies. **B.** Distribution of label density. **C.** Co-ocurrence matrix for odor descriptors. The 8 most frequent descriptors are removed for visual clarity, and remaining descriptors re-ordered using spectral clustering. Main odor groups with examples are highlighted. The color range is on a log-scale, and normalized such that each row and column sums to 1.

2 Molecular Featurizations

The featurizations of molecules are available in `leffingwell_embeddings.npz`, which you can load using the `numpy.load` function in the NumPy package in Python. There are five arrays stored in the file.

- **SMILES**: the structure of each molecule in the dataset specified as a SMILES string.
- **mordred_embeddings**: Mordred descriptors for each molecule, with null and zero-variance features dropped.
- **bfp_embeddings**: Bit-based molecular fingerprint descriptors for each molecule
- **cfp_embeddings**: Count-based molecular fingerprint descriptors for each molecule
- **mpnn_embeddings**: Learned neural network embeddings for each molecule

2.1 Mordred Features

```
The Mordred features are available in 'leffingwell_embeddings.npz'.
Load them with numpy.load, and access the 'mordred_embeddings' field.
Each row is a molecule, matching 'leffingwell_data.csv'.
```

There are several available hand-crafted featurizations for molecules, which are popular in the field of olfactory neuroscience. Both *Dragon* (closed source, [11]) and *Mordred* (open source, [9]) are approaches that include many thousands of computed molecular features. They are an agglomeration of several types of molecular information and statistics, such as counts of atom types, graph topology statistics, and acid/base counts. Some of these features are easily interpretable (e.g. *number of Carbon atoms*) and some are not (e.g. *spectral moment of order 4 from distance/detour matrix*). We use *Mordred* in the present work because it is open source, and we found no appreciable difference in predictive performance between these features and *Dragon* features.

2.2 Molecular Fingerprints

```
The fingerprint features are available in 'leffingwell_embeddings.npz'.
Load them with numpy.load, and access the 'bfp_embeddings' field for
bit-based features or 'cfp_embeddings' field for count-based features.
Each row is a molecule, matching 'leffingwell_data.csv'.
```

Molecular fingerprints encode topological environments of a molecular graph into a fixed-length vector. An environment is a fragment of the molecular graph, and indicates the presence of a single atom type or a functional group, e.g. an alcohol or ester group. This approach to featurizing molecules is popular in the field of medicinal chemistry; traditionally, bit-based Morgan fingerprints have been used in chemoinformatics for retrieving nearest neighbor molecules using Tanimoto similarity [12]. The more commonly used bit variant records the presence of a given environment (e.g., is there an ester in this molecule?), while the count variant records the number of instances of a given environment (e.g. how many ester groups are there in this molecule?). This information is hashed into a fixed-length vector. There are two tunable parameters: max topological radius and fingerprint vector size. The max topological radius determines the largest fragment which the fingerprint can represent. Fingerprint vector size affects how likely a hash collision can occur. We tune both of these parameters to maximize predictive performance.

In our baseline experiments, we explicitly compare bit-based path descriptors fingerprints (bFP) and count-based Morgan fingerprints (cFP), and find cFP to generally be superior. The cheminformatics package RDKit was used to generate both types of fingerprints [13].

2.3 Learned Graph Neural Network Embeddings

```
The GNN features are available in 'leffingwell_embeddings.npz'.
Load them with numpy.load, and access the 'mpnn_embeddings' field.
Each row is a molecule, matching 'leffingwell_data.csv'.
```

Most machine learning models require regularly-shaped input (e.g. a grid of pixels, or a vector of numbers) as input. Recently, Graph Neural Networks (GNNs) have enabled the use of irregularly-shaped inputs, such as graphs, to be used directly in machine learning applications [14]. Fields of use include predicting friendships in social network graphs, citation networks in academic literature, and most germane for this work, classification and regression tasks in chemistry [1].

All deep neural network architectures build representations of input data at their intermediate layers. The success of deep neural networks in prediction tasks relies on the quality of their learned representations, often referred to as embeddings [15]. For instance, ImageNet embeddings are often used as-is to make predictions on unrelated image tasks [16, 17], and with the advent of the BERT model and its cousins, this ability to use pre-trained embeddings is becoming common in natural language processing [18]. The structure of a learned embedding can even lead to insights on the task or problem area, and the embedding can even be an object of study itself [19, 20].

We save the activations of the penultimate fully connected layer as a fixed-dimension "odor embedding". The GNN model must transform a molecule's graph structure into a fixed-length represen-

tation that is useful for classification. Although the utility of learned neural network embeddings of molecules is still young and relatively unproven [21, 22], we still anticipate that a learned GNN embedding on an odor prediction task may include a semantically meaningful and useful organization of odorant molecules. We describe this method in detail in our preprint [10].

3 SOR Prediction Performance Benchmark

We benchmark classification performance for each odor descriptor in our dataset, as a multi-label classification problem. We compare the GNN model against random forest models (RF) and k-nearest neighbor models (KNN) on bit-based RDKit fingerprints (bFP), count-based Morgan fingerprints (cFP), and Mordred features. We report several metrics (Table 1), as each metric can highlight different performance characteristics. We also provide a table of per-odor performances. We primarily compare models on mean AUROC, averaged across odor descriptors; AUROC performance by descriptor is shown in Figure 2. We trained non-graph based fully-connected neural networks on cFP and bFP features, but their performance is indistinguishable from the RF model (data not shown).

	AUPRC	AUROC	F1
GNN	**0.434 [0.431, 0.478]**	**0.913 [0.908, 0.923]**	**0.406 [0.372, 0.423]**
RF-Mordred	0.367 [0.346, 0.408]	0.882 [0.871, 0.891]	0.336 [0.310, 0.346]
RF-Mordred/cFP	0.359 [0.346, 0.380]	0.881 [0.860, 0.888]	0.333 [0.296, 0.333]
RF-bFP	0.337 [0.337, 0.373]	0.855 [0.851, 0.861]	0.325 [0.291, 0.339]
RF-cFP	0.357 [0.354, 0.398]	0.867 [0.860, 0.880]	0.329 [0.306, 0.331]
kNN-bFP	0.342 [0.354, 0.393]	0.820 [0.808, 0.833]	0.330 [0.302, 0.340]
kNN-cFP	0.324 [0.314, 0.352]	0.805 [0.787, 0.816]	0.302 [0.273, 0.309]

Table 1: **Odor descriptor prediction results.** mean, 95% CI [lower, upper] bounds reported. Numbers reported are an unweighted mean across all 138 odor descriptors; see Supplemental Table 3 for results reported by odor label. Precision/recall decision thresholds are optimized for F1 score on a cross-validation split created from the training set. The best values for each metric are in bold. Models include graph neural networks (GNN), random forest (RF) and k-nearest neighbor. Featurizations include bit-based RDKit fingerprints (bFP), count-based Morgan fingerprints (cFP), and Mordred features. There was no statistical winner as measured by recall, and thus it is omitted; these scores ranged from 0.365 to 0.393, with high overlap amongst all models.

Table of Per-Descriptor Results

AUROC and AUPRC performance results by descriptor for the GNN model and the Random Forest model with Mordred features.

	Datapoints		AUROC		AUPRC	
	Train	Test	GNN	RF-Mordred	GNN	RF-Mordred
alcoholic	68	17	1.00	0.99	0.87	0.77
aldehydic	21	4	0.93	0.90	0.31	0.11
alliaceous	53	15	0.94	0.93	0.32	0.38
almond	40	10	0.90	0.89	0.31	0.31
animal	38	10	0.86	0.82	0.47	0.23
anisic	34	9	0.99	0.98	0.59	0.28
apple	190	49	0.95	0.92	0.62	0.51
apricot	25	7	0.92	0.83	0.12	0.10
aromatic	43	12	0.89	0.88	0.14	0.15
balsamic	131	33	0.94	0.95	0.57	0.55
banana	52	15	0.93	0.91	0.47	0.33
beefy	21	5	0.98	0.97	0.34	0.20

Continued on next page

	Datapoints		AUROC		AUPRC	
	Train	Test	GNN	RF-Mordred	GNN	RF-Mordred
berry	55	16	0.88	0.85	0.21	0.17
black currant	18	5	0.88	0.88	0.43	0.29
brandy	23	7	0.99	0.96	0.53	0.13
bread	22	5	0.89	0.90	0.30	0.25
brothy	16	7	0.94	0.84	0.17	0.07
burnt	99	32	0.94	0.94	0.42	0.48
buttery	75	22	0.95	0.92	0.65	0.49
cabbage	22	5	0.97	0.95	0.30	0.19
camphoreous	55	15	0.90	0.95	0.55	0.45
caramellic	128	33	0.91	0.92	0.61	0.60
catty	16	4	0.99	0.81	0.45	0.15
chamomile	19	5	0.99	0.90	0.65	0.28
cheesy	110	32	0.90	0.91	0.53	0.63
cherry	34	7	0.91	0.82	0.26	0.21
chicken	23	6	0.96	0.85	0.17	0.05
chocolate	22	6	0.85	0.88	0.56	0.58
cinnamon	17	8	0.90	0.87	0.30	0.29
citrus	158	43	0.92	0.91	0.55	0.49
cocoa	58	21	0.92	0.87	0.42	0.38
coconut	34	8	0.97	0.90	0.57	0.47
coffee	44	13	0.94	0.92	0.36	0.30
cognac	62	16	0.97	0.90	0.46	0.35
coumarinic	29	8	0.99	0.96	0.73	0.47
creamy	62	22	0.83	0.84	0.23	0.23
cucumber	20	5	0.98	0.88	0.51	0.34
dairy	53	16	0.91	0.87	0.21	0.21
dry	25	6	0.93	0.93	0.19	0.32
earthy	130	43	0.79	0.80	0.28	0.30
ethereal	156	39	0.95	0.92	0.68	0.57
fatty	326	81	0.92	0.90	0.68	0.60
fermented	81	25	0.92	0.89	0.57	0.41
fishy	46	11	0.92	0.91	0.56	0.46
floral	434	119	0.89	0.86	0.55	0.49
fresh	187	50	0.77	0.75	0.21	0.23
fruity	1112	280	0.91	0.88	0.87	0.81
garlic	67	18	0.98	0.98	0.65	0.62
gasoline	38	10	0.95	0.94	0.50	0.40
grape	37	10	0.89	0.75	0.31	0.20
grapefruit	21	6	0.94	0.94	0.49	0.18
grassy	23	6	0.91	0.82	0.24	0.20
green	723	185	0.81	0.81	0.63	0.64
hay	33	8	0.96	0.83	0.42	0.19
hazelnut	25	6	0.98	0.98	0.31	0.24
herbal	319	87	0.84	0.79	0.47	0.41
honey	63	19	0.96	0.96	0.56	0.48
horseradish	17	3	0.94	0.93	0.36	0.22
jasmine	34	8	0.96	0.97	0.60	0.55
ketonic	24	6	1.00	0.98	0.79	0.55
leafy	43	10	0.82	0.70	0.16	0.16
leathery	16	4	0.87	0.74	0.30	0.27
lemon	20	5	0.88	0.75	0.31	0.20
malty	21	4	0.81	0.70	0.15	0.05
meaty	174	44	0.95	0.93	0.52	0.48
medicinal	27	9	0.97	0.89	0.57	0.49

Continued on next page

	Datapoints		AUROC		AUPRC	
	Train	Test	GNN	RF-Mordred	GNN	RF-Mordred
melon	66	18	0.94	0.92	0.40	0.33
metallic	17	8	0.80	0.82	0.34	0.40
milky	32	8	0.78	0.76	0.10	0.10
mint	96	27	0.96	0.94	0.57	0.46
mushroom	45	13	0.80	0.61	0.12	0.10
musk	16	4	0.54	0.74	0.50	0.34
musty	72	20	0.81	0.81	0.31	0.31
nutty	175	49	0.86	0.84	0.55	0.48
odorless	57	14	0.99	0.99	0.79	0.64
oily	159	41	0.95	0.92	0.66	0.48
onion	96	29	0.97	0.95	0.61	0.66
orange	41	13	0.90	0.82	0.21	0.33
orris	22	4	0.94	0.98	0.32	0.58
peach	35	12	0.86	0.83	0.20	0.10
pear	72	19	0.90	0.89	0.43	0.39
phenolic	63	18	0.96	0.96	0.73	0.59
pine	26	9	0.96	0.90	0.47	0.41
pineapple	109	30	0.91	0.85	0.43	0.43
plum	21	7	0.89	0.85	0.21	0.17
popcorn	18	5	0.95	0.85	0.59	0.64
potato	23	6	0.97	0.95	0.41	0.25
pungent	92	25	0.89	0.90	0.39	0.34
radish	16	4	0.91	0.92	0.56	0.44
ripe	26	5	0.94	0.87	0.07	0.03
roasted	156	39	0.96	0.96	0.50	0.62
rose	124	39	0.95	0.91	0.59	0.52
rum	45	14	0.88	0.87	0.35	0.25
savory	64	18	0.94	0.93	0.31	0.28
sharp	43	11	0.93	0.94	0.38	0.39
smoky	35	11	0.98	0.95	0.61	0.43
solvent	30	10	0.99	0.96	0.60	0.47
sour	35	9	0.87	0.85	0.21	0.26
spicy	158	45	0.90	0.88	0.55	0.45
strawberry	20	6	0.93	0.90	0.16	0.07
sulfurous	197	55	0.97	0.97	0.67	0.59
sweet	652	173	0.77	0.75	0.52	0.50
tea	27	8	0.86	0.82	0.24	0.51
tobacco	43	12	0.96	0.89	0.50	0.31
tomato	18	5	0.90	0.83	0.19	0.34
tropical	157	44	0.92	0.88	0.56	0.48
vanilla	49	12	0.99	0.98	0.71	0.61
vegetable	130	35	0.90	0.88	0.44	0.39
violet	20	6	0.90	0.89	0.45	0.34
warm	33	7	0.75	0.73	0.04	0.17
waxy	197	47	0.94	0.95	0.47	0.53
winey	141	33	0.91	0.85	0.31	0.18
woody	167	44	0.88	0.86	0.50	0.57

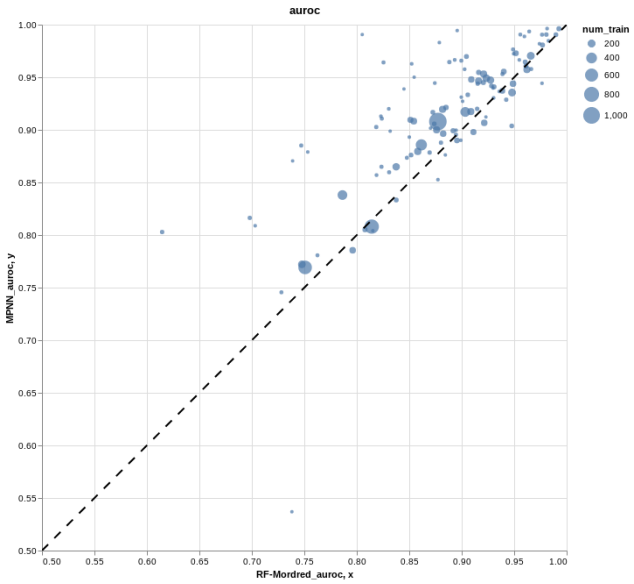

Figure 2: **Comparison of RF-Mordred and GNN, broken down by odor descriptor.** Each dot represents an odor descriptor, with size representing the number of positive examples. GNN outperforms RF-Mordred on nearly all odor descriptors.

References

[1] Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. MoleculeNet: a benchmark for molecular machine learning. *Chemical Science*, 9(2):513–530, 2018.

[2] John B O Mitchell. Machine learning methods in chemoinformatics. *Wiley Interdiscip. Rev. Comput. Mol. Sci.*, 4(5):468–481, September 2014.

[3] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. Computational capabilities of graph neural networks. *IEEE Trans. Neural Netw.*, 20(1):81–102, January 2009.

[4] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272, International Convention Centre, Sydney, Australia, 2017. PMLR.

[5] Karen J Rossiter. Structure-odor relationships. *Chem. Rev.*, 96(8):3201–3240, 1996.

[6] John C Leffingwell. Leffingwell & associates, 2005.

[7] David K Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Gómez-Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P Adams. Convolutional Networks on Graphs for Learning Molecular Fingerprints. In *Advances in Neural Information Processing Systems*, pages 2215–2223, 2015.

[8] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *J. Chem. Inf. Model.*, 50(5):742–754, May 2010.

[9] Hirotomo Moriwaki, Yu-Shi Tian, Norihito Kawashita, and Tatsuya Takagi. Mordred: a molecular descriptor calculator. *J. Cheminform.*, 10(1):4, February 2018.

[10] Benjamin Sanchez-Lengeling, Jennifer N Wei, Brian K Lee, Richard C Gerkin, Alán Aspuru-Guzik, and Alexander B Wiltschko. Machine learning for scent: Learning generalizable perceptual representations of small molecules. October 2019.

[11] Andrea Mauri, Viviana Consonni, Manuela Pavan, and Roberto Todeschini. Dragon software: An easy approach to molecular descriptor calculations. *Match*, 56(2):237–248, 2006.

[12] Gerald Maggiora, Martin Vogt, Dagmar Stumpfe, and Jürgen Bajorath. Molecular similarity in medicinal chemistry. *J. Med. Chem.*, 57(8):3186–3204, April 2014.

[13] RDKit: Open-source cheminformatics. http://www.rdkit.org.

[14] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *CoRR*, abs/1901.00596, 2019.

[15] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: a review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(8):1798–1828, August 2013.

[16] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

[17] Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 806–813, 2014.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. October 2018.

[19] Daniel L K Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, 19(3):356–365, 2016.

[20] Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda Viégas, and Martin Wattenberg. Visualizing and measuring the geometry of BERT. *arXiv preprint arXiv:1906.02715*, 2019.

[21] 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 Cent Sci*, 4(2):268–276, February 2018.

[22] Alex Zhavoronkov, Yan A Ivanenkov, Alex Aliper, Mark S Veselov, Vladimir A Aladinskiy, Anastasiya V Aladinskaya, Victor A Terentiev, Daniil A Polykovskiy, Maksim D Kuznetsov, Arip Asadulaev, Yury Volkov, Artem Zholus, Rim R Shayakhmetov, Alexander Zhebrak, Lidiya I Minaeva, Bogdan A Zagribelnyy, Lennart H Lee, Richard Soll, David Madge, Li Xing, Tao Guo, and Alán Aspuru-Guzik. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nat. Biotechnol.*, 37(9):1038–1040, September 2019.